

ПАРАДИГМЫ ЦИФРОВОЙ ЭКОНОМИКИ

Финансовый университет
при Правительстве Российской Федерации
К столетию со дня образования

ПАРАДИГМЫ ЦИФРОВОЙ ЭКОНОМИКИ

Технологии искусственного интеллекта
в финансах и финтехе

Монография

Под редакцией

М. А. Эскиндарова, В. И. Соловьева

Москва – 2019

УДК 336.02

ББК 65.261

П 18

Рецензенты:

Гатауллин Т. М., доктор экономических наук, кандидат физико-математических наук, профессор кафедры математических методов в экономике и управлении, заместитель директора центра цифровой экономики Государственного университета управления

Назипов Д. А., кандидат экономических наук, первый вице-президент Газпромбанка, заведующий кафедрой «Банковская автоматизация и информационные технологии» Финансового университета

Ответственные редакторы:

Эскиндаров М. А., академик РАО, доктор экономических наук, профессор

Соловьев В. И., доктор экономических наук, профессор

Авторы:

Абдикеев Н. М. (§ 4.3), *Барк В. Б.* (§ 3.2), *Бекетнова Ю. М.* (§ 2.3), *Блохин Н. В.* (гл. 7), *Бондарева Е. В.* (§ 3.3), *Бондаренко О. А.* (§ 2.2), *Брагин В. А.* (§ 3.2), *Бурякова А. О.* (§ 4.5), *Бучнева А. В.* (гл. 7), *Варнавский А. В.* (§ 4.5), *Волкова Е. С.* (§ 2.1), *Гайдамака А. И.* (§ 4.3), *Гисин В. Б.* (§ 2.1), *Золотарева Е. Л.* (§ 3.2, § 3.3), *Карпунин М. А.* (гл. 8), *Коротеев М. В.* (§ 3.4), *Куклина Д. О.* (§ 4.4), *Лосев А. А.* (§ 4.3), *Макрушин С. В.* (гл. 5–8), *Милованов Д. М.* (гл. 6), *Муравьев А. В.* (гл. 5), *Панюкова В. В.* (§ 4.2), *Пухов И. А.* (§ 4.4), *Славгородский А. А.* (§ 4.4), *Славин Б. Б.* (Введение, § 4.1), *Соловьев В. И.* (гл. 1, § 2.1, гл. 3, § 4.3, § 4.4), *Сухань А. А.* (§ 3.4), *Титко М. С.* (§ 4.4), *Титов Н. А.* (§ 3.2, § 3.5, гл. 7), *Феклин В. Г.* (§ 2.2)

П 18 **Парадигмы цифровой экономики:** Технологии искусственно-го интеллекта в финансах и финтехе: Монография / Под ред. М. А. Эскиндарова, В. И. Соловьева. — М.: Когито-Центр, 2019. — 325 с.

ISBN 978-5-89353-550-1

УДК 336.02

ББК 65.261

Монография обобщает новые результаты в области применения технологий искусственного интеллекта в финансах и финтехе. Отдельные главы посвящены экосистеме и ландшафту финтеха (включая обсуждение интеллектуальных технологий, на которых основаны принципиально новые модели бизнеса), применению технологий машинного обучения в кредитном скоринге, оценке устойчивости банков, выявлению банков, вовлеченных в отмывание доходов, прогнозированию рыночных трендов и разработке торговых стратегий, а также использованию технологий вики, вики-графов, семантических сетей и концептуальных карт в задачах разработки экспертных систем, баз знаний и метамоделей банковской архитектуры.

© ФГБОУВО «Финансовый университет», 2019

ISBN 978-5-89353-550-1

СОДЕРЖАНИЕ

Предисловие	9
Введение	12
Глава 1	
Экосистема и ландшафт финтеха.	17
1.1. Финтех.	17
1.2. Предпосылки возникновения финтеха	21
1.3. Экосистема финтеха	23
1.4. Бизнес-модели финтеха	26
1.5. Ландшафт финтеха	39
Глава 2	
Интеллектуальные технологии в кредитном скоринге и оценке устойчивости финансовых организаций.	42
2.1. Применение машинного обучения в кредитном скоринге.	42
2.2. Интеллектуальные методы анализа устойчивости банков	58
2.3. Интеллектуальные методы выявления кредитных организаций, вовлеченных в отмывание доходов.	68

Глава 3	
Интеллектуальные технологии прогнозирования состояний финансовых рынков	81
3.1. Традиционные модели машинного обучения, используемые для прогнозирования рыночных трендов	81
3.2. Алгоритм прогнозирования моментов разворотов рыночных трендов на основе скользящих окон	86
3.3. Применение сверточных нейронных сетей для решения задач классификации и прогнозирования	91
3.4. Применение генеративно-сопоставительных нейронных сетей для решения задач классификации и прогнозирования	113
3.5. Применение систем, основанных на обучении с подкреплением, для решения задач классификации и прогнозирования	122

Глава 4	
Цифровые платформы и маркетплейсы	136
4.1. Цифровые платформы как тренд корпоративной автоматизации	136
4.2. Цифровые платформы в управлении цепями поставок	142
4.3. Межотраслевой маркетплейс для участников создания новых высокотехнологичных продуктов	147
4.4. Интеллектуальная система мониторинга вовлеченности студентов на основе технологий интернета вещей	157
4.5. Формализация экономических отношений в условиях развития автоматизации	169

Глава 5	
Технологии вики и их применение	175
5.1. Вики-технологии	175
5.2. Википедия	195
5.3. Википедия как источник, описывающий окружающую действительность	207

Глава 6	
Технологии семантических сетей и их применение	209

6.1. Введение в семантику и Semantic Web	209
6.2. Стек технологий Семантической паутины	218
6.3. RDF	222
6.4. OWL	227
6.5. SPARQL	234
6.6. Примеры онтологий и семантических сетей	237
6.7. Применение семантических сетей	242

Глава 7

Технологии Вики-графа и их применение	244
7.1. Вики-граф.	244
7.2. Графовые базы данных	247
7.3. Построение Вики-графа	254
7.4. Анализ Вики-графа	259
7.5. Практическое применение Вики-графа.	263

Глава 8

Технологии концептуальных карт и их применение.	268
8.1. Суть и специфика концептуальных карт	268
8.2. Применение концептуальных карт.	272
8.3. Инструменты работы с концептуальными картами	280
8.4. Пример адаптации технологии концептуальных карт для создания прикладного семантической технологии для банка	285

Литература	293
-----------------------------	------------

ПРЕДИСЛОВИЕ

Монография «Парадигмы цифровой экономики» обобщает результаты исследований Финансового университета в области применения цифровых технологий, прежде всего технологий искусственного интеллекта, в частности, инструментов обработки и анализа данных, машинного обучения, интернета вещей в финансовом секторе и особенно в финтехе.

Книга содержит как обзоры новейших мировых исследований, так и оригинальные результаты разработки прикладных систем искусственного интеллекта.

В первой главе проводится обзор экосистемы финансовых инноваций, порожденных технологиями, бизнес-моделей, основанных на этих инновациях, обсуждается ландшафт и экосистема финтеха в России и за рубежом.

Вторая глава посвящена методологии применения машинного обучения в скоринге. В ней проводится обзор методов искусственного интеллекта, используемых в скоринге, а затем обсуждаются две модели машинного обучения, разработанные в Финуниверситете: модель анализа устойчивости коммерческих банков и модель выявления банков, участвующих в отмывании доходов.

Третья глава посвящена интеллектуальным торговым стратегиям, использующимся в управлении активами. Обсуждаются традиционные подходы к прогнозированию моментов разворотов рынка, а также новые результаты, полученные в Финуниверситете с помощью моделей, основанных на сверточных и генеративно-состя-

Предисловие

зательных нейронных сетях, обучении с подкреплением, а также скользящих окнах.

Четвертая глава посвящена цифровым платформам и маркетплейсам. Особое внимание уделено обсуждению перспектив создания цифровых платформ в государственном управлении и логистике. Также предлагается концепция разработки межотраслевого маркетплейса для участников рынков высокотехнологичной продукции. Затем описывается разработанная в Финансовом университете система автоматического мониторинга вовлеченности студентов на основе анализа видеопотоков, поступающих с размещенных в аудиториях камер. Завершается четвертая глава обсуждением формализации экономических отношений в условиях развития применения сквозных технологий.

Пятая глава посвящена технологиям вики и их применению. На данный момент технологии вики и построенная на этих технологиях Википедия оказались самым продуктивным инструментом построения слабо формализованных баз знаний. Помимо общих вопросов, рассматривается возможность применения данных технологий в корпоративной среде. Отдельный подраздел посвящен Википедии как ключевому глобальному информационному артефакту частично структурированного знания.

В шестой главе рассматриваются ключевые элементы современного технологического стека семантических технологий, относящихся к области Семантической паутины (Semantic Web), рассматриваются ключевые понятия и концепции, на которых базируются современные семантические технологии: технология представления данных и метаданных RDF, язык описания онтологий OWL, SPARQL – язык построения запросов к данным, описанным в формате RDF. Обсуждаются примеры построения онтологий и семантических сетей на базе этих технологий, как имеющие вид крупных инфраструктурных проектов, обеспечивающих базис для реализации прикладных семантических технологий, так и описывающие применение семантических сетей в бизнесе, в том числе в банках.

В седьмой главе обсуждаются технологии, которые позволяют на основе массива данных Википедии строить прикладные семантические системы. Рассматривается понятие веб-графа как частного случая графа, построенного на основе корпуса гипертекстовых документов. Дается определение Вики-графа и рассматриваются возможности, которые он предоставляет на фоне аналогичных структур

Предисловие

данных, построенных на базе других массивов гипертекстовых документов. Рассматривается возможность построения аналогичных структур данных для корпоративных и банковских приложений. Важной частью главы является знакомство с ключевыми технологиями для работы с Вики-графом — графовыми базами данных. Далее по результатам проведенного в Финуниверситете исследования русскоязычного сегмента Википедии описывается практически реализованная авторами технология построения Вики-графа, приводятся основные результаты анализа структуры построенного авторами Вики-графа и анализ аналогичных результатов других авторов. В конце главы рассматриваются прикладные аспекты использования вики-графов для банковских и корпоративных приложений.

Восьмая глава посвящена технологиям концептуальных карт и их приложениям. По сути концептуальные карты являются еще одним подходом для формирования баз знаний, готовящих почву для внедрения семантических технологий. Таким образом, технологии построения концептуальных карт наряду с вики-технологиями представляют еще одну форму прикладных семантических технологий. Более того, технологии концептуальных карт могут иметь синергию с вики-технологиями при построении баз знаний и их адаптации к решению бизнес-задач в банках. В седьмой главе раскрывается суть концептуальных карт, основные направления использования этой формы представления информации, приводится анализ прикладных программных пакетов для работы с концептуальными картами, описывается возможность использования концептуальных карт для формирования онтологий и возможности по адаптации и прикладному использованию концептуальных карт в банках и других крупных организациях.

ВВЕДЕНИЕ

28 июля 2017 г. Распоряжением Правительства Российской Федерации № 1632-р была принята программа «Цифровая экономика Российской Федерации». Становлению цифровой экономики в России призвана способствовать и Стратегия развития информационного общества, принятая в мае того же 2017 г.

Под цифровой экономикой понимается «хозяйственная деятельность, в которой ключевым фактором производства являются данные в цифровом виде, обработка больших объемов и использование результатов анализа которых по сравнению с традиционными формами хозяйствования позволяют существенно повысить эффективность различных видов производства, технологий, оборудования, хранения, продажи, доставки товаров и услуг» [1].

В программе «Цифровая экономика» были определены основные показатели цифровизации России по направлениям нормативного регулирования, кадрового обеспечения и образования, формирования технологических заделов и исследовательских компетенций, создания инфраструктуры и системы безопасности.

Обсуждение путей развития экономики в результате цифровизации или диджитализации деятельности предприятий идет уже давно. Так термину «цифровая экономика» уже больше 20 лет (книга канадского ученого и публициста Дона Тапскотта «The Digital Economy» вышла в 1996 г.), а использование информационных технологий (ИТ) в бизнесе вообще имеет полувековую историю.

В России коммерческие компании уже много лет интенсивно внедряют ИТ, а соответствующая отрасль экономики, сформированная компаниями, оказывающими услуги в области информационных технологий, показывает неплохие результаты не только внутри страны, но и на международных рынках.

Цифровые технологии успешно внедряются и в управлении государственными услугами в рамках электронного правительства, и с целью автоматизации деятельности органов власти.

Почему же именно сегодня вопросы цифровизации встали так остро, что понадобилось принятие программы развития цифровой экономики на национальном уровне?

Чтобы ответить на этот вопрос, необходимо рассмотреть в динамике этапы внедрения средств корпоративной автоматизации и эволюцию роли ИТ в экономике. В начальный период своего развития информационные технологии в основном использовались в качестве инструмента ускорения времени и повышения точности планирования и учета ресурсов для обеспечения деятельности сложных многокомпонентных производств, требующих учета многочисленных типов материала, времени поставки, последовательности переделов и т. п.

Неудивительно, что этот этап развития ИТ был связан с созданием отдельных вычислительных центров на крупных предприятиях, в задачу которых входило проведение расчетов.

С развитием средств персонального вычислительного оборудования и средств сетевых коммуникаций информационные технологии стали выполнять еще одну функцию на предприятиях – автоматизацию бизнес-процессов, поддержку коллективной работы в единой информационной среде.

Именно этот этап развития ИТ был ознаменован успехами внедрения комплексных систем класса ERP¹, систем электронного документооборота. По мере совершенствования технологий они стали давать конкурентные преимущества бизнесу, повышая капитализацию компанию.

Началась интенсивная автоматизация всех сфер деятельности компаний, включая внедрение систем класса CRM², позволяющих

1 ERP (*англ.* Enterprise Resource Planning) – планирование ресурсов предприятия.

2 CRM (*англ.* Customer Relationship Management) – управление взаимоотношениями с клиентами.

управлять отношениями с клиентами, систем класса BPM³ для управления эффективностью бизнеса, систем класса ECM⁴ для управления информационным контентом предприятия. Стали востребованы инструменты BI⁵ для анализа коммерческой деятельности и программные продукты для проектного управления. В машиностроении получили распространение PLM⁶ системы, позволяющие управлять созданием и послепродажной эксплуатацией сложных изделий на всем протяжении их «жизни», от разработки проектной документации до утилизации. В строительстве начали использоваться BIM⁷ технологии, своего рода цифровые двойники зданий, позволяющие эффективно выстроить все процессы строительства: от согласования проектной документации до сдачи его в эксплуатацию.

Однако массовая автоматизация бизнеса оказывается еще не была кульминацией в использовании ИТ в экономике, она лишь сформировала основу для технологической трансформации бизнеса, которая и стала визитной карточкой современного цифрового «хайпа».

Но прежде, чем говорить об особенностях следующего этапа — цифровой трансформации, надо отметить, что финансовый, и в первую очередь банковский, сектор (особенно это касается организаций, работающих с розничными клиентами) всегда был лидером в области автоматизации.

Это связано с тем, что аналогично биллинговым системам для телекоммуникационных компаний банковские информационные системы ведения счетов клиентов и электронные платежные системы являются основой бизнеса. По этой причине банки стали первыми, кто выстроил жесткую систему управления бизнес-процессами, внедрил у себя системы управления рисками.

Неудивительно, что и международные банковские стандарты, разрабатываемые Базельским комитетом по банковскому надзору,

-
- 3 BPM (*англ.* Business performance management) — управление эффективностью бизнеса.
 - 4 ECM (*англ.* Enterprise content management) — управление корпоративным контентом.
 - 5 BI (*англ.* Business intelligence) — инструменты аналитической отчетности.
 - 6 PLM (*англ.* Product lifecycle management) — управление жизненным циклом продукции.
 - 7 BIM (*англ.* Building information modeling) — информационное моделирование здания.

в отличие от международных стандартов ISO, используемых обычными предприятиями, более строгие к качеству организации деятельности, к управлению рисками и к информационной прозрачности, и главное — обязательные к исполнению. Именно лидерство финансовых организаций в области автоматизации привело к тому, что они оказались первыми в эпоху цифровой трансформации.

Цифровая трансформация бизнеса — это уже новый этап развития корпоративных ИТ, который связан с тем, что информационные технологии стали широко использоваться не только в деятельности самих предприятий, но и среди населения.

Безусловно, это стало возможным за счет технологических достижений: за счет развития сети Интернет и удешевления телекоммуникаций, в результате появления новых мобильных устройств (от ноутбуков и планшетов до смартфонов), снижения стоимости вычислительного оборудования.

Если на предыдущем этапе ИТ помогали делать бизнес более эффективным, то в условиях цифровой трансформации были «сломааны» организационные границы предприятий: торговые компании через системы B2B⁸ получили доступ к информационным системам поставщиков, заказывая товар порой еще до его производства, у клиентов через личные кабинеты и Интернет появилась возможность покупать прямо со склада торговых организаций, сотрудники компаний получили возможность работать в информационной системе компаний, находясь сколь угодно далеко от нее (в командировке, дома, в отпуске).

«Размывание» границ предприятия как раз и стала основой изменения или трансформации бизнеса, поскольку часть своей деятельности компании стали передавать на самообслуживание клиентам и партнерам.

Знаменитый сервис Booking.com создал цифровую платформу, которая позволила убрать посредников между отелями и их клиентами, причем клиенты не просто сами теперь заказывают себе отели, но и рейтингуяют их, т. е. выступают еще и в роли надзорных органов.

Внедрение цифровых сервисов радикально изменило услуги аренды машин с водителем (такси), переформатировало отрасль средств массовой информации, как по форме (блоги на видеохостингах и в социальных сетях, текстовые каналы в мессенджерах, и т. д.),

8 B2B (англ. Business to business) — информационное взаимодействие между компаниями.

Введение

так и по участникам, допустив в нее самодетельных блогеров и независимых вещателей. И поскольку финансовая отрасль была передовой в области использования ИТ, то она оказалась и в авангарде цифровой трансформации.

Расширение доступа населения к Интернет привело к возможности перевести в сеть платежные услуги — появились так называемые электронные деньги, электронные кошельки. Причем такого рода услуги, как оказалось, вполне могут оказывать и небанковские организации. Развитие мобильных технологий привело к тому, что телеком операторы также стали оказывать платежные услуги, предлагая населению оплачивать те или иные услуги, необязательно телекоммуникационные, с использованием предоплаты за телефон.

Банковские организации не отставали от коллег по расширению рынка услуг за счет небанковских отраслей. Предоставление доступа к банковским сервисам через Интернет дал возможность предложить клиенту услуги страхования, возможность оплатить услуги ЖКХ и купить авиабилеты и т. п. Цифровые технологии привели к реальной трансформации деятельности организаций, и финансовая отрасль оказалась в эпицентре этой трансформации. Финтех, таким образом, можно считать авангардом цифровой трансформации экономики.

Финтех демонстрирует, что цифровая трансформация, изменяя возможности организации, может выводить из-под государственного надзора многочисленные сервисы. Так было с электронными деньгами и платежами через телекоммуникационных операторов, так происходит сегодня через сетевые сообщества, использующие для обмена криптовалюты. Аналогичная потеря государственного контроля наблюдается в цифровизации рынка аренды такси, средств массовой информации.

В условиях цифровой трансформации необходима более динамичная государственная политика, и финтех здесь может стать хорошим полигоном для поиска новых форм регулирования, прежде всего, связанных с привлечением бизнес-сообщества.

Финтех уже сегодня использует все современные цифровые технологии, и искусственный интеллект, и технологии работы с большими данными, и инструментарий распределенных реестров, и биотехнологии. Поэтому его можно использовать и как полигон для обкатки прорывных технологий, обозначенных в правительственной программе «Цифровая экономика».

Глава 1

ЭКОСИСТЕМА И ЛАНДШАФТ ФИНТЕХА

1.1. Финтех

Базельский комитет по банковскому надзору определяет *финансовые технологии*, или *финтех*, как «порожденные технологиями финансовые инновации, приводящие к созданию новых бизнес-моделей, приложений, процессов или продуктов, которые впоследствии скажутся на финансовых рынках, институтах или производстве финансовых услуг» [2].

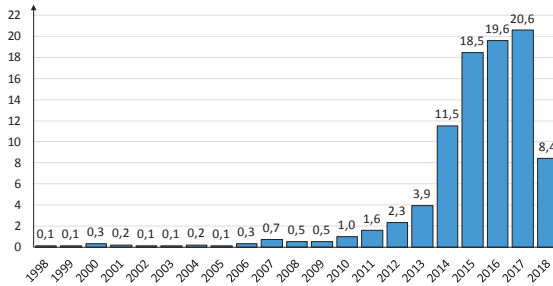
Мировые инвестиции в финтех демонстрируют феноменальный рост. В опубликованном компанией *Venture Scanner* отчете о суммарных инвестициях в финтех-стартапы за первый квартал 2018 г. [3] представлены данные о 1588 компаниях из 65 стран, получивших в 1998–2018 гг. суммарное венчурное финансирование в размере 90,6 млрд долл. США (рисунок 1а).

Кроме того, в отдельном отчете об инвестициях в технологии распределенного реестра [4] сообщается о 330 компаниях, получивших финансирование в размере 6,2 млрд долл. США на инициативы, связанные с технологией блокчейна и криптовалютами (рисунок 1б).

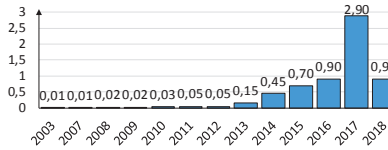
Инвестиции в российский финтех по сравнению с мировым масштабом достаточно малы. По оценке директора Центра финансовых технологий Фонда «Сколково» П. Новикова суммарные инвестиции в российские финтех-компании последние несколько лет держатся на уровне 50 млн долл. США в год на фоне нескольких миллиардов долл. США только в Китае [5].

Глава 1

Консалтинговая компания *KPMG* с 2014 г. ежегодно публикует отчет *Fintech 100*. В отчете *Fintech 100* за 2017 г. [6] представлены 50 компаний — состоявшихся лидеров финтех-отрасли и 50 наиболее динамичных новых финтех-стартапов. Распределение компаний по странам проиллюстрировано рисунком 2, а распределение по областям деятельности — рисунком 3а.



а)



б)

Рис. 1. Суммарные мировые инвестиции в финтех (млрд долл. США, данные за 2018 г. указаны по состоянию на 11.04.2018): а) инвестиции в финтех-стартапы (Financial Technology Startup Highlights – Q1 2018. Venture Scanner. 30.04.2018. URL: <https://www.venturescanner.com/financial-technology>); б) инвестиции в блокчейн-стартапы (Blockchain Technology Sector Overview – Q1 2018. Venture Scanner. 22.04.2018. URL: <https://www.venturescanner.com/blockchain-technology>)

В таблице 1 приведены данные о компаниях, занимающих в данном рейтинге первые 10 мест (пять из них китайские, три — американские).

В 2016 г. во вторую часть рейтинга (в список 50 «восходящих звезд» финтеха) входила российская компания *2can* (АО «Смартфин»), предоставляющая решение для приема платежей с банковских карт через смартфоны. В 2017 г. в рейтинг не попала ни одна российская компания.

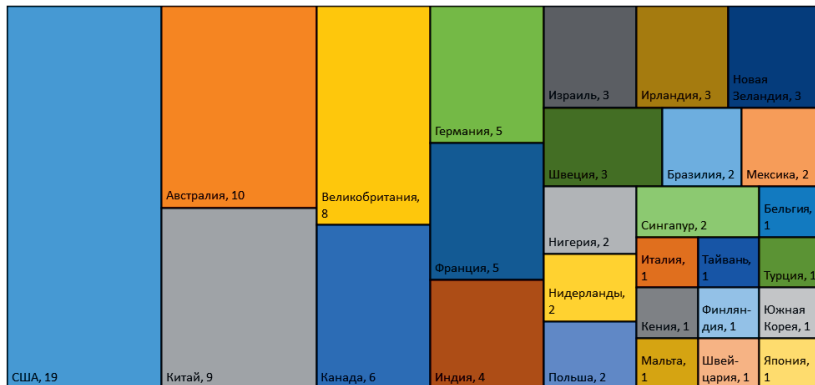


Рис. 2. Распределение финтех-компаний из рейтинга «KPMG Fintech 100» за 2017 г. по странам (2017 Fintech 100: Leading Global Fintech Innovators. KPMG. URL: <https://home.kpmg.com/content/dam/kpmg/it/pdf/2018/03/H2-Fintech-Innovators-2017.pdf>)

Таблица 1

Первые десять мест в рейтинге «Fintech 100» компании KPMG за 2017 г.

Номер в рейтинге	Компания	Сектор финтеха	Страна
1	Ant Financial	платежи	Китай
2	ZhongAn	страхование	Китай
3	Qudian (Qufenqi)	кредитование	Китай
4	Oscar	страхование	США
5	Avant	кредитование	США
6	Lufax	рынки капитала	Китай
7	Kreditech	кредитование	Германия
8	Atom Bank	кредитование	Великобритания
9	JD Finance	кредитование	Китай
10	Kabbage	кредитование	США

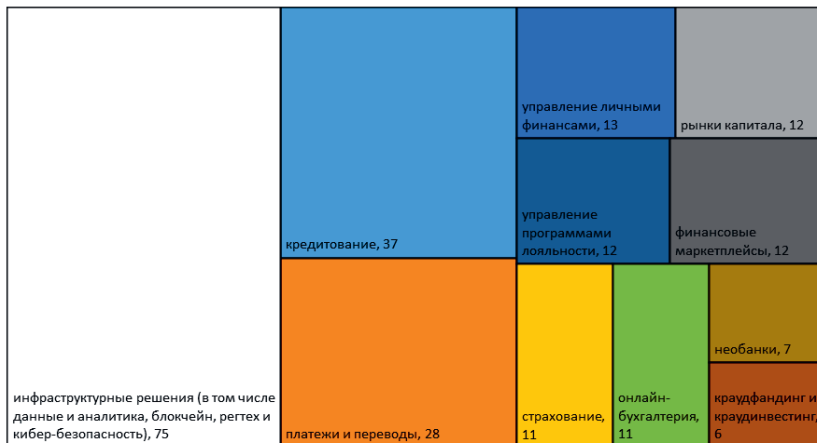
Источник: 2017 Fintech 100: Leading Global Fintech Innovators. KPMG. URL: <https://home.kpmg.com/content/dam/kpmg/it/pdf/2018/03/H2-Fintech-Innovators-2017.pdf>.

Глава 1

Опубликованная компанией *Rusbase* в 2018 г. Карта российского финтех-рынка [7] демонстрирует приблизительно такое же распределение компаний по областям деятельности, как и в мире (рисунок 3б).



а)



б)

Рис. 3. Распределение финтех-компаний по областям деятельности: а) рейтинг «KPMG Fintech 100» (2017) (2017 Fintech 100: Leading Global Fintech Innovators. KPMG. URL: <https://home.kpmg.com/content/dam/kpmg/it/pdf/2018/03/H2-Fintech-Innovators-2017.pdf>); б) карта российского финтех-рынка (2018) (Fintech: Карта российского рынка. Rusbase. URL: <https://rb.ru/fintech>)

Все это свидетельствует о том, что сегодня финтех — это не только горячие обсуждения будущего финансовой сферы в средствах массовой информации, но и реальные проекты, на самом деле меняющие банковскую и финансовую деятельность.

1.2. Предпосылки возникновения финтеха

Интернет-революция, произошедшая в начале 1990-х годов, серьезно повлияла на мировые финансовые рынки. Самое важное, что в результате появления Интернета существенным образом снизилась стоимость финансовых транзакций. Электронные финансовые сервисы, внедрившиеся во всех виды финансовых услуг (в том числе в банковскую деятельность, страхование, торговлю ценными бумагами), позволили физическим и юридическим лицам не только получать информацию о финансовых продуктах и услугах, но и проводить операции со счетами без физического контакта с финансовыми организациями. В 1990-х годах появилось много новых бизнес-моделей в области цифровых финансов, в том числе интернет-банкинг и мобильный банкинг, доступные онлайн брокерские услуги, мобильные платежи. Многие из этих изменений привели к сокращению количества банковских отделений и филиалов.

Особенно сильно интернет-технологии повлияли на банковскую деятельность. Поскольку практически каждый банковский бизнес-процесс в существенной степени основан на обработке данных, за счет внедрения информационных технологий существенным образом увеличилась скорость этих процессов при снижении операционных расходов и улучшении как внутреннего взаимодействия между банковскими подразделениями, так и взаимодействия с существующими и потенциальными клиентами [8, 9].

Еще один пример цифровых финансовых сервисов — онлайн-торговля ценными бумагами, обеспечивающая конкурентное преимущество за счет снижения транзакционных издержек, а также возможности предоставления клиентам доступа к высококачественным аналитическим отчетам инвестиционных компаний.

Распространение смартфонов в середине 2000-х гг. привело к росту предложения и использования мобильных финансовых услуг, которые технологически мало чем отличаются от услуг, предоставляемых через Интернет, но поскольку со смартфонами люди практически не расстаются (в отличие от компьютеров), финансовые и банков-

ские сервисы, доступные с мобильных устройств, дали возможность юридическим и физическим лицам управлять своими банковскими счетами, совершать банковские операции и операции с ценными бумагами в режиме 24/7.

Среди основных трендов, способствовавших появлению финтех-инноваций, следует, в первую очередь, отметить обобществление ресурсов. *Facebook*, самая большая социальная сеть в мире, не владеет пользовательским контентом. *Uber*, самая большая транспортная компания, не владеет автомобилями. *AirBnb*, самая большая гостиничная сеть, не владеет жильем. При этом типичный банк владеет материальными активами (капитал банка, недвижимость, техника и оборудование) и нематериальными активами (лицензии, технологии и бизнес процессы, бренд).

Другой важный тренд – это рост «новой экономики»: темпы роста цены акций традиционных крупнейших компаний сильно отстают от темпа роста компаний «новой технологической волны».

Третий тренд – распространение информационных технологий, Интернета и мобильных устройств: проникновение Интернета в мире растет ежегодно на 11%, и уже 50 лет действует закон Мура, в соответствии с которым мощность процессоров удваивается каждые два года.

Четвертый и пятый тренды – развитие социальных сетей и революция поколений. Сегодняшние пользователи финансовых услуг с рождения привыкли жить с вычислительными устройствами, а число пользователей социальных сетей исчисляется миллиардами.

О финтех-инновациях стали говорить после кризиса 2008 г., когда цифровые и мобильные финансовые сервисы дополнились возможностями искусственного интеллекта и интеграции с социальными сетями. Финтех-стартапы отличались от традиционных финансовых организаций прежде всего предложением новых услуг. Одни стартапы предлагали традиционные банковские услуги тем клиентам, с которыми банки в силу разных причин не работали. Другие финтех-компании предлагали принципиально новые услуги, основанные на обработке и анализе данных. Некоторое время широко обсуждалась перспектива превращения банков в инфраструктурные сервисные организации, предоставляющие финтех-компаниям базовые услуги по управлению счетами клиентов, тогда как финтех-компании будут предоставлять конечным потребителям все финансовые услуги, которые станут нишевыми и дифференцированными.

Но поскольку банки в 2010-х гг. в существенной степени развивались за счет внедрения и совершенствования технологий, фактически именно они получили основные выгоды от финтеха – как путем развития финтех-инициатив внутри себя, так и за счет приобретения финтех-стартапов.

Сегодня большинство банков и финансовых организаций серьезно относятся к финтеху и в своих стратегиях существенное внимание уделяют конкуренции, сосуществованию и сотрудничеству с финтех-стартапами. Как правило, банки позиционируют себя в стратегиях как провайдера финансовых услуг, агрегатора платежей и торговых платформ, а также как маркетплейс – «супермаркет» финансовых услуг, предлагаемых сторонними компаниями, и программ лояльности.

1.3. Экосистема финтеха

В работе [10] экосистема финтеха определяется как взаимодействие предпринимателей, государства и финансовых организаций. В статье [11] выделены пять основных элементов экосистемы финтеха:

- финтех-стартапы (предоставляющие услуги по платежам и переводам, управлению капиталом, кредитованию и финансированию, торговле ценными бумагами, страхованию и т. д.);
- разработчики технологий (предоставляющие услуги в области аналитики больших данных и искусственного интеллекта, технологии блокчейна и криптовалют, облачных вычислений, социальных сетей и т. д.);
- государственные органы – финансовые регуляторы и законодательные органы;
- клиенты – юридические и физические лица;
- традиционные финансовые учреждения (например, традиционные банки, страховые компании, брокерские фирмы и венчурные капиталисты).

В центре экосистемы находятся финтех-стартапы, которые создают в области платежей и переводов, управления капиталом, кредитования и финансирования, операций с ценными бумагами, страхования и др. При этом происходит снижение операционных расходов, открываются нишевые рынки, предоставляются более персонализированные услуги, чем те, которые предлагают традиционные финансовые

организации. В результате происходит разделение финансовых услуг, оказывающее неблагоприятное влияние на традиционные банки [12]. Потребитель получает возможность не полагаться на единственную финансовую организацию для всех своих нужд, а выбирать различные услуги от различных финтех-компаний. Так, юридическое лицо может получить кредиты не в традиционном банке, а в сервисах взаимного кредитования «Поток» и «Город денег», оперативно, за две недели, выпустить облигации с помощью компании «Бонди-Бокс», принимать платежи с помощью сервиса *Simplepay* и использовать онлайн-бухгалтерию «Мое дело». Физические лица также могут воспользоваться площадками взаимного кредитования типа *Bezbanka.ru* для получения кредита под более низкий процент, чем в обычных банках, без справок о доходах и поручителей, а другие физические лица на тех же площадках могут давать в долг, зарабатывая больше, чем по вкладам в банках. Небанковские платежные сервисы, например, «Пэй-ап. Платежи», позволяют безопасно совершать покупки в интернет-магазинах, оплачивать различные услуги, а также переводить деньги на карты любых российских банков. Чтобы перевести деньги на карту, выпущенную зарубежным банком, можно воспользоваться онлайн-сервисом *PayPal*.

Разработчики технологий предоставляют цифровые платформы для социальных сетей, искусственного интеллекта и обработки данных, облачных вычислений, кибербезопасности, блокчейна, мобильных технологий. Облачные технологии позволяют финтех-стартапам быстро развертывать онлайн-платформы без капитальных вложений в инфраструктуру и при необходимости эластично масштабироваться. На основе технологий искусственного интеллекта и обработки данных разрабатываются продвинутое скоринговые системы, алгоритмические торговые стратегии и системы робоэдвайзинга, создаются уникальные персонализированные клиентские предложения, совершенствуется клиентский сервис. Социальные сети способствуют росту сообществ, например, по взаимному кредитованию. Операторы мобильной связи могут предоставлять инфраструктуру для совершения мобильных платежей и мобильного банкинга. В свою очередь, индустрия финтеха генерирует существенные потоки доходов в пользу технологических компаний.

Естественно, законодатели и регуляторы являются важнейшими участниками экосистемы финтеха, поскольку именно они определяют правила. Так, сегодня активно обсуждается законопроект

«О цифровых финансовых активах», направленный на регулирование отношений, возникающих при создании, выпуске, хранении и обращении цифровых финансовых активов, а также осуществлении прав и исполнении обязательств по смарт-контрактам [13], а также законопроект «Об альтернативных способах привлечения инвестиций (краудфандинге)», который предназначен для регулирования отношений по привлечению инвестиций с использованием технологических платформ краудфандинга, а также по организации работы таких платформ [14]. Мегарегулятор в лице Банка России может упростить многие процедуры, например, путем создания специальной регулятивной площадки для апробации инновационных финансовых технологий и сервисов. С другой стороны, предписание мегарегулятора может остановить на длительное время работу финтех-сервиса (так, старейший российский сервис взаимного кредитования «Вдолг.ру» был приостановлен в 2016 г. из-за того, что его партнер, РНКО РИБ, получил предписание Банка России на ограничение проведения платежей в пользу физических лиц и индивидуальных предпринимателей).

Если важнейшими клиентами в традиционном банкинге являются крупные организации, то финтех прежде всего ориентирован на физических лиц, малый и средний бизнес. Значительную часть пользователей финтех-услуг составляют люди из «поколения Y», и это благоприятно для финтеха, поскольку технологически развитые представители этого поколения в ближайшее несколько десятилетий будут составлять основную часть населения.

Традиционные финансовые организации также занимают важное место в экосистеме финтеха. Имея конкурентные преимущества перед финтех-стартапами в масштабах и финансовых ресурсах, в начале финтех-эры традиционные банки и финансовые компании стремились воспрепятствовать развитию финтеха, понимая его разрушительное влияние на традиционные финансовые услуги. Однако в настоящее время произошла смена отношения традиционных банков и финансовых компаний к финтеху, многие банки развивают внутри себя финтех-инициативы. Так, сервис взаимного кредитования «Поток» предлагается Альфа-Банком, Сбербанк активно использует сервисы, основанные на машинном обучении, в различных процессах — от кредитной фабрики до клиентского сервиса, управляющая компания «Альфа-Капитал» внедряет разработанные совместно с Финуниверситетом инвестиционные стратегии, осно-

ванные на машинном обучении, ведущие банки объединились в Ассоциацию ФинТех и совместно разрабатывают ряд проектов на основе децентрализованной платформы обмена и хранения информации «Мастерчейн» (депозитарную систему для учета закладных, реестр банковских гарантий, цифровой аккредитив, проект «Знай своего клиента»). В апреле 2018 г. банки – члены Ассоциации ФинТех запустили прототип Системы быстрых платежей для осуществления мгновенных межбанковских платежей и переводов по номеру телефона и адресу электронной почты. Активно ведутся Ассоциацией ФинТех и работы в области идентификации, аутентификации и управления цифровой идентичностью, а также открытых интерфейсов.

Среди необанков, т. е. банков, не имеющих физических отделений и полностью работающих в цифровом пространстве, можно отметить два цифровых банка, являющиеся самостоятельными, независимыми единицами и работающими под собственной лицензией: *Tinkoff Bank* и Модульбанк. Кроме того, существуют банки *RocketBank*, *Touch Bank* и «Точка», работающие под лицензиями традиционных банков и фактически представляющих собой отдельные направления в работе этих традиционных банков. Интересный проект *Talk-Bank* работает под лицензией Транскапиталбанка и ведет все свои операции в мессенджере *Viber* (до блокировки *Telegram TalkBank* работал и в этом мессенджере).

1.4. Бизнес-модели финтеха

Финтех открывает новые способы хранения, заимствования, инвестирования финансовых средств, платежей и покупок, а также обеспечения безопасности. Существует девять основных типов бизнес-моделей финтеха:

- платежи и переводы;
- управление активами;
- краудфандинг;
- взаимное кредитование;
- операции с ценными бумагами;
- онлайн-банкинг;
- онлайн-бухгалтерия;
- страховые сервисы;
- криптовалюты и другие применения технологии блокчейна.

Платежи и переводы

Платежные сервисы занимают значительную долю среди всех финтех-проектов. Это связано с относительной простотой платежей как финансовых продуктов. Во-первых, финтех-компании, специализирующиеся на платежах, могут относительно быстро и дешево наращивать клиентскую базу. Во-вторых, именно в области платежей сегодня уровень развития технологий постоянно позволяет внедрять новые возможности, основанные на инновациях. В-третьих, платежи – это финансовая услуга, наиболее часто используемая как юридическими, так и физическими лицами.

Среди платежных сервисов для физических лиц можно отдельно выделить мобильные кошельки (например, *Qiwi*, «Яндекс. Деньги»), прямые платежи между картами разных банков («Пэй-ап. Платежи»), сервисы обмена валюты (многие онлайн-обменники, например, российские сервисы *New Line* и *ProstoCash*, позволяют обменивать биткойны или эфириумы на рубли или доллары; хотя автору не известны российские платформы взаимного (*peer-to-peer*) обмена традиционных валют, например, обменять рубли на другие валюты и совершить при необходимости трансграничный перевод можно с помощью британских сервисов *Kantox* и *TransferWise*).

В значительной степени развитие платежных финтех-проектов в России связано с большим числом сервис-провайдеров, принимающих плату за своих услуги, и их потребностью делать это оперативнее и с удобными технологиями для клиентов.

Тем не менее, в платежах, переводах и обмене валют в России господствующее положение остается у банков, которые предоставляют платежные и валютно-обменные услуги на высоком уровне и с обеспечением достаточной скорости (пока речь не заходит об обмене криптовалют и трансграничных переводах, где финтех-сервисы предоставляют клиентам возможности, которые традиционные банки дать не могут).

Появляются инновационные платежные сервисы, основанные на использовании технологий искусственного интеллекта. Так, в мае 2017 г. Банк «Открытие» представил мобильное приложение «Открытие. Переводы» на платформах *Android* и *iOS*. Одна из функций этого приложения – «перевод по фото»: денежный перевод можно совершить, сфотографировав получателя платежа на камеру смартфона или выбрав его фотографию из галереи (рисунок 4). Далее избра-

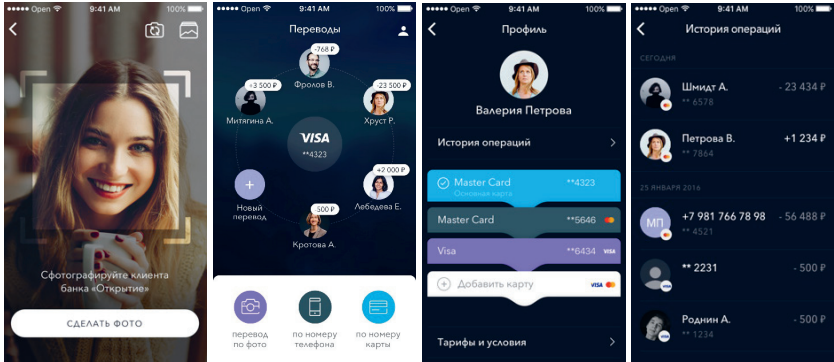


Рис. 4. Интерфейс мобильного приложения «Открытие. Переводы»

жение передается в банковскую систему распознавания лиц, которая идентифицирует получателя и возвращает в приложение номер его банковской карты. Отправителю остается выбрать свою карту, с которой будет осуществлен перевод.

Кроме того, в потребительском сегменте активно развивается внедрение услуг, упрощающих совершение платежей в традиционных магазинах. Так, в рекордно короткие сроки в России стали стандартом де-факто *POS*-терминалы, поддерживающие бесконтактные технологии *MasterCard PayPass* и *Visa PayWave*, сегодня никого не удивляет, когда клиенты совершают платеж, прикладывая к терминалу смартфон, поддерживающий технологии *Google Wallet*, *Apple Pay* или *Samsung Pay*. Аналогичные технологии внедряют производители самой различной мобильной электроники (например, умные часы *Garmin* поддерживают технологию *Garmin Pay*). В настоящее время идет масштабное внедрение технологии *Samsung Pay* для подключения к телефонам карт российской платежной системы «Мир».

Управление активами

Популярностью у клиентов пользуются финтех-решения для управления личными финансами, которые позволяют клиентам визуализировать свои финансовые операции на счетах в разных банках, анализируют денежные потоки пользователей, выявляют закономерности, формируют персональные рекомендации. В процессе работы с витринами таких решений клиенты осознают важность контроля

своих расходов и накоплений, у них повышается интерес к работе с банком. Визуализация расходов и прогнозная модель помогает планировать жизненные активности и достигать финансовых целей.

Одной из популярных бизнес-моделей управления финансовыми активами являются автоматизированные менеджеры по управлению капиталом – робоэдвайзеры, предоставляющие финансовые консультации за небольшую часть стоимости реального консультанта и использующие алгоритмы искусственного интеллекта и обработки данных. Например, сервис *Cashoff* позволяет управлять бюджетом, агрегируя транзакции по категориям, предоставляя отчеты о тратах и персональные рекомендации о сокращении расходов (рисунок 5).

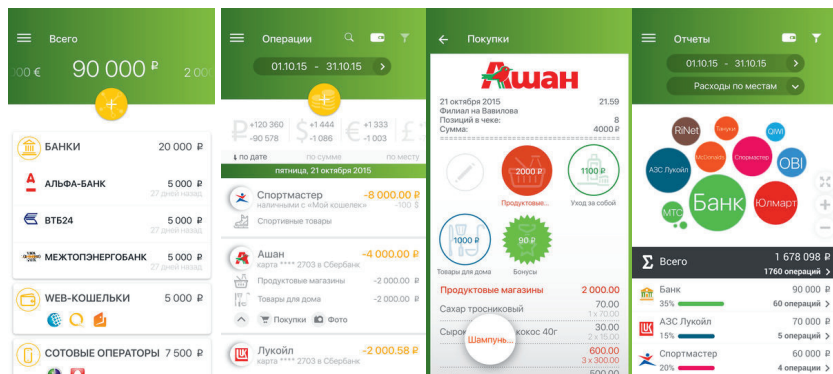


Рис. 5. Интерфейс мобильного приложения *Cashoff*

Сервис *Panda Money* позволяет копить на реализацию мечты в игровой форме: каждое накопление – это процесс кормления малыша Панды (рисунок 6).

Краудфандинг

Краудфандинг – это бизнес-модель сервисов, которые позволяют привлекать средства для создания новых продуктов, организации компаний или благотворительной деятельности. Платформа краудфандинга обеспечивает взаимодействие инициаторов проектов, нуждающихся в финансировании, и потенциальных вкладчиков, которые могут быть заинтересованы в поддержке этих проектов. Платформа автоматизирует прием средств на проект по различным

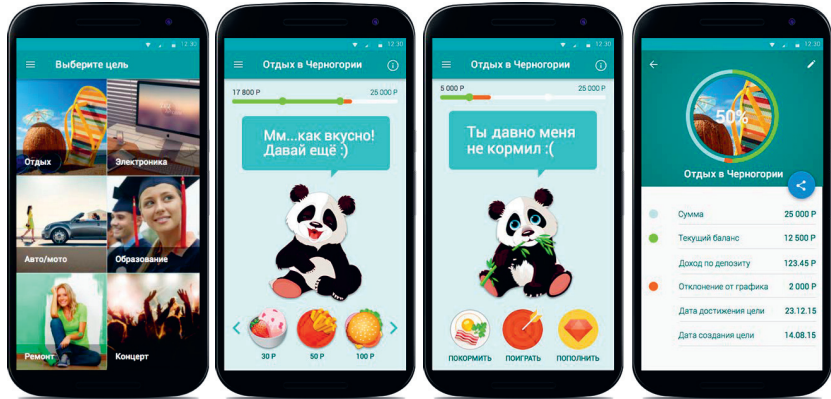


Рис. 6. Интерфейс мобильного приложения *Panda Money*

каналам финансирования и обеспечивает документальное сопровождение сделки.

Объем мирового рынка краудфандинга в 2015 г. составлял 34,4 млрд долл. Ожидается, что к 2020 г. он превысит 100 млрд долл. США и создаст 2 млн новых рабочих мест [15].

Рынок краудфандинга в России также растет. По данным Банка России объем российского рынка краудфандинга вырос с 6 млрд руб. в 2016 г. до 11 млрд руб. в 2017 г. [16].

Наиболее популярные формы краудфандинга – это краудфандинг, основанный на вознаграждениях (например, вкладчики получают возможность приобрести финансируемый ими продукт с существенной скидкой, как в российской платформе *BoomStarter* (рисунок 7) – аналоге самой популярной в мире краудфандинговой платформы *KickStarter*) и краудфандинг, основанный на пожертвованиях (участники делают пожертвования, ничего не ожидая взамен).

Интересный пример нового сервиса в области краудфандинга – платформа «Скидываемся», выросшая из проведенного банком «Открытие» в 2016 г. конкурса финтех-стартапов *OpenFights*. Платформа позволяет автоматизировать сбор денег на общее дело – на подарок другу, на форму для хоккейной команды или на экскурсию для класса. Сервис позволяет автоматически напоминать всем участникам, которым разослана ссылка на событие, о необходимости скинуться, автоматически переводит собранные деньги на карту организатора. При этом участники видят в реальном вре-

Экосистема и ландшафт финтеха

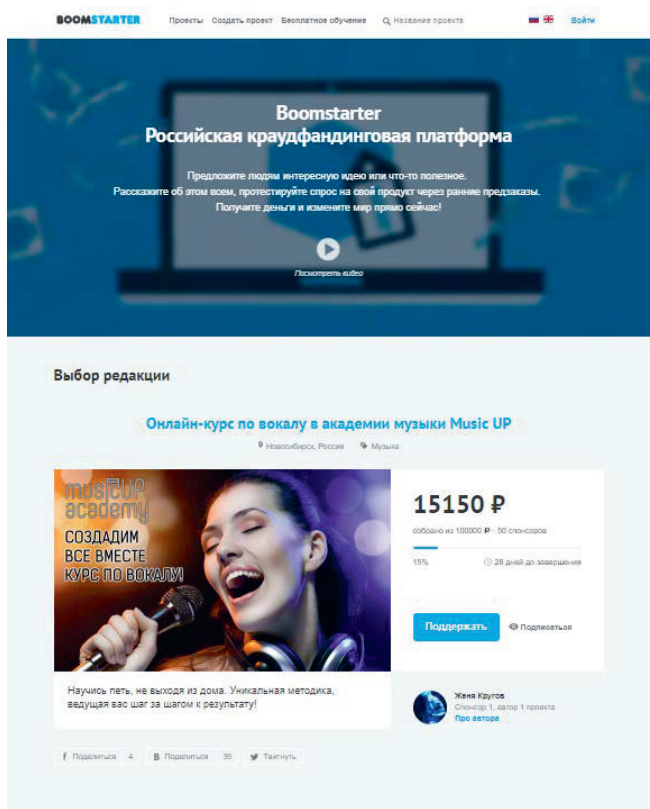


Рис. 7. Интерфейс платформы Boomstarter

мени, сколько собрано денег, кто уже внес свою часть, а кто еще нет (рисунки 8).

Взаимное кредитование

Такую бизнес-модель реализуют финтех-проекты, автоматизирующие прямое кредитование физических и юридических лиц без посредничества банков (*peer-to-peer*, *P2P*). *P2P*-площадки размещают

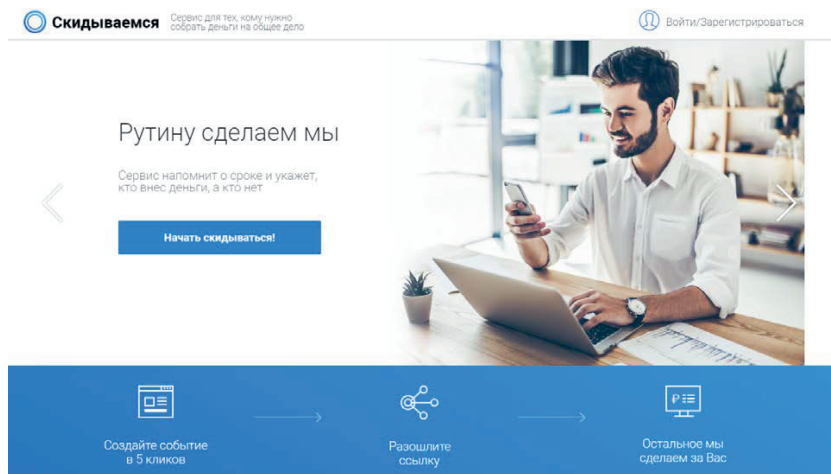


Рис. 8. Интерфейс сервиса «Скидываемся» (<https://chipin.online/landing>)

предложения о наличии свободных средств у потенциальных кредиторов и собирают заявки на получение кредита от потенциальных заемщиков. Оформление сделок происходит, как правило, в рамках общегражданских правовых отношений в виде заключения между физическими лицами договора займа.

Наиболее популярные российские платформы взаимного кредитования – «Поток», «Безбанка», «Город денег».

Например, по состоянию на начало мая 2018 г. в сервисе «Безбанка» было заключено более 46 тыс. сделок по взаимному кредитованию на общую сумму более 348,5 млн руб. (рисунок 9).

Объем сделок, совершенных на старейшей российской платформе взаимного кредитования «Вдолг.ру» до момента приостановки ее работы в 2016 г. составил 422,5 млн руб.

Сервис «Поток» привлекает деньги индивидуальных кредиторов – физических лиц, и диверсифицирует инвестиции, распределяя их минимум по 20 компаниям, которые получают кредиты. При этом полная статистика доходности публикуется в открытом доступе. С 2017 по май 2018 г. были выданы 2411 займов, из которых 741 выплачен полностью, у 1670 еще не истек срок погашения. При этом дефолты были объявлены в 3,2% случаев, а средняя чистая доходность составила 21,8% годовых [17].

Экосистема и ландшафт финтеха

96 bezbanka.ru
 БезБанка – без проблем

Время в Москве
 3 мая 2018, чт
 14: 05: 09

ВОЙТИ | ЗАРЕГИСТРИРОВАТЬСЯ

В системе БезБанка зарегистрировано 133583 пользователей. Заключено 46362 успешных сделок на общую сумму 348 674 489 рублей.

О СИСТЕМЕ | КАК ВЗЯТЬ | КАК ДАТЬ | FAQ/ПОМОЩЬ | ПЕРЕВОДЫ ОТ 0%

Взять займ | Дать займ

рублей
 на _____ месяцев
 НАЙТИ | Расширенный поиск

УВЕЛИЧИТЬ УРОВЕНЬ ДОВЕРИЯ ЛЕГКО!
 Закажи проверку в
 Кредитном бюро Русского Стандарта

РУССКИЙ СТАНДАРТ
 КРЕДИТНОЕ БЮРО

Топ заявок на получение займа

	тип	в долг	к возвр.	срок	перепл.
Светлана Н. 79	★	100000	140000	20 д	42%
Сергей И. 63	★	150000	200000	6 м	34%
Сергей И. 63	★	150000	200000	6 м	34%
Светлана Б. 40	★	100000	130000	50 д	32%
Светлана Б. 40	★	300000	400000	6 м	34%
Сергей И. 63	★	150000	190000	4 м	28%
Сергей И. 63	★	100000	120000	2 м	22%
Юрий О. 12	★	100000	273900	60 м	175%
Елена Ч. 111	★	35000	50000	15 д	44%
Елена Ч. 111	★	50000	65000	15 д	32%
Вадим П. 43	★	100000	195000	14 м	97%
Нурана Г.	★	6000	11000	7 д	88%
Нурана Г.	★	3000	5000	7 д	71%
Николай Б. 116	★	5000	7000	8 д	43%
Олег К. 201	★	5000	7200	8 д	47%
Евгения Е. 16	★	15000	19438	30 д	32%

Все заявки (409) на 18 926 626 руб.

Топ-предложения

	тип	в долг	к возвр.	срок	перепл.
Маргарита Б. 2	★	4000	6000	7 д	53%
Маргарита Б. 2	★	3000	5000	7 д	71%
Валентина П. 13	★	5000	7125	8 д	45%
Валентина П. 13	★	4000	6000	7 д	53%
Валентина П. 13	★	3000	4125	7 д	42%
Наталья Д. 17	★	4000	6200	10 д	58%
Наталья Д. 17	★	5000	8000	10 д	63%
Наталья Д. 17	★	3000	4700	7 д	61%
Наталья Д. 17	★	10001	14600	10 д	48%
Юлия В. 106	★	30000	32000	7 д	9%
Юлия В. 106	★	30000	32000	7 д	9%
Юлия В. 106	★	30000	32000	7 д	9%
Юлия В. 106	★	30000	32000	7 д	9%
Юлия В. 106	★	30000	32000	7 д	9%
Юлия В. 106	★	30000	32000	7 д	9%
Юлия В. 106	★	100000	160000	6 м	62%

Все предложения дать займ (119) на 3 027 508 руб.

Рис. 9. Интерфейс сервиса «Безбанка» (<https://bezbanka.ru>)

Операции с ценными бумагами

К этому классу бизнес-моделей относятся решения для автоматизации работы на финансовых рынках: выбор инструмента, оценка и прогноз поведения цен с выдачей персональных рекомендаций, электронный документооборот с брокерами, операции по брокерским счетам, портфельная оптимизация и портфельное управление. Широкий круг платежных инструментов для ввода/вывода денег, поддержка мобильных платформ делает такие системы удобными не только для инвесторов-любителей, но и для казначейств юридических лиц, предпочитающих вкладывать свободные средства в ценные бумаги.

Например, сервис *eToro* фактически представляет собой социальную сеть инвесторов, позволяющую копировать инвестиционные стратегии других участников проекта (рисунок 10).

А управляющая компания «Альфа-Капитал» предлагает своим клиентам инвестиционную стратегию, основанную на разработанных специалистами департамента анализа данных, принятия решений и финансовых технологий Финуниверситета алгоритмах машинного обучения для определения моментов разворота рыночных трендов [18].

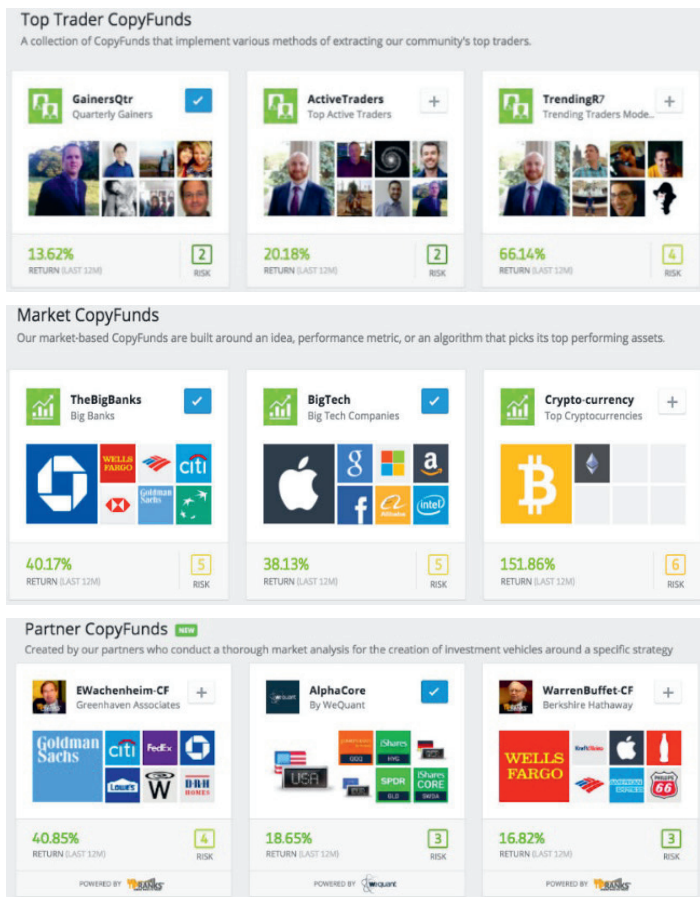


Рис. 10. Копирование стратегий в сервисе *eToro*

Онлайн-банкинг

К обязательному функционалу мобильного банковского приложения уже относятся перевод между собственными счетами/картами и на карты других клиентов; оплата коммунальных услуг, мобильной и стационарной телефонной связи, Интернета, ТВ; создание и редактирование шаблонов платежей; обмен валют; вход по короткому цифровому коду или графическому ключу, открытие вкладов и счетов. Активно расширяется функционал переводов с карты на карту и производных от них услуг, в том числе, пополнение счета/карты с карты другого банка, а также сервисов по анализу движения денежных средств.

В ежегодном исследовании эффективности мобильных банковских приложений для физических лиц *Marksw Webb Mobile Banking Rank 2017* [19] первые пять мест в категории «Приложения для iPhone» занимают *Tinkoff Bank*, Бинбанк, Почта Банк, Альфа-Банк и Банк «Русский Стандарт», в категории «Приложения для Android» – *Tinkoff Bank*, Бинбанк, Альфа-Банк Почта Банк, и Сбербанк. Пять лучших мобильных банков на платформе *Windows Mobile* – это *Tinkoff Bank*, Бинбанк, Банк «Уралсиб», Почта Банк и *RocketBank*. На планшетах *iPad* лучшими являются *Tinkoff Bank*, Почта Банк, Альфа-Банк, ВТБ24 и Сбербанк, а на планшетах *Android* – Почта Банк, ВТБ24, Банк «Русский Стандарт», Райффайзенбанк и Банк «Открытие».

Онлайн-бухгалтерия

К этому классу относятся решения для услуг по персональному бухгалтерскому учету для юридических лиц. Как правило, индивидуальные предприниматели, малый и семейный бизнес, ограничены в средствах и не могут себе позволить держать выделенный офис – бухгалтера, юриста, секретаря и пр. А экономя на качестве и кадрах, делая все самостоятельно, они отвлекаются от основного бизнеса, и в силу недостаточной компетенции в области бухгалтерского учета и отчетности могут получить значительные потери от претензий контрольных органов и выявленных случаев нарушения законодательства.

По-видимому, уже 1,5 млн собственников и руководителей небольших компаний (при общем количестве юридических лиц, зарегистрированных в России, составляющем чуть менее 5 млн) поль-

зуются облачными сервисами, позволяющими за 10–20 тыс. руб. в год с помощью сайтов и мобильных приложений самостоятельно выставлять счета клиентам, рассчитывать зарплату сотрудникам, вести бухгалтерский и налоговый учет, готовить отчетность и автоматически сдавать ее в налоговые органы через Интернет, автоматически обмениваться данными с банками и т. д. В таких сервисах может достаточно быстро освоиться практически любой пользователь, даже не имеющий никакого представления о бухгалтерском учете.

Среди популярных российских сервисов онлайн-бухгалтерии – «Мое дело», «Контур. Бухгалтерия», «Контур. Эльба», «Небо», «Кнопка», «Финолог» и др. Свои сервисы онлайн-бухгалтерии предлагают и банки, начиная со Сбербанка.

Страховые сервисы

Страховые услуги, основанные на использовании различных технологий, в том числе телематических датчиков, беспроводной связи, мобильных приложений и др., и облегчающие как удобство приобретения страховых продуктов, удобство их использования, а также снижение цены.

К сожалению, проекты цифровых страховых сервисов, основанных на использовании геопозиционирования для изменения страховых тарифов в реальном времени в зависимости от географического положения транспортного средства (в простейшем варианте с оплатой страховой премии только за время или расстояние, которое транспортное средство находится в движении – как в популярном американском сервисе *metromile*), изменение тарифов в зависимости от стиля вождения, определяемого на основании данных телеметрии и использования технологий интернета вещей, успехом в России пока не увенчались.

Также автору неизвестны российские платформы взаимного страхования, аналогичные созданной в 2010 г. в Германии платформе *Friendsurance*, которая ежегодно возвращает клиентам часть собранных страховых премий, которые не были потрачены на погашение ущерба (в 2017 г. на *Friendsurance* был застрахован стотысячный клиент).

По-видимому, в ближайшее время в России должны, наконец, появиться новые сервисы, в том числе в области сегментации и ско-

ринга страхователей, создания инновационных страховых продуктов, оптимизации операционных затрат страховщиков за счет внедрения искусственного интеллекта, распределения страховых рисков и затрат между компанией и клиентами, агрегирования целевого трафика и внедрения технологий интернета вещей.

Криптовалюты и другие применения технологии блокчейна

Блокчейн – это децентрализованный электронный механизм эмиссии, учета, обмена специальными цифровыми активами. При этом обычно информация о транзакциях не шифруется и всегда доступна в открытом виде. Криптография используется не для ограничения доступа к данным о транзакциях, а для гарантирования неизменности цепочки блоков транзакций.

И хотя помимо криптовалют большинство блокчейн-проектов пока не принесли значимых результатов, прогресс уже заметен: во многих областях появляются реальные проекты, основанные на блокчейне, а не просто обещания.

Важным направлением в технологии блокчейна являются смарт-контракты – основанные на криптографии алгоритмы, описывающие условия сделок в соответствии с правилами платформы, в рамках которой реализуются эти смарт-контракты, и обеспечивающие автоматическое выполнение этих условий.

Использование технологии блокчейна повышает уровень доверия к голосованиям и гарантирует неизменность результатов, так как любой желающий может следить за ходом голосования, став участником сети блокчейна.

Сегодня в Москве на основе технологий блокчейна работает площадка «Активный гражданин» для проведения открытых референдумов в электронной форме. Проект реализован на платформе *Etherium*. Для каждого голосования создается смарт-контракт, в котором осуществляется хранение и подсчет голосов. При этом все голоса также дублируются на компьютеры независимых наблюдателей в режиме реального времени (рисунок 11).

Технология блокчейна позволяет заменить и упростить услуги удостоверения, например, нотариального, а также ведения реестров. С января по июль 2018 г. в Москве идет пилотный проект по использованию блокчейна для мониторинга достоверности сведений из Единого государственного реестра недвижимости.

28 ноября 2017 г. Сбербанк и Альфа-Банк провели первую платежную транзакцию с помощью технологии блокчейна. Мегафон перечислил 1 млн рублей со счета в Альфа-Банке своей дочерней компании «Мегалабс» на счет в Сбербанке. Для проведения перевода Сбербанк установил три узла сети блокчейна, а Альфа-Банк – один узел. Также банки установили клиентское приложение. Может показаться, что гораздо проще было воспользоваться традиционными способами платежей, однако данный проект на самом деле является знаковым: в результате получен опыт межбанковских расчетов без использования международных межбанковских финансовых каналов связи типа *SWIFT*.

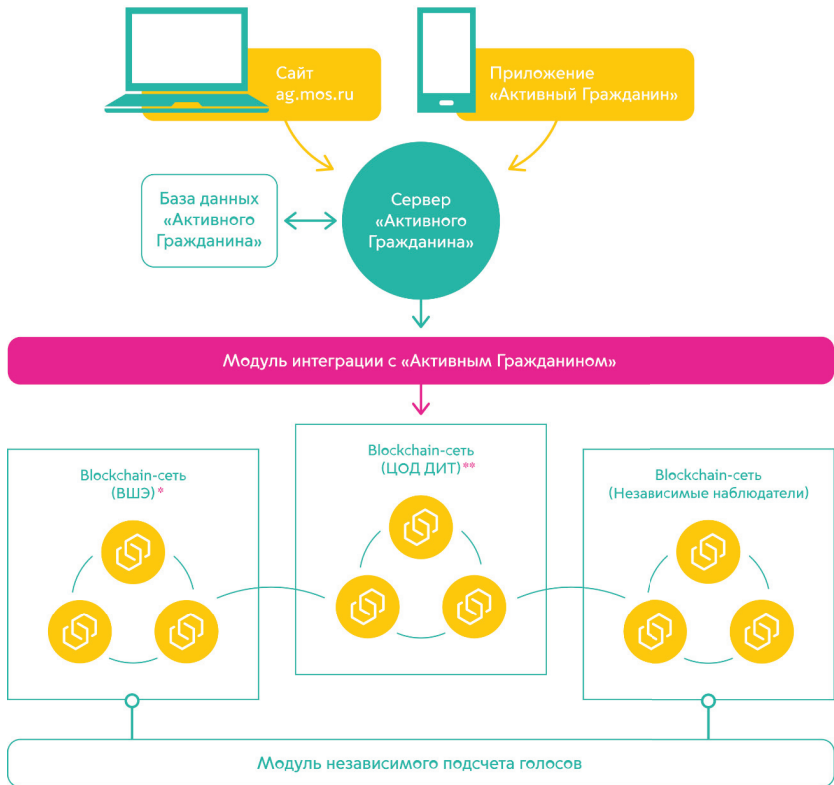


Рис. 11. Схема работы блокчейн-платформы «Активный гражданин» (<https://ag.mos.ru/blockchain>)

1.5. Ландшафт финтеха

В таблице 2 представлена авторская классификация финтех-инноваций по уровню изменения пользовательского опыта и уровню изменения бизнес-модели.

Таблица 2
Ландшафт финтеха

	Новые продукты и услуги, новые функциональные возможности	Улучшения существующей функциональности продуктов и услуг
Новые бизнес-модели	<ul style="list-style-type: none"> • ICO • криптовалюты • алгоритмический трейдинг • краудфандинг • взаимное кредитование • быстрые международные платежи 	<ul style="list-style-type: none"> • смарт-контракты • открытые интерфейсы • приложения – агрегаторы карт
Улучшения традиционных бизнес-моделей	<ul style="list-style-type: none"> • робоэдвайзинг • электронные кошельки • необанки • управление личными финансами • Интеллектуальный скоринг 	<ul style="list-style-type: none"> • мобильные платежи • блокчейн-реестры • регтех • кибербезопасность

В будущем индустрия финансовых услуг будет сильно отличаться от сегодняшней. Ожидается, что рынок будет более конкурентным и более эффективным, что обеспечит клиентам гораздо больший выбор. Новые коммерческие предложения, новые продукты, услуги и новые рынки появятся в индустрии финансовых услуг.

Но по состоянию на 2018 г. банки все еще продолжают доминировать в российском финансовом секторе. Они значительно меняются, будучи действительно заинтересованными в инвестировании в цифровые технологии и финтех-инновации, которые могли бы способствовать повышению эффективности и снижению затрат на существующие бизнес-процессы, внедрению новых услуг, улучшению обслуживания клиентов и клиентского опыта.

Российский «Тинькофф Банк» является крупнейшим небанком в мире без филиалов и офисов, существуют и другие российские небанки, как самостоятельные, так и работающие в форме подразделений традиционных банков. При этом абсолютное большинство

российских традиционных банков предоставляют инновационные онлайн и мобильные банковские приложения с функциями интеллектуальных платежей и переводов, а также решениями по управлению личными финансами. Многие банки предоставляют своим клиентам также услуги онлайн-бухгалтерии, конкурирующие с новыми небанковскими сервисами типа «Мое Дело».

Крупные российские банки экспериментируют с распределенными реестрами, пытаясь получить преимущества от внедрения блокчейна и смарт-контрактов в свои процессы. Есть также много блокчейн-стартапов, но значимых услуг, основанных на использовании технологий распределенных реестров, на российском рынке пока не появилось.

Немногие из банков внедряют нетрадиционные услуги, такие как взаимное кредитование (в качестве примера можно привести сервис «Поток» Альфа-Банка, конкурирующий с небанковскими платформами, такими как BezBanka.ru).

В области краудфандинга и новых интеллектуальных услуг по операциям с ценными бумагами существуют стартапы, оказывающие услуги в ограниченном масштабе, традиционными банками и инвестиционными компаниями ведутся исследования и разработки, но эти рынки в России только зарождаются.

Что касается интеллектуальных страховых услуг, то в России их на сегодняшний день практически нет, все инициативы по внедрению технологичных страховых продуктов в России не удались.

Следует отметить, что участники российской финтех-экосистемы имеют относительно финтеха разные точки зрения.

Российские банки ищут пути улучшения процессов. Например, только некоторые из российских банков используют искусственный интеллект для улучшения обслуживания клиентов, а модели машинного обучения в большинстве банков внедряются в основном не для создания новых ценностных предложений, а для снижения затрат на существующие процессы. Наиболее интересными финтех-проектами для банков являются открытые API, различные реестры, кибербезопасность и регтех.

Государство, в свою очередь, не видит больших различий между финтехом, регтехом и криптовалютами и направляет основные усилия на инициативы, связанные с регулированием обращения криптовалют и внедрением блокчейна в регтех и кибербезопасность.

Экосистема и ландшафт финтеха

С другой стороны, клиенты заинтересованы в новых и более удобных функциях мобильных приложений, они ждут новых ценностных предложений, включая быстрые международные денежные переводы, новые услуги по робоэдвайзингу, управлению личными финансами, взаимному кредитованию.

Финтех-инициативы пока не привели к радикальной трансформации финансового сектора в России. Для развития рынка необходимо согласование точек зрения всех участников экосистемы, и когда оно произойдет, появится много новых инновационных финансовых услуг. Но поскольку нет никакой гарантии, что банки будут продолжать доминировать в новой экосистеме, они сегодня стараются развивать финтех-инициативы внутри себя, стимулируя разработчиков технологий и финтех-стартапы направлять свои усилия на такие инновации, которые направлены на совершенствование процессов, а не на открытие новых рынков.

Глава 2

ИНТЕЛЛЕКТУАЛЬНЫЕ ТЕХНОЛОГИИ В КРЕДИТНОМ СКОРИНГЕ И ОЦЕНКЕ УСТОЙЧИВОСТИ ФИНАНСОВЫХ ОРГАНИЗАЦИЙ

2.1. Применение машинного обучения в кредитном скоринге

Качественный и количественный рост рынка финансовых инструментов поставил перед финансовыми институтами задачу изучения закономерностей, характеризующих современные финансовые рынки, и разработки адекватных инструментов оценки различного рода рисков и управления ими. Одно из важных направлений современных исследований в этой сфере — оценка кредитных рисков. Рост спроса на потребительские кредиты привел к увеличению конкуренции на рынке кредитования. Банки и другие кредитные институты столкнулись с необходимостью обрабатывать большие объемы данных со все возрастающей скоростью. Современные требования к объему обрабатываемых данных и скорости их обработки таковы, что процессы обработки должны быть практически полностью автоматизированы. Это требования автоматизации распространяется не только на непосредственную цифровую обработку, но и на процедуры настройки, адаптации и даже построения соответствующих количественных моделей. Традиционно используемые в кредитовании модели, такие, как скоринг, стали комбинироваться с новыми вычислительными методами, которые относят к области так называемого машинного обучения или интеллектуального анализа данных.

В общем *кредитный скоринг* может быть определен как технология, позволяющая кредитной организации решить вопрос о предоставлении кредита заявителю с учетом его характеристик, таких как воз-

раст, доход, семейное положение и др. Естественно, такие технологии возникли вместе с появлением торговли и потребностью в кредитовании. Идеи и методы скоринга, соответствующие их современному пониманию, были впервые сформулированы в работе [20].

После принятия соглашений Базель II (и особенно Базель III) стало возможным и необходимым применять процедуры внутреннего рейтинга для оценки общих параметров риска. Это сделало более значительной роль кредитного скоринга и заставило финансовые институты постоянно совершенствовать используемые ими количественные модели.

Достаточно полное представление о работах, посвященных методам кредитного скоринга, до 2003 г. дает работа [21]. Обзоры более поздних публикаций можно найти в [22, 23]. Многочисленные обзоры посвящены отдельным технологиям кредитного скоринга и сравнительному анализу применяемых методов.

Обсудим основные работы, в которых применяются методы кредитного скоринга, основанные на методологии интеллектуального анализа данных.

Линейная регрессия

Линейная регрессия связывает характеристики заемщика, представленные вектором $x \in R^n$ с целевой переменной $y \in [-1, 1]$:

$$y = \beta_0 + \langle \beta, x \rangle + \varepsilon,$$

где ε – случайная ошибка с нулевым средним. При решении вопроса об отнесении y к тому или иному классу величина $\beta_0 + \langle \beta, x \rangle$ трактуется как условное математическое ожидание $E(y|x)$. В работе [24] на основе линейной регрессии построены скоринговые карты. Отметим, что в последние годы линейная регрессия в чистом виде не используется, хотя по-прежнему служит важным инструментом в смешанных моделях.

Логистическая регрессия

Логистическая регрессия – один из основных инструментов кредитного скоринга. В публикациях логистическая регрессия, как правило, используется для сравнения с другими методами [25, 26, 27] или в комбинации с ними [28, 29].

Логистическая регрессия используется в кредитном скоринге для вычисления вероятности отказа $P(y=1|x)$ выдачи заемщику, имеющему характеристики. Вероятность представляется в виде

$$P(y=1|x) = \frac{1}{1 + e^{-(\alpha + \beta^T x)}}.$$

Для оценки коэффициентов α и β_i (координаты вектора β) используется метод максимального правдоподобия. Оценка проводится на обучающем множестве.

Дискриминантный анализ

Дискриминантный анализ – один из наиболее популярных методов скоринга и кредитного скоринга, в частности. Дискриминантный анализ восходит к работе Фишера [30]. Дискриминантный анализ был одним из первых методов, применявшихся для построения систем кредитного скоринга. Проблемы, связанные с применением дискриминантного анализа в кредитном скоринге, проанализированы в [31]. В настоящее время дискриминантный анализ продолжает использоваться в кредитном скоринге непосредственно [32]. Он часто служит эталоном, с которым сравниваются другие методы [33]. Ряд исследований связан с повышением точности дискриминантного анализа за счет применения новых процедур [34].

Деревья решений

Метод берет свои истоки в работе [35]. Представление о современном состоянии дел дает работа [36]. Если говорить о кредитном скоринге, то здесь деревья применяются в основном для классификации [37].

Опишем коротко существо методов, связанных с построением деревьев. Говорят, что переменная X порядковая, если принимаемые ею числовые значения упорядочены существенным для классификации образом. В противном случае переменную называют категориальной. Алгоритм автоматического обнаружения взаимосвязи (Automatic Interaction Detector, AID) – один из первых алгоритмов построения классификационных деревьев – последовательно расщепляет данные в каждом узле. В случае порядковой переменной ветвление происходит по условиям вида « $X \leq c$ », в случае категориальной переменной – по условиям вида « $X \in A$ ». Пусть $S(t)$ – мно-

жество номеров данных в обучающей выборке, относящихся к узлу t . Обозначим через \bar{y}_t среднее (по $S(t)$) значение объясняемой переменной Y . Величина

$$imp(t) = \sum_{i \in S(t)} (y_i - \bar{y}_t)^2$$

служит показателем «загрязненности» узла t . Алгоритм AID выбирает такое расщепление, которое минимизирует сумму показателей «загрязненности» по непосредственно следующим узлам. Процесс расщеплений заканчивается, когда снижение загрязненности становится меньше предустановленного порога.

Алгоритмы типа THAID (Theta Automatic Interaction Detector) распространяют описанный метод на категориальные переменные. Здесь в качестве показателя загрязненности используется энтропия или индекс Джини. В более современных алгоритмах CART (Classification and Regression Trees) правила останковки, применяемые в алгоритмах AID и THAID, заменены правилами выращивания и удаления новых ветвей. Применяются также алгоритмы CHAID (Chi-squared Automatic Interaction Detector) и C4.5. В [38] приведены сравнительные характеристики различных алгоритмов кредитного скоринга, включая и алгоритмы CART. Отмечено, что алгоритмы CART уступают другим алгоритмам. Однако некоторые новые идеи и усовершенствования моделей, связанных с построением деревьев, позволяют существенно повысить их эффективность [39, 40].

К алгоритмам, связанным с деревьями принятия решений, при-мыкают алгоритмы извлечения знаний (Rule Extraction, RX), ориентированные на работу с большими данными [41].

Машины опорных векторов

Метод опорных векторов как метод статистической классификации был предложен в работе [42]. Суть метода состоит в следующем. Пусть задано обучающее множество $\{(x^{(j)}, y^{(j)})\}_{j=1, 2, \dots, l}$, где $x^{(j)} \in X \subset R^n$ — признаковое описание объекта, $y^{(j)} \in \{-1, 1\}$ — бинарный классификатор. Уравнение вида

$$\langle w, x \rangle - w_0 = 0, w \in R^n,$$

задает гиперплоскость с нормальным вектором w , разделяющую в пространстве R^n классы «хороших» ($y^{(j)} = 1$) и «плохих» ($y^{(j)} = -1$) объектов.

Оптимальная разделяющая гиперплоскость определяется как решение оптимизационной задачи:

$$\begin{aligned} & \|\mathbf{w}\| \rightarrow \min; \\ & y^{(j)}(\langle \mathbf{w}, \mathbf{x}^{(j)} \rangle - w_0) \geq 1, \quad j = 1, 2, \dots, l. \end{aligned}$$

В случае, когда разделяющая гиперплоскость существует, величина $2/\|\mathbf{w}\|$ — ширина полосы между точками разных классов. Задача нахождения оптимальной разделяющей гиперплоскости может быть решена с использованием теоремы Куна–Таккера. Пусть

$$L(\mathbf{w}, w_0, \boldsymbol{\lambda}) = \frac{1}{2} \langle \mathbf{w}, \mathbf{w} \rangle - \sum_{j=1}^l \lambda_j (y^{(j)} (\langle \mathbf{w}, \mathbf{x}^{(j)} \rangle - w_0) - 1),$$

соответствующая функция Лагранжа. Объект обучающей выборки $x^{(j)}$ называется *опорным вектором*, если $\lambda_j > 0$ и $\langle \mathbf{w}, x^{(j)} \rangle - w_0 = y^{(j)}$. Вектор \mathbf{w} является линейной комбинацией опорных векторов:

$$\mathbf{w} = \sum_j \lambda_j y^{(j)} \mathbf{x}^{(j)}.$$

Таким образом, для фактического построения вектора \mathbf{w} используется сравнительно небольшое число объектов обучающей выборки. Это свойство разреженности отличает метод опорных векторов от классических линейных разделителей типа дискриминанта Фишера.

Если разделяющая плоскость не существует (обучающая выборка линейно не разделима), постановка оптимизационной задачи корректируется: к целевой функции добавляется сумма штрафов за ошибки.

Возможен также переход к нелинейному разделителю с использованием ядра. Под ядром понимается функция $K(x, x')$, $x, x' \in X$ такая, что $K(x, x') = \langle \varphi(x), \varphi(x') \rangle$ для некоторого отображения $\varphi: X \rightarrow R^m$. Используя отображение φ , линейный разделитель можно строить в пространстве R^m [43].

Задача квадратичной оптимизации в методе опорных векторов может быть сформулирована в двойственной форме: найти

$$\max_{\boldsymbol{\lambda}} \left(\sum_j \lambda_j + \frac{1}{2} \sum_{i,j} \lambda_i \lambda_j y^{(i)} y^{(j)} K(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}) \right)$$

при выполнении условий $0 \leq \lambda_j \leq c_j$ для всех j и $\sum_j \lambda_j y_j = 0$.

Параметры c_j контролируют относительную ценность показателей. Наиболее употребительны следующие ядерные функции:

$K(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}) = \langle \mathbf{x}^{(i)}, \mathbf{x}^{(j)} \rangle$ — линейная модель;

$K(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}) = (\langle \mathbf{x}^{(i)}, \mathbf{x}^{(j)} \rangle + 1)^d$ — полиномиальная модель степени d ;

$K(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}) = \exp\left(-\frac{\|\mathbf{x}^{(i)} - \mathbf{x}^{(j)}\|^2}{2\sigma^2}\right)$ — гауссова радиальная базисная функция (RBF) с параметром σ .

Для нового объекта предсказание строится по формуле:

$$\hat{y} = \operatorname{sgn}\left(\sum_j \lambda_j y^{(j)} K(\mathbf{x}^{(j)}, \mathbf{x}) + b_j\right),$$

где:

$$b_j = \sum_i \lambda_i y^{(i)} K(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}).$$

Работа [44] — одна из первых, в которой метод опорных векторов использовался для решения задачи кредитного скоринга. Система опорных векторов относительно семейства ядер использовалась для кредитного скоринга в работе [45].

Байесовские сети

Отправной точкой для применения байесовских сетей в кредитном скоринге послужила, вероятно, работа [46]. В этой работе приведено обобщение так называемого простого (наивного) байесовского метода, в соответствии с которым выбирается с наибольшей апостериорной информацией. Применение наивного байесовского метода обосновано в случае, когда атрибуты независимы. Как отмечают авторы указанной работы, в кредитном скоринге это предположение нереалистично: например, нельзя игнорировать взаимосвязь таких показателей, как возраст, образование, доход. Авторы развили идеи работы [47]. В самом общем виде байесовская сеть представляет собой ацикличный ориентированный граф. При обучении формируются условные распределения вероятности вида $P(Y|X_p, \dots, X_k)$, где Y — вершина, а X_p, \dots, X_k — ее «родители» на графе. Байесовская сеть определяет совместное распределение вершин. Например, наивный Байесовский метод получается, если взять категориальную переменную в качестве корневой вершины, а все атрибуты — в качестве ее «детей». Неформально, обучение байесовской сети состоит

в ее максимальной адаптации к обучающему набору данных. Оптимизация проводится относительно скоринговой функции. Наиболее употребительными являются байесовская скоринговая функция и функция, основанная на принципе минимальной длины описания (MDL). Эти функции асимптотически приводят к одинаковому результату обучения, однако на конечных наборах функция MDL зарекомендовала себя лучше.

Пусть $B=(G, \Theta)$ — Байесовская сеть (G — граф, Θ — соответствующее распределение вероятностей), а $D=\{u_1, \dots, u_n\}$ — обучающий набор (каждое u_i присваивает значения всем вершинам графа). Тогда

$$MDL(B|D) = \frac{\log N}{2} |B| - LL(B|D),$$

где $|B|$ — число параметров сети,

$$LL(B|D) = \sum_{i=1}^N \log(P_B(u_i))$$

измеряет объем информации, необходимой для того, чтобы описать D , основываясь на распределении вероятностей P_B .

Скоринговая функция MDL асимптотически корректна.

Байесовские сети применялись для кредитного скоринга во многих работах, см., напр.: [48, 49, 50, 51, 52, 53].

Нейронные сети

Нейронная сеть преобразует набор входных переменных в набор выходных переменных и моделирует как линейные преобразования, так и нелинейные. Преобразования осуществляются с помощью нейронов, представляющих собой упрощенную модель нейронов головного мозга. Нейроны связаны в сеть односторонними каналами передачи информации. Каждый нейрон может быть активирован поступающими входными сигналами, и в активном состоянии выдает выходные сигналы. В нейронной сети имеется слой входных нейронов — это те нейроны, на которые поступают значения входных переменных, слой выходных нейронов — из выходных сигналов этих нейронов формируются выходные переменные, и скрытые слои. Нейронные сети различаются своей структурой, числом скрытых слоев, функциями активации.

В работе [54] проанализированы пять моделей нейронных сетей, используемых в кредитном скоринге: многослойный персептрон (MLP); смесь экспертов (МОЕ); сеть радиальных базисных функций (RBF); квантование обучающего вектора (LVQ); нечеткий адаптивный резонанс (FAR). Эффективность применения нейронных сетей перечисленных типов в кредитном скоринге сравнивалась с эффективностью применения классических параметрических методов (линейный дискриминантный анализ и логистическая регрессия), непараметрических методов (k ближайших соседей или k -NN, ядерной оценке плотности) и классификационных деревьев. Полученные результаты показали, что многослойные персептроны показывают далеко не самую высокую точность, сети типа смеси экспертов и сети радиальных базисных функций показывают в кредитном скоринге вполне удовлетворительный результат. Из классических методов наиболее точным оказался метод логистической регрессии. Сети, основанные на нечетком адаптивном резонансе, оказались наименее точными. Не уступающие другим сетям по распознаванию «плохих» заемщиков, они существенно хуже работают по распознаванию «хороших» заемщиков.

Генетические алгоритмы

Определенная специфика применения генетических алгоритмов в кредитном скоринге состоит в том, что популяция образована классификационными деревьями. Алгоритмы мутации и скрещивания применяются к деревьям. В остальном структура алгоритмов стандартная. После создания исходной популяции повторяются процессы мутации и скрещивания с последующей оценкой. В качестве оценки берется относительное число ошибок классификации. В работе [55] показано, что на тестовых наборах результаты работы генетических алгоритмов (при 1000 поколений) оказались в числе лучших.

Комбинированные методы

К числу гибридных и комбинированных относятся методы, в которых применяются различные техники кредитного скоринга для повышения эффективности. Наиболее употребительны три метода комбинирования (ensemble methods): беггинг (bagging – bootstrap aggregating), бустинг (boosting) и стекинг (stacking).

Беггинг был введен в работе [56]. Основная идея метода – построение набора предикторов, которые в совокупности (после определенного агрегирования) дают более совершенный предиктор.

Схематично беггинг применительно к кредитному скорингу выглядит следующим образом. Предполагается, что имеется алгоритм обучения, который по обучающему множеству L строит предиктор $\varphi(x, L)$, выдающий u при заданном x . Основываясь на обучающем множестве L , можно построить набор обучающих множеств $\{L_k\}_{k=1, \dots, K}$ (как правило, того же объема, что и L). Эти множества состоят из тех же объектов, выбранных случайным образом из L (возможно, с повторениями). Положим K_+ равным числу тех k , для которых $\varphi(x, L_k)$ дает положительный ответ. Агрегированный предиктор $\bar{\varphi}$ выдает положительный ответ, если $K_+ > 0,5$.

Применение беггинга оказывается особенно эффективным в тех случаях, когда основной алгоритм обучения неустойчив – сильно зависит от небольших изменений в обучающем множестве.

Основная идея бустинга – сформировать на основе слабого (в смысле точности) алгоритма сильный алгоритм классификации. В процессе формирования сильного алгоритма слабый алгоритм «доучивается» за счет того, что перераспределяются веса примеров из обучающей выборки: в случае верного распознавания вес снижается. В случае неверного – повышается. Представление о бустинге дает следующий пример.

Пусть X – пространство, $\{(x^{(j)}, y^{(j)})\}_{j=1, 2, \dots, 1}$ – обучающая выборка. Базовый алгоритм запускается в серии раундов $t = 1, \dots, T$. Обозначим через $D_t(j)$ вес, присвоенный объекту в раунде t (первоначальное распределение весов $D_0(j)$ можно взять равномерным). Задача обучения в раунде t – найти такое отображение $h_t(x)$ со значениями в $\{-1, 1\}$, которое минимизирует вероятность ошибки:

$$\varepsilon_t = \sum_{h_t(x^{(j)}) \neq y^{(j)}} D_t(j).$$

Обновление весов происходит следующим образом. Пусть

$$\alpha_t = \frac{1}{2} \ln \left(\frac{1 - \varepsilon_t}{\varepsilon_t} \right).$$

Тогда

$$D_{t+1}(j) = \frac{D_t(j) \exp(-\alpha_t y^j h_t(x^{(j)}))}{Z_t},$$

где Z_i – нормирующий множитель (так, чтобы $D_{i+1}(j)$ было распределением). Финальный алгоритм распознавания имеет вид:

$$H(x) = \text{sign} \left(\sum_{t=1}^T \alpha_t h_t(x) \right).$$

Этот способ бустинга, основанный на использовании экспоненциальной функции потерь, носит название *AdaBoost*. Он заставляет алгоритм переобучаться при наличии большого количества шумовых прецедентов. Для минимизации данного эффекта можно использовать логистическую функцию потерь (такой алгоритм называется *LogitBoost*).

При стекинге происходит комбинирование нескольких алгоритмов с помощью некоторого комбинатора. Как правило, в роли комбинатора выступает логистическая регрессия. Теоретические основы стекинга были заложены в работе [57].

В кредитном скоринге применение комбинированных методов широко распространено. Характерными примерами могут служить работы [58, 59, 60].

Общий вид модели кредитного скоринга

Рассмотрим в общем виде модель кредитного скоринга и методы сравнения разных моделей.

Условимся называть исход хорошим (*Good*), если $y=0$, и плохим (*Bad*), если $y=1$. В классической постановке задача прогнозирования состоит в том, чтобы по набору наблюдаемых объектов (x, y) определить $E[Y|x] = E[y=1|x]$. Если бы были известны условные вероятности $P(\text{Bad}|x)$, принимать оптимальные решения о предоставлении кредита было бы несложно. Пространство атрибутов, обычно слишком велико, чтобы можно было эмпирически оценить вероятности $P(\text{Bad}|x)$. Стандартный подход – построение скоринговой функции $s(x)$. Апостериорная вероятность $P(\text{Bad}|s) = P(\text{Bad}|s(x)=s)$ используется для построения прогнозов, как замена $P(\text{Bad}|x)$.

Пусть A – модель кредитного скоринга. Значение скоринговой функции $s_A(x)$ можно рассматривать как реализацию некоторой случайной величины S_A . Обозначим через $f(s_A|y)$ плотность вероятности условного прогноза S_A при данном y , а через $v(s_A)$ вероятность того, что значение скоринговой функции окажется равным S_A .

Основы сравнения моделей кредитного скоринга заложены в работах [61, 62, 63, 64].

Пусть A и B – скоринговые модели. Говорят, что модель A является достаточной для модели B , если существует функция h , обладающая следующими свойствами:

- 1) $h(s_B|s_A) \geq 0$ для любых s_A, s_B ;
- 2) $\sum_{s_B} h(s_B, s_A) = 1$ при любом s_A ;
- 3) $\sum_{s_A} h(s_B|s_A) f_A(s_A|y) = f_B(s_B|y)$

Если A является достаточной для B , то B можно считать более неопределенной – функция h придает значениям s_B дополнительную случайность.

Говорят, что модель B не родственна модели A , если u не зависит от s_B при заданном s_A , т. е. $P(y|s_A, s_B) = P(y|s_A)$.

Для заданных скоринговых моделей A и B определим комбинированную скоринговую модель C , полагая $s_C = P(\text{Good}|s_A, s_B)$. Комбинированная модель является достаточной для моделей A и B . Модель A является достаточной для модели C тогда и только тогда, когда модель B не родственна модели A .

Рассмотрим теперь оценку скоринговой модели с точки зрения полезности.

Допустим, что предоставление кредита хорошему заемщику приносит доход в размере 1, а плохому – в размере $-\alpha \leq 0$ (убыток). Положим $\pi(s) = P(\text{Good}|s)$. Ожидаемый доход R при выдаче кредита заемщику, имеющему скоринг s , составляет:

$$E[R|s] = \pi(s) - \alpha(1 - \pi(s)).$$

Будем также считать, что отказ в кредите приносит доход 0 независимо от типа заемщика. Решение о выдаче кредита принимается, если $E[R|s] \geq 0$. Поскольку функция $\pi(s)$ монотонно возрастает, существует такое значение s^* , что $E[R|s^*] = 0$. Если значение скоринговой функции превосходит s^* , кредит одобряется, если нет – заемщик получает отказ.

Таким образом,

$$E[R] = \sum_{s \geq s^*} E[R|s] v(s).$$

Поскольку s^* зависит от α , математическое ожидание дохода $E[R]$ также зависит от α . Эта величина может использоваться для сравнения моделей кредитного скоринга: скоринговая модель A является достаточной для скоринговой модели B тогда и только тогда, когда $E_A[R] \geq E_B[R]$ для всех α .

Оценка качества алгоритмов кредитного скоринга

Одним из способов определения качества модели машинного обучения является разделение выборки на обучающую, которая используется для идентификации параметров алгоритма, и контрольную, для каждого объекта которой проводится сравнение класса, предсказанного алгоритмом, и истинного класса объекта.

При этом наиболее распространенные методы оценки алгоритмов кредитного скоринга основываются на матрице ошибок: все объекты контрольной выборки разбивают на четыре категории в зависимости от комбинации истинного ответа y и ответа $a(x)$, выданного алгоритмом:

	$a(x) = 1$	$a(x) = 0$
$y = 1$	<i>TP</i>	<i>FN</i>
$y = 0$	<i>FP</i>	<i>TN</i>

Здесь *TP* – сокращение для *True Positive*, *FN* – *False Negative*, и аналогично в двух оставшихся случаях.

Поскольку целью применения алгоритмов классификации в кредитном скоринге является сортировка объектов скоринга на хорошие и плохие, эффективность алгоритмов оценивается путем сопоставления для каждого объекта из контрольного набора данных класса, спрогнозированного алгоритмом, с реальным классом этого объекта.

Задача кредитного скоринга имеет две особенности. Во-первых, классификация плохого кредита как хорошего обходится дороже, чем классификация хорошего кредита как плохого, а во-вторых, в обучающей выборке хороших клиентов всегда больше чем плохих.

В связи с первой особенностью в задаче кредитного скоринга применяются следующие метрики качества алгоритмов:

Глава 2

$Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$	— доля правильно классифицированных кредитов;
$Precision = \frac{TP}{TP+FP}$	— точность, т. е. доля правильно классифицированных плохих кредитов среди всех наблюдений, отнесенных алгоритмом к плохим кредитам;
$Recall = \frac{TP}{TP+FN}$	— полнота, т. е. оценка способности алгоритма распознавать плохие кредиты;
$Negative Predictive Value = \frac{TN}{TN+FN}$	— доля правильно классифицированных хороших кредитов среди всех наблюдений, отнесенных алгоритмом к хорошим кредитам;
$Specificity = \frac{TN}{TN+FP}$	— специфичность, т. е. оценка способности алгоритма распознавать хорошие кредиты;
$F1 Score = \frac{2(Precision \times Recall)}{Precision + Recall}$	— среднее гармоническое точности и полноты;
$False Negative Rate = \frac{FN}{TP+FN}$	— доля плохих кредитов, неправильно отнесенных к хорошим.
$False Positive Rate = \frac{FP}{TN+FP}$	— доля хороших кредитов, неправильно отнесенных к плохим.

Кривая ошибок (Receiver Operating Characteristic, ROC) отображает изменение отношения доли *Recall* верно классифицированных плохих кредитов в их общем количестве к доле *False Positive Rate* хороших кредитов, неправильно отнесенных к плохим, при варьировании порога решающего правила.

ROC-кривая получается следующим образом. Предположим, что результат работы алгоритма $a(x)$ зависит от параметра, например, порогового значения, и алгоритм имеет вид:

$$a(x) = \text{Entier} [s(x) > s^*].$$

При $s^* = \infty$ получаем $SEN = 0$ и $FPR = 0$, при $s^* = -\infty$ $SEN = 1$ и $FPR = 1$. Когда s^* изменяется от $-\infty$ до ∞ , точка (FPR, SEN) описывает кривую, которая называется ROC-кривой. Площадь под кривой ошибок (*AUC*) служит показателем качества алгоритма. Равенство $AUC = 0,5$ означа-

ет, что алгоритм относит объекты к категориям наугад. Чем больше AUC , тем качественнее алгоритм. Часто используется также характеристика, называемая индексом Джини (площадь между кривой и диагональю):

$$Gini = 2AUC - 1.$$

Важной проблемой при построении скоринговых моделей является тот факт, что доля плохих кредитов в выборке сильно меньше доли хороших (обычно от 2 до 30%). В такой ситуации малую ошибку на обучающей и тестовой выборках может давать модель, которая предлагает всех клиентов признавать хорошими.

Возможными решениями этой проблемы являются введение различных стоимостей ошибок первого и второго рода или модификация обучающей выборки для изменения репрезентативности выборки.

Для изменения репрезентативности выборки используют два основных метода: дублирование миноритарного класса (*oversampling*) и удаление мажоритарного класса (*undersampling*). Недостатком первого метода является тот факт, что простое дублирование прецедентов может не влиять никаким образом на одни методы обучения, и вести к переобучению других. При удалении объектов, относящихся к мажоритарному классу, возможна потеря важной для классификации информации, что также нежелательно. Наиболее часто для решения проблемы несбалансированности выборки применяется метод синтетического размножения объектов миноритарного класса SMOTE (*Synthetic Minority Oversampling Technique*), который реализуется следующим образом:

1. Вычисляется разность $d = x_b - x_a$ между векторами x_a , x_b признаков соседних объектов a , b из миноритарного класса.
2. Вектор признаков для нового синтезированного объекта:

$$X_{\bar{a}} = X_a + cd, \text{ где } c \sim N(0,1)$$

Существуют разные модификации метода SMOTE, в которых при генерации объектов миноритарного класса используются ближайшие соседи, как из миноритарного, так и из мажоритарного класса, и генерируемые объекты располагаются ближе к границе разделения классов или дальше от нее.

Однако на практике метод SMOTE очень часто приводит к переобучению моделей, кроме того, он весьма затратен по используемым

вычислительным ресурсам и по времени, и проблема несбалансированности классов в скоринговых выборках, как правило, представляет собой отдельную сложную задачу.

Программная реализация алгоритмов машинного обучения в кредитном скоринге

Программные продукты, используемые для автоматизации решения задач интеллектуального анализа данных и машинного обучения, можно разделить на три класса:

- 1) коммерческие статистические пакеты;
- 2) открытые среды;
- 3) облачные решения.

Исторически в банках для решения задач анализа данных, в частности, связанных со скорингом, использовались коммерческие программные продукты, наиболее часто применялся набор продуктов SAS, реже пакеты IBM SPSS и Statistica. Эти три линейки продуктов предоставляют схожие функциональные возможности, включающие средства аналитической подготовки данных, готовые и настраиваемые шаблоны алгоритмов машинного обучения, в том числе моделей линейной и логистической регрессии, деревьев и лесов решений, градиентного бустинга, опорных векторов, нейронных сетей и др. Кроме того, в этих пакетах возможна настройка параметров моделей и использование интерактивных техник оценки качества.

В последние годы банки, не отказываясь полностью от применения коммерческих пакетов типа SAS, стали использовать и открытые среды Python/R/Spark. Преимуществом этих сред является, прежде всего, возможность использования гораздо большего количества алгоритмов, чем в коммерческих пакетах. Но если, например, в здравоохранении и промышленности с распространением открытых сред Python и R от применения коммерческих пакетов практически отказались в пользу использования открытых библиотек, то в банках решения SAS пока используются чаще, чем Python и R.

Язык R создавался как специальное средство для статистических вычислений, он стал первой открытой средой, которая начала активно использоваться для анализа данных. Наиболее часто используемые библиотеки для машинного обучения в R – это gpart и CARET (алгоритмы классификации и регрессии), randomForest (алгоритм случай-

ных лесов), nnet (нейронные сети), e1071 (одна из первых библиотек машинного обучения в R, содержащая реализацию метода опорных векторов, наивный байесовский классификатор и ряд других методов), kernlab (метод опорных векторов), gbm (градиентный бустинг), ROCR (визуализация метрик качества алгоритмов классификации).

Язык Python стал самым популярным средством для анализа данных после выхода отлично документированной библиотеки scikit-learn, в которой реализовано большое количество алгоритмов машинного обучения. Кроме scikit-learn, популярны также библиотеки TensorFlow и Theano (эти библиотеки также реализуют различные методы анализа данных, но выигрывают у scikit-learn только в количестве реализованных техник работы с нейронными сетями). Для использования аппарата детерминированного, нечеткого и байесовского логического вывода в Python применяется библиотека ru-inference. Основное преимущество Python перед R — более высокая скорость выполнения скриптов.

Альтернативным решением для анализа данных в случае, когда быстродействия Python недостаточно, выступает Apache Spark — открытая масштабируемая кластерная вычислительная платформа, ориентированная на вычисления в оперативной памяти. При этом библиотека MLlib, которая в Spark реализует возможности машинного обучения, по количеству алгоритмов пока существенно уступает scikit-learn, но активно развивается.

В последние годы появились облачные платформы машинного обучения. Основным преимуществом таких систем является гибкая масштабируемость — выделение и высвобождение вычислительных ресурсов происходит мгновенно в соответствии с решаемыми задачами. Amazon Machine Learning реализует только базовые алгоритмы бинарной и мультиклассовой классификации, а также регрессии, Google Machine Learning Engine предоставляет возможность запуска моделей TensorFlow в облачной среде. Платформа Microsoft Azure Machine Learning Studio предоставляет собой мощное решение, позволяющее с помощью простого графического интерфейса строить модели машинного обучения с использованием множества стандартных алгоритмов классификации, регрессии, кластерного анализа и поиска аномалий, а также встраивать в эти модели собственный код на SQL, Python и R. Похожий инструмент имеется и у IBM — Watson Machine Learning. Однако несмотря на все преимущества облачных средств анализа данных, в банках они используются редко в связи

с опасениями по поводу безопасности передачи в облачные хранилища конфиденциальных данных о клиентах.

2.2. Интеллектуальные методы анализа устойчивости банков

С сентября 2013 г. Центральный Банк Российской Федерации проводит активную политику по проверке и отзыву лицензий у целого ряда российских банков, допускающих в своей деятельности различные нарушения. Так, с момента начала этого процесса лицензий на осуществление банковских операций лишились более чем 300 банков. Эта политика, как заявляют в самом Банке России, направлена на «очистку» рынка банковских услуг от таких игроков, которые не соответствуют требованиям закона, совершают сомнительные операции или имеют серьезные финансовые проблемы. Более того, целью Банка России является не только приведение банковского сектора в соответствующий нормативным требованиям вид, но и дальнейшее укрупнение и укрепление остающихся банков.

На фоне этого крайне актуальным становится совершенствование методологии анализа устойчивости банков, в том числе развитие применения машинного обучения для выявления у банков трудностей на ранних стадиях их формирования и развития, анализа реального текущего финансового положения банка и рейтинга его устойчивости.

Основной фокус исследования, представленного в данном разделе, заключается в оценке вероятности отзыва лицензии у банков вследствие невыполнения им обязательных нормативов и дефицита необходимых собственных банковских средств на покрытие рисков, связанных с проведением активных операций. Для проведения исследования была выгружена соответствующая банковская статистика с сайта Банка России [65].

Для каждого банка были выгружены следующие показатели:

- норматив достаточности собственных средств (капитала) Н1.0;
- норматив достаточности базового капитала банка Н1.1;
- норматив достаточности основного капитала банка Н1.2;
- норматив мгновенной ликвидности банка Н2;
- норматив текущей ликвидности банка Н3;
- норматив долгосрочной ликвидности банка Н4;

Интеллектуальные технологии в кредитном скоринге

- норматив максимального размера крупных кредитных рисков Н7;
- норматив максимального размера кредитов, банковских гарантий и поручительств, предоставленных банком своим участникам (акционерам) Н9.1;
- норматив совокупной величины риска по инсайдерам банка Н10.1;
- норматив использования собственных средств (капитала) банка для приобретения акций (долей) других юридических лиц Н12;
- норматив соотношения размера ипотечного покрытия и объема эмиссии облигаций с ипотечным покрытием Н18;
- собственный капитал банков;
- кредиты, выданные юридическим лицам – резидентам;
- кредиты, выданные юридическим лицам – нерезидентам;
- кредиты, выданные юридическим лицам – государственным компаниям;
- кредиты, выданные юридическим лицам – просроченные;
- кредиты, выданные индивидуальным предпринимателям;
- кредиты, выданные физическим лицам;
- текущие средства;
- срочные средства юридических лиц;
- срочные средства физических лиц.

При этом информация по выданным кредитам, текущим и срочным средствам была агрегирована по счетам бухгалтерского баланса, так как в готовом виде эти показатели на сайте Банка России не приводятся.

Информация по названным показателям была собрана в разбивке по месяцам (по состоянию на первую дату месяца, следующего за отчетным) за период с 01.11.2016 по 01.10.2017 г. Представлять выборку в виде квартальных значений нецелесообразно по причине недостаточности объема выборки (если исследуемый период равен одному году) и по причине уменьшения вклада в модель значений за более отдаленный период (если увеличивать исследуемый период). В итоговой выборке оказались 593 кредитные организации, из них у 63 банков за указанный период была отозвана лицензия на осуществление банковских операций.

Для корректного обучения модели по банкам с отозванной лицензией статистика была восстановлена в два этапа:

Глава 2

- 1) данные по таким банкам были выгружены за период в один год до отзыва лицензии,
- 2) затем данные были «сдвинуты» в общей базе показателей по всем банкам так, чтобы последние имеющиеся до отзыва лицензии данные находились в одном ряду со значениями показателей других банков на 01.10.2017 г.

В качестве индикатора устойчивости банка была введена логическая переменная «Индекс», для которой определены два состояния:

- 0 – банк продолжает функционировать;
- 1 – банк находится в состоянии нестабильности, дефолта.

Дефолт регистрируется в том случае, если имеется приказ Банка России об отзыве лицензии на осуществление банковских операций.

Следовательно, итоговая база показателей состоит из 593 банков, 21 показателя (перечислены выше), собранных за 12 месяцев и результирующего столбца «Индекс», в котором содержится информация об отзыве у банка лицензии.

Продемонстрируем классификацию банков по подготовленным данным на примере применения алгоритма *ADABoost*. Схема машинного обучения представлена на рисунке 12.

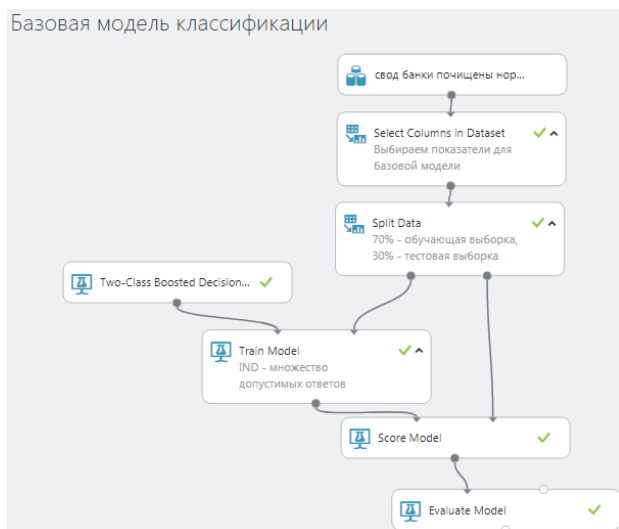


Рис. 12. Схема базовой модели

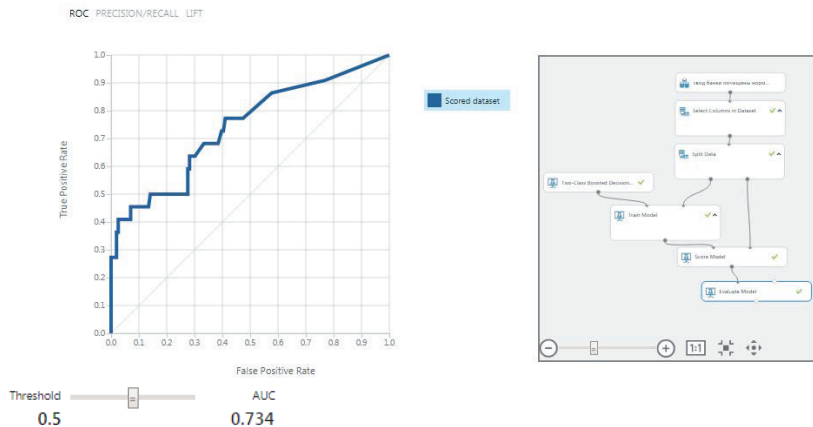


Рис. 13. ROC-кривая и показатели точности для базовой модели

На рисунке 13 представлена ROC-кривая и показатели точности для базовой модели. Так, показатель AUC для построенной модели равен 0,734.

Теперь построим схему машинного обучения по поиску аномалий таким образом, чтобы иметь возможность сравнить две модели: одна ветка на схеме определяет аномалии по методу One-Class Support Vector Machine, вторая ветка – по методу PCA-Based Anomaly Detection (рисунок 14). Для реализации алгоритмов поиска аномалий добавим на схему модуль Edit Metadata, с помощью которого пометим результирующий параметр – отозвана лицензия или нет. В свойствах Edit Metadata пометим используемый в выгрузке столбец IND как поле Label и присвоим этому столбцу аналогичное название. Как уже было описано, необходимо обучать модель на банках, у которых лицензия отозвана не была. Для этого в модуле Split Data установим свойство разделения выборки по значениям в поле Label: «Label».

Анализируя ROC-кривую и AUC (рисунок 15) для данных двух методов, можно сделать вывод, что метод PCA-Based Anomaly Detection является более точным по сравнению с One-Class Support Vector Machine.

Алгоритмы обнаружения аномалий, так же как и классификации, реализуем для базовой и для расширенной модели. В таблице 3 приводятся показатели качества по всем реализованным алгоритмам. Таким образом, выбираем алгоритм *ADABOOST*.

Глава 2

Поиск аномалий-отзыв лицензии у банка

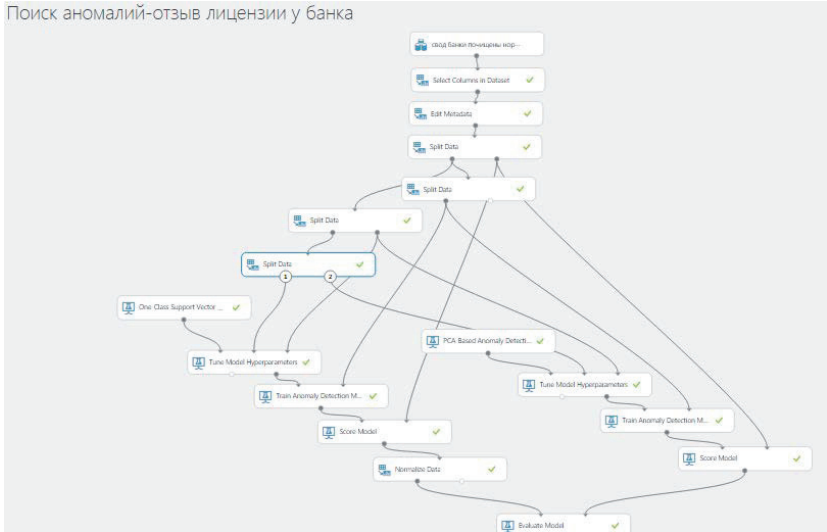


Рис. 14. Схема машинного обучения – поиск аномалий

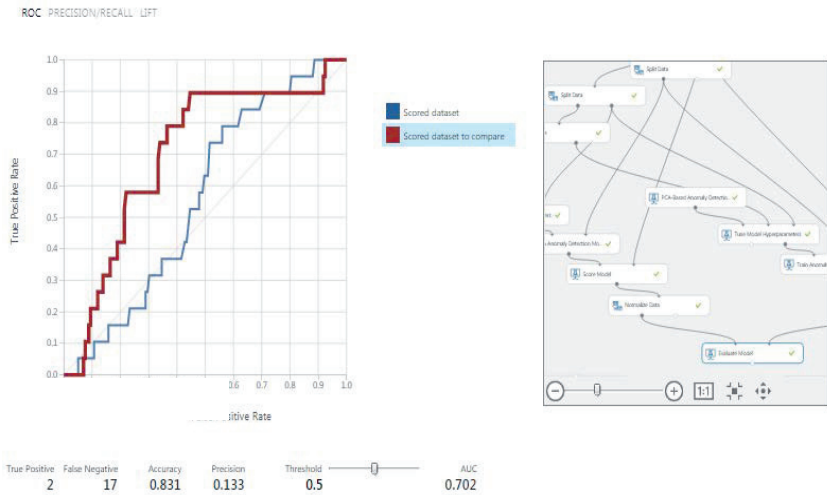


Рис. 15. Показатели качества моделей поиска аномалий

Таблица 3

Показатели AUC для алгоритмов анализа устойчивости банка

	Обнаружение аномалий		Классификация			
	One-Class Support Vector Machine	PCA-Based Anomaly Detection	Two-class logistic regression	Two-class decision forest	Adaboost	Two-class neural network
Базовая модель	0,560	0,702	0,565	0,625	0,734	0,472
Расширенная модель	0,593	0,736	0,578	0,689	0,805	0,563

Оценка вероятности отзыва лицензии

Рассмотрим расширенную модель, полученную с использованием метода *ADABOOST*. Наилучший показатель AUC, равный 0,805, имеет модель классификатора, построенная на основе значений обязательных нормативов с добавлением статистической информации о собственном капитале банка, его текущих средствах, просроченных кредитах, выданных юридическим лицам, а также кредитах, выданных индивидуальным предпринимателям и физическим лицам. Модель относит банк к категории кредитных организаций с отозванной лицензией на осуществление банковских операций в том случае, если вероятность отзыва лицензии для данного банка больше либо равна 0,5.

Для построенной модели анализа банков выгрузим предсказанные вероятности отзыва лицензии и проанализируем результат (таблица 4).

В столбце «Состояние банка на 01.04.18» банку вручную присвоена 1 в случае нестабильного положения – проходит финансовое оздоровление с целью недопущения банкротства, планируется реорганизация, демонстрируется отрицательный финансовый результат, банком получена негативная средняя оценка клиентов, и 0 в противном случае.

Как видно из таблицы 4, модель присвоила 35 банкам вероятность отзыва лицензии больше 0,5. При анализе деятельности этих банков было выявлено, что по состоянию на 01.04.2018 семь из попавших в таблицу банковских организаций функционируют стабильно.

Глава 2

Таблица 4

Оценка вероятности отзыва лицензии

№ лицензии	Название банка	Факт отзыва лицензии на 01.10.17	Оценка вероятности отзыва лицензии	Состояние банка на 01.04.18
2763	АКБ «Инвестторгбанк» (ПАО)	0	0,9870	1
3380	АО «Телекоммерц БАНК»	0	0,9440	1
2275	ПАО «Банк Уралсиб»	0	0,9060	1
970	ООО «КБ «Конфидэнс Банк»	0	0,9041	1
728	ООО «Татагропромбанк»	1	0,8868	1
2289	АО «Банк Русский Стандарт»	0	0,8835	1
55	ПАО «Московский акционерный Банк „Темпбанк“»	0	0,8464	1
2110	АКБ «Пересвет» (АО)	0	0,8320	1
3176	ПАО «Балтинвестбанк»	0	0,8156	1
1751	ПАО «Мособлбанк»	0	0,8147	0
2827	АО «Бинбанк Диджитал»	0	0,7996	1
3279	Банк «Траст» (ПАО)	0	0,7961	1
1132	АО «Социнвестбанк»	0	0,7882	1
1557	АО «ВУЗ-банк»	0	0,7882	1
1581	«Тимер Банк» (ПАО)	0	0,7799	1
3200	ООО КБ «Европейский Стандарт»	0	0,7771	1
3360	ПАО «Крайинвестбанк»	0	0,7766	1
554	АО КБ «Солидарность»	0	0,7755	1
558	АО Банк «Советский»	0	0,7709	1
2888	АО «Рост Банк»	0	0,7701	1
3252	АО «Газэнергобанк»	0	0,7685	1
128	ПАО «Балтийский Банк»	0	0,7683	0
3186	ООО КБ «Кредит Экспресс»	0	0,7651	1
3519	ООО «Икано Банк»	0	0,6489	0
2796	Банк «Вятич» (ПАО)	0	0,6373	0

Интеллектуальные технологии в кредитном скоринге

№ лицензии	Название банка	Факт отзыва лицензии на 01.10.17	Оценка вероятности отзыва лицензии	Состояние банка на 01.04.18
3287	Банк «ВБРР» (АО)	0	0,6055	0
812	ПАО «Уралтрансбанк»	0	0,5701	1
1414	ПАО «М2М Прайвет Банк»	1	0,5618	1
3396	АО «МБ Банк»	0	0,5567	0
2209	ПАО Банк «ФК Открытие»	0	0,5417	1
3275	КБ «Гаранти Банк – Москва» (АО)	1	0,5350	1
3499	Коммерческий банк «ВРБ» (ООО)	0	0,5117	0
2942	ООО КБ «НКБ»	1	0,5084	1
3250	ИКБ «ОЛМА-Банк» (ООО)	1	0,5073	1
2211	КБ «РЭБ» (АО)	1	0,5064	1

Кратко охарактеризуем деятельность банков, которые согласно модельным прогнозам имеют высокую вероятность отзыва лицензии:

- АКБ «Инвестторгбанк» (ПАО) – в настоящее время проходит санация банка, 100% акций находятся у банка-санатора Транскапиталбанк; до 2025 г. запланирована реорганизация банка в форме присоединения к Транскапиталбанку. Средняя оценка клиентов 2,59 из 5 (по данным агрегатора banki.ru);
- АО «Телекоммерц Банк» – лицензия на осуществление банковских операций отозвана 21.03.2018 г.;
- ПАО «Банк Уралсиб» – средняя оценка среди клиентов 1,8 из 5. 12.10.2017 г. у страховой группы «Уралсиб» была отозвана лицензия на осуществление ОСАГО;
- ООО КБ «Конфидэнс Банк» – по итогам 2016 г. и трех месяцев 2017 г. банк демонстрирует отрицательный финансовый результат;
- ООО «Татагропромбанк» – лицензия на осуществление банковских операций отозвана 05.04.2018 г.;
- АО «Банк Русский Стандарт» – средняя оценка клиентов 1,86 из 5, ранее банк испытывал серьезные трудности с выполнени-

- ем достаточности капитала всех уровней. Негативный международный рейтинг (Moody's);
- ПАО «Московский акционерный Банк „Темпбанк“» – лицензия на осуществление банковских операций отозвана 02.10.2017 г.;
 - АКБ «Пересвет» (АО) – с 21.10.2016 г. началась санация, санация проходила и в 2017 г. На 01.04.2017 г. банк относил 90% всех своих активов к категориям проблемных и безнадежных;
 - ПАО «Балтинвестбанк» – средняя оценка клиентов 1,82 из 5. В 2016 г. проходила санация. Изменение филиальной сети: на январь 2017 г. банк насчитывал 5 филиалов и 26 офисов, на декабрь 2017 г. – 1 офис;
 - АО «Бинбанк Диджитал» – входит в группу «Бинбанк». В сентябре 2017 г. Банк России ввел в Бинбанке временную администрацию в рамках мер по повышению финансовой устойчивости, в настоящее время проходит процесс санации;
 - Банк «Траст» (ПАО) – 15.03.2018 г. введена временная администрация в лице УК ФКБС. 20.03.2018 г. Банк России принял решение об уменьшении уставного капитала банка «Траст» до 1 руб. Сообщалось, что на базе банка «Траст» должен быть создан банк «плохих» долгов, в который будут направлены «плохие» активы Бинбанка, Банка «Открытие», Промсвязьбанка и др.;
 - АО «Социнвестбанк» – идет санация с сентября 2015 г., запланирована реорганизация в форме присоединения к банку-инвестору;
 - АО «ВУЗ-банк» – по итогам 2016 г. банк понес чистый убыток в размере 971,5 млн руб. Значительная часть выданных межбанковских кредитов просрочена. На 01.06.2017 г. капитал является отрицательным, и банк регулярно нарушает ряд нормативов. В срок до 01.10.2025 г. запланирована реорганизация банка;
 - «Тимер Банк» (ПАО) – средняя оценка клиентов 1,88 из 5, с 20.02.2017 по 22.02.2018 г. Банк России назначил в Тимер Банке временную администрацию в лице АСВ;
 - ООО «КБ «Европейский Стандарт» – лицензия на осуществление банковских операций отозвана 04.12.2017 г.;
 - ПАО «Крайинвестбанк» – по итогам 2017 г. получил убыток в размере 1 млрд руб. Средняя оценка клиентов 1,6 из 5;
 - АО КБ «Солидарность» – в банке два раза проводилась санация, запланирована реорганизация банка в форме присоединения к банку-инвестору;

Интеллектуальные технологии в кредитном скоринге

- АО Банк «Советский» – с октября 2015 г. ведется процесс санации, с февраля 2018 г. введена временная администрация в лице УК «ФКБС»;
- АО «Рост Банк» – с 15.03.2018 г. Банк России утвердил план, согласно которому он будет участвовать в осуществлении мер по финансовому оздоровлению «Рост Банка», назначена временная администрация в лице УК «ФКБС»;
- АО «Газэнергобанк» – средняя оценка клиентов 1,52 из 5. Банк находится в процессе санации, утратил собственный капитал – на 01.11.2017 г. собственный капитал был отрицательным (–414,8 млн руб.);
- ООО КБ «Кредит Экспресс» – лишился лицензии на осуществление банковских операций 15.03.2018 г.;
- ПАО «Уралтрансбанк» – по итогам 2017 г. банк понес убыток в размере 295,2 млн руб., за два месяца 2018 г. банк понес убыток в размере 203,7 млн руб.;
- ПАО Банк «ФК Открытие» – средняя оценка клиентов 1,8 из 5. 11.12.2017 г. Банк России стал владельцем свыше 99,9% акций ФК «Открытие». По итогам 2017 г. банк понес чистый убыток в размере 220,5 млрд руб.

Для оценки качества полученной модели вероятности отзыва лицензии рассчиталась доля правильных прогнозов:

$$Accuracy = \frac{P}{N} = \frac{28}{35} = 0,8$$

где P – количество банков, по которым по состоянию на 01.04.2018 г. отмечена нестабильная деятельность, а N – общее количество банков, отнесенных моделью к категории прекративших существование. Таким образом, можно сделать вывод, что модель вероятности отзыва лицензии дает довольно точный результат, и полученный рейтинг банковских организаций можно применять на практике.

Динамика вероятности отзыва лицензии

Используя построенную модель оценки вероятности отзыва лицензии и новую выгрузку по банкам, проанализируем динамику вероятности прекращения банком своей деятельности ввиду отзыва лицензии. Для этого необходимо сначала подготовить выгрузку данных за период с 01.11.2017 по 01.04.2018 г. с сайта Банка России. Для каждого

банка из таблицы 4, у которого по состоянию на 01.11.2017 не была отозвана лицензия, были выгружены следующие показатели:

- значения обязательных нормативов банка;
- собственный капитал банков;
- кредиты, выданные юридическим лицам – просроченные;
- кредиты, выданные индивидуальным предпринимателям;
- кредиты, выданные физическим лицам;
- текущие средства.

Если у банка за исследуемый период лицензия была отозвана, то по таким банкам бралась статистика за 6 мес. до отзыва лицензии, а затем эти данные были «сдвинуты» в общей базе показателей по всем банкам так, чтобы последние имеющиеся до отзыва лицензии данные находились в одном ряду со значениями показателей других банков на 01.04.2018 г.

Полученная выборка используется в качестве тестовой для построенной модели. Выгрузим предсказанные вероятности отзыва лицензии и сравним результат с расчетами, проведенными по выгрузке за период с 01.11.2016 по 01.10.2017 г. (таблица 5).

Сравнивая вероятности отзыва лицензии у банка по состоянию на 01.10.2017 г. и 01.04.2018 г., можно сделать вывод, что практически по всем банкам, чья деятельность охарактеризована как нестабильная (таблица 5), за исключением только Тимер Банка и Солидарности, вероятность отзыва лицензии возросла. Вместе с тем снизилась вероятность отзыва лицензии у банков, чья деятельность охарактеризована как стабильная, за исключением ПАО «Мособлбанк», ПАО «Балтийский Банк», Банка «ВБРР» (АО).

Таким образом, динамика вероятности отзыва лицензии подтверждает хорошую прогностическую способность модели качества модели, построенной с помощью метода *ADABOOST*, и возможность использования рейтинга банков, полученного с помощью этой модели, на практике.

2.3. Интеллектуальные методы выявления кредитных организаций, вовлеченных в отмывание доходов

После принятия Федерального закона от 23.07.2013 г. № 251-ФЗ «О внесении изменений в отдельные законодательные акты Российской Федерации в связи с передачей Центральному банку Российской Феде-

Таблица 5
Динамика вероятности отзыва лицензии

№ лицензии	Название банка	Факт отзыва лицензии на 01.04.18	Оценка вероятности отзыва лицензии на 01.10.17	Оценка вероятности отзыва лицензии на 01.04.18
2763	АКБ «Инвестторгбанк» (ПАО)	0	0,9870	0,9816
3380	АО «Телекоммерц БАНК»	1	0,9440	0,9608
2275	ПАО «Банк Уралсиб»	0	0,9060	0,9130
970	ООО КБ «Конфидэнс Банк»	0	0,9041	0,9485
728	ООО «Татагропромбанк»	0	0,8868	0,9380
2289	АО «Банк Русский Стандарт»	0	0,8835	0,9142
2110	АКБ «Пересвет» (АО)	0	0,8320	0,8473
3176	ПАО «Балтинвестбанк»	0	0,8156	0,8904
1751	ПАО «Мособлбанк»	0	0,8147	0,8240
2827	АО «Бинбанк Диджитал»	0	0,7996	0,8016
3279	Банк «ТрасТ» (ПАО)	0	0,7961	0,8199
1132	АО «Социнвестбанк»	0	0,7882	0,7905
1557	АО «ВУЗ-банк»	0	0,7882	0,7964
1581	«Тимер Банк» (ПАО)	0	0,7799	0,7672
3200	ООО КБ «Европейский Стандарт»	1	0,7771	0,8214
3360	ПАО «Крайинвестбанк»	0	0,7766	0,7917
554	АО КБ «Солидарность»	0	0,7755	0,7689
558	АО Банк «Советский»	0	0,7709	0,7856
2888	АО «Рост Банк»	0	0,7701	0,8060
3252	АО «Газэнергобанк»	0	0,7685	0,7927
128	ПАО «Балтийский Банк»	0	0,7683	0,7720
3186	ООО КБ «Кредит Экспресс»	1	0,7651	0,7951
3519	ООО «Икано Банк»	0	0,6489	0,6105
2796	Банк «Вятич» (ПАО)	0	0,6373	0,6289
3287	Банк «ВБРР» (АО)	0	0,6055	0,6182
812	ПАО «Уралтрансбанк»	0	0,5701	0,6450
3396	АО «МБ Банк»	0	0,5567	0,5307
2209	ПАО Банк «ФК Открытие»	0	0,5417	0,5463
3499	Коммерческий банк «ВРБ» (ООО)	0	0,5117	0,4897

рации полномочий по регулированию, контролю и надзору в сфере финансовых рынков» Банку России был придан статус мегарегулятора финансового рынка страны с 01.09.2013 г. Одной из задач, стоящих перед Банком России, является укрепление и оздоровление банковского сектора страны, предотвращение формирования и накопления системных рисков в финансовой сфере, появления новых «пузырей» и перегрева в отдельных сегментах рынка.

Субъекты первичного финансового мониторинга в Российской Федерации определены в ст. 5 Федерального закона «О противодействии легализации (отмыванию) доходов, полученных преступным путем, и финансированию терроризма» № 115-ФЗ от 07.08.2001 г. К ним относятся организации, осуществляющие операции с денежными средствами и иным имуществом, а также иные лица, чьи права и обязанности в сфере противодействия легализации преступных доходов и финансированию терроризма установлены ст. 7 данного закона.

Банки обслуживают в качестве своих клиентов в том числе финансовые и нефинансовые учреждения, представляя собой особую категорию финансовых учреждений. Кредитные организации осуществляют почти все виды финансовой деятельности. В связи с этим, кредитные организации занимают особое место в системе финансового мониторинга.

Однако вовлеченность кредитных организаций в противоправную деятельность, создание и организация работы схем по отмыванию денежных средств, оказанию теневого финансового услуг с использованием банковской инфраструктуры, является серьезной проблемой, создающей высокие риски финансовой безопасности государства.

По мнению экспертов, в кризис может увеличиться количество кредитных организаций, занимающихся выводом капитала из России, нарушающих положения Федерального закона от 07.08.2001 г. № 115-ФЗ «О противодействии легализации (отмыванию) доходов, полученных преступным путем, и финансированию терроризма».

Отзыв лицензии банка является крайней мерой воздействия, так как не только создает дополнительную нагрузку на бюджет страны, но и может способствовать увеличению социальной напряженности. Таким образом, крайне важно, иметь не только текущую оценку состояния кредитной организации, но и его прогнозных значений. Получение прогноза финансового состояния банка необходимо для своевременного принятия соответствующих управленческих решений, перераспределения человеческих и иных ресурсов.

В настоящее время для диагностики финансового состояния кредитной организации требуется изучение, систематизация и обработка большого объема информации, основную часть которой составляет официальная банковская отчетность. Однако полученные результаты не всегда в полной мере доступно и наглядно могут охарактеризовать финансовую безопасность банка, по причине громоздкости и порой неочевидности алгоритмов расчета показателей. Предотвратить такие негативные ситуации и увеличить скорость реакции на изменения внешней среды, что, в свою очередь, определяет успех в управлении, позволяет использование специального математического инструментария. По причине своей малой (по сравнению с другими методами, способами и механизмами) материалоемкости он является основным элементом системы поддержки принятия решений. Кроме того, специальный математический инструментарий дает возможность подготовить и провести многочисленные эксперименты сравнительно быстро.

Аналитики Банка России, рейтинговые агентства при анализе деятельности кредитных организаций руководствуются целым рядом признаков и критериев. Характерно, что подобные оценки могут носить субъективный, политизированный характер. Изучение подходов методов свертки и скаляризации векторов позволяют сделать вывод о перспективности метода главных компонент факторного анализа.

Применение метода главных компонент факторного анализа позволило синтезировать интегральные оценки кредитных организаций, указывающие на ухудшение финансового состояния банка, как правило, предшествующее отзыву лицензии.

Базельский комитет по банковскому надзору выделяет пул рисков для кредитных организаций, которые не осуществляют должной проверки своих клиентов в целях противодействия легализации преступных доходов:

- ущерб репутации банка;
- ухудшение финансового положения банка;
- правовые последствия;
- концентрация невозвратных кредитов [66].

Перечисленные риски тесно связаны друг с другом, а их наложение может привести к отрицательным последствиям.

Риск потери репутации заключается в том, что добросовестные клиенты, крупный бизнес не захотят продолжать деловые отношения с банком, подозреваемым или обвиняемым в отмывании денег или финансировании терроризма, поддерживающим связи с преступным сообществом.

Риск ухудшения финансового положения банка связан с возможностью возникновения прямых или косвенных убытков, по причине нарушения закономерностей устройства внутренних процессов, неэффективностью работы систем и персонала или вмешательством внешних факторов. Подозрения клиентов кредитной организации в несостоятельности менеджмента операционными рисками также может серьезно повредить деловой репутации банка.

Негативные правовые последствия могут выражаться в обращении судебных взысканий к кредитной организации, наложении штрафов, запрете осуществлять операционную деятельность, что приведет к ухудшению финансового положения или даже прекращению ее деятельности.

Концентрация кредитов может привести к убыткам от отдельных групп заемщиков или группы связанных заемщиков при указании недостоверной информации о соответствующих клиентах банка. Концентрация «плохих» кредитов может привести к снижению ликвидности банка в случае одновременного изъятия средств крупными вкладчиками, что характерно при отмывании денег.

Другие финансовые организации, такие как страховые компании, инвестиционные фонды, операторы рынка ценных бумаг, также подвержены перечисленным рискам, связанным с легализацией преступных доходов.

На основании статьи 57 Федерального закона «О Центральном банке Российской Федерации» от 10.07.02 № 86-ФЗ Банком России установлен ряд отчетных форм для кредитных организаций, для предоставления сведений на регулярной основе [67].

Применение кластерного анализа для анализа состояния кредитных организаций

Для целей анализа банковской финансовой отчетности отчетная форма № 101 является самой информативной. Форма 101 содержит информацию об объемах и структурных показателях требований и обязательств банка, в ней отражаются результаты прове-

Интеллектуальные технологии в кредитном скоринге

Банк	Reg.№	БИК	ФО	Сотня по активам	Включен в перечень системо-значимых КО	Стратеги	Количество клиентов-участников операций месяца
АЛЬТЕРНАТИВА	3452	044525115	ЦФО	7	0	0	3
АЛЬФА-БАНК	1326	044525593	ЦФО	1	1	1	35457
АМЕРИКЭН ЭКСПРЕСС БАНК	3460	044525717	ЦФО	5	0	0	1
АНЕЛИК РУ	3443	044583616	ЦФО	7	0	0	12
АНКОР БАНК	889	049209778	ПФО	3	0	0	17
АО БАНК ЗЕНИТ СОЧИ	232	040096717	ЮФО	4	0	0	126
АО БАНК ИННОВАЦИИ И РАЗВИТИЯ	2547	044583104	ЦФО	4	0	0	173
А БАНК МПБ	2646	044525283	ЦФО	5	0	0	61
АО БАТЛБАНК	1765	041203715	ЮФО	6	0	0	73
АО МС БАНК РУС	2789	044525450	ЦФО	3	0	0	5
АО РОСЭКСИМБАНК	2790	044525192	ЦФО	1	0	1	11
АО ЮНИКРЕДИТ БАНК	1	044525545	ЦФО	1	1	1	1873
АПБАБАНК	2404	044525238	ЦФО	5	0	0	54
АРБ-ИНКАСС	3353	044525170	ЦФО	7	0	0	0
АРЕСБАНК	2914	044583551	ЦФО	2	0	0	69
АРСЕНАЛ	1281	042204757	ЦФО	6	0	0	42
АСПЕКТ	1868	044583731	ЦФО	3	0	0	15
АССОЦИАЦИЯ	3000	044525585	ПФО	6	0	0	51
АТБ	608	044525401	ЦФО	5	0	0	77
БАЙКАЛБАНК	732	042282751	СФО	3	0	0	128
БАЙКАЛИНВЕСТБАНК	2776	044599244	СФО	2	0	0	263
БАЙКАЛКРЕДОВАНК	2632	048142736	СФО	3	0	0	356
БАЛАКОВО-БАНК	1067	042520706	ПФО	4	0	0	75
БАЛТИЙСКИЙ БАНК	2990	042520872	СЗФО	6	0	1	23
БАЛТИНВЕСТБАНК	444	046359728	СЗФО	7	0	0	15

Рис. 16. Фрагмент исходных данных

денных банковских операций за отчетный период, публикуется ежемесячно. На рисунке 16 представлен фрагмент исходных данных.

Банковская сфера подлежит мониторингу со стороны надзорных органов государственной власти, при этом количество банков в России очень велико, в связи с чем, требуется применение риск-ориентированного подхода для выявления наиболее подверженных злоупотреблениям субъектов. Дистанционный мониторинг, анализ состояния банков позволит формировать «группы риска», т. е. банки, потенциально подверженные негативным проявлениям. Так называемые «системы раннего предупреждения» (*Early Warning Systems, EWS*) – или дистанционные методы мониторинга – не могут дать однозначного ответа является ли тот или иной банк вовлеченным в противоправную деятельность или нет. В тоже время, они су-

щественно сокращают расходы органов надзора, которые в первую очередь будут проверять банки, оказавшиеся в группе риска по результатам дистанционного обследования. Это приведет к повышению эффективности банковского надзора, повышению стабильности кредитно-финансовой системы страны.

Выделяют следующие подходы к предиктивному анализу кредитных организаций. Во-первых, прогнозирование надежности банка на основе анализа исторических данных других аналогичных банков, у которых была отозвана лицензия. В этом случае могут быть применены модели бинарного выбора. Во-вторых, использование рейтингов кредитных организаций, составляемых рейтинговыми агентствами. В этом случае прогнозная модель будет отражать мнение экспертов рейтингового агентства.

В-третьих, прогнозирование надежности банка на основе анализа процентных ставок, например, процентных ставок по депозитам физических лиц. Вкладчики ожидают от банков с рискованной финансовой политикой более высокие процентные ставки, которые, в свою очередь, могут служить сигналом об избыточном риске финансовой политики банка.

И в-четвертых, прогнозирование надежности банка на основе оценок его технической эффективности. Ряд исследований в области эконометрики связывают техническую эффективность (или точнее, эффективность по издержкам) кредитной организации с ее надежностью.

При решении задачи рейтингования кредитных организаций приходится сравнивать объекты, заданные упорядоченными наборами признаков (векторами), для которых отношение «больше—меньше» не определено. Эта неопределенность носит фундаментальный характер и для преодоления требует изыскания и синтеза адекватных интегральных скалярных характеристик объектов векторной природы. Одно из перспективных направлений разрешения этой проблемы может быть реализовано на основе метода главных компонент факторного анализа.

Каждый объект финансовой системы характеризуется большим количеством стохастически связанных признаков, поэтому для решения достаточно сложных задач классификации объектов применяют кластерный анализ.

Кластерный анализ позволяет разбивать множество исследуемых объектов и их признаков на однородные группы, или класте-

ры. При этом решается задача классификации данных, происходит выявление в ней соответствующей структуры.

Информации о классифицируемых объектах представим в виде матрицы типа «объект–свойство»:

$$X = \begin{matrix} & \begin{matrix} x_{11} & x_{12} & \dots & x_{1k} \\ x_{21} & x_{22} & \dots & x_{2k} \\ \dots & \dots & \dots & \dots \\ x_{n1} & x_{n2} & \dots & x_{nk} \end{matrix} \\ \begin{matrix} X \\ n^*k \end{matrix} & \end{matrix}$$

где x_{ij} — значение j -го признака на i -м объекте наблюдения.

В качестве метрики выберем евклидово расстояние между объектами, которое рассчитывается по формуле:

$$d_E(O_i, O_j) = \sqrt{\sum_{l=1}^k (x_{il} - x_{jl})^2}.$$

Визуальное или экспертное обследование не всегда дают положительный результат, особенно при большом объеме исследуемых данных, на помощь приходят методы кластеризации, которые необходимы для обнаружения структуры в данных.

Применим метод K -средних. Чтобы найти количество кластеров, для первоначального разбиения имеющейся совокупности данных, применим метод Уорда и получим вертикальную дендрограмму (рисунок 17).

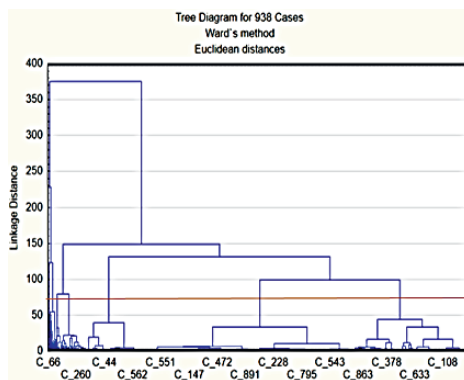


Рис. 17. Дендрограмма по методу Уорда

Глава 2

Пусть дан вектор $X=(X_1, X_2, \dots, X_k)^T$. Так как исходные признаки отличаются масштабом измерений, то центрируем и нормируем их.

Требуется снизить размерность признакового пространства до m признаков ($m < k$).

Линейная модель факторного анализа имеет вид:

$$x_i = a_{i1}f_1 + a_{i2}f_2 + \dots + a_{im}f_m + d_i v_i, \quad i = \overline{1, k},$$

где f_r – r -й общий фактор, $r = \overline{1, k}$; v_i – i -й характерный фактор; a_{ir} – весовой коэффициент, или коэффициент весовой нагрузки i -го признака на r -м общем факторе; d_i – весовой коэффициент i -го признака на i -м характерном факторе.

При применении метода главных компонент было выделено два фактора, это позволяет представить кластеры банков графически на декартовой плоскости.

При анализе данных при помощи метода главных компонент важнейшее значение имеет интерпретация полученных результатов.

На основе корреляционной матрицы признаков, характеризующих кредитные организации, получены дисперсии первых пяти главных компонент, а также коэффициенты корреляции признаков банков с главными компонентами – внутренними факторами.

В ходе исследования признакового пространства анализируемых многомерных объектов были выбраны наиболее информативные показатели с минимальной внутригрупповой корреляцией и их производные:

- активы,
- прирост активов по сравнению с предыдущим периодом,
- капитал,
- прирост капитала по сравнению с предыдущим периодом,
- кредиты предприятиям и организациям,
- потребительские кредиты,
- ценные бумаги,
- оборот средств в банкоматах,
- вклады физических лиц,
- прирост вкладов физических лиц по сравнению с предыдущим периодом,
- средства предприятий и организаций,
- прирост средств предприятий и организаций по сравнению с предыдущим периодом.

В таблице 6 представлена корреляционная матрица между признаками, которая является всегда симметричной. Диагональные элементы этой матрицы – единицы. Она может использоваться для оценки взаимосвязи между признаками и объектами. Как видно из таблицы 6, некоторые показатели коррелируют между собой. Сильно коррелирующие показатели обозначены красным цветом. Такие показатели, как «Капитал», «Кредиты предприятиям и организациям», «Средства предприятий и организаций» имеют высокие коэффициенты корреляции, свыше 0,97. Такая взаимосвязь представляется весьма очевидной с точки зрения здравого смысла. Действительно, если кредитная организация активно сотрудничает с юридическими лицами в части привлечения заемных средств и процентного кредитования, то капитал растет вследствие наращивания денежной массы.

Таблица 6
Корреляционная матрица признаков

Показатель	1	2	3	4	5	6	7	8	9	10	11	12
1	1											
2	0,0119	1										
3	-0,5658	-0,0008	1									
4	0,7705	0,0304	-0,3922	1								
5	-0,4573	0,0013	0,9840	-0,3313	1							
6	-0,5375	0,0152	0,8490	-0,2714	0,8456	1						
7	-0,0233	-0,0934	0,0334	-0,0335	0,0305	0,0067	1					
8	-0,0110	-0,0032	0,0025	-0,0328	0,0106	0,0313	0,3001	1				
9	-0,0011	-0,0648	0,0027	-0,0130	0,0003	0,0262	0,0999	0,4773	1			
10	0,0010	-0,0003	-0,0021	0,0014	-0,0018	0,0003	-0,0023	0,0088	0,0095	1		
11	-0,4687	0,0005	0,9739	-0,3199	0,9726	0,7944	0,0350	-0,0061	-0,0039	-0,0025	1	
12	-0,0077	-0,1004	0,0028	-0,0242	0,0032	0,0257	0,5716	0,7114	0,8236	0,0086	-0,0069	1

Корреляция признаков указывает на то, что они не ортогональны. Хотя это иногда просматривается интуитивно, в данном случае на место субъективных оценок приходят точные объективные количественные оценки, которыми являются коэффициенты корреляции. Далее применим метод главных компонент факторного анализа.

На основе корреляционной матрицы признаков были получены дисперсии первых пяти главных компонент, а также коэффициенты корреляции признаков субъектов финансово-кредитного сектора с главными компонентами (внутренними факторами).

По диаграмме, представленной на рисунке 18, видно, что первые три главных компоненты объясняют 83% общей дисперсии исходных признаков.

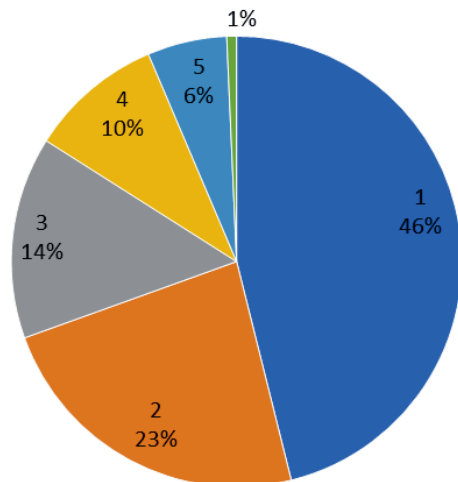


Рис. 18. Характеристика суммарного вклада главных компонент в общую дисперсию

Анализ весов и нагрузок найденных главных компонент, а также консультации с экспертами в области финансового мониторинга дают основание полагать, что ухудшение финансового состояния кредитной организации, предшествующее отзыву лицензии, отражает вторая главная компонента. Первая главная компонента отражает среднее значение входящих в расчет признаков, показывает величину банка и масштаб его инвестиционно-финансовой деятельности.

По оси абсцисс отложим значения второй главной компоненты, а по оси ординат – первой главной компоненты для каждой кредитной организации, банки окажутся размещенными в новых осях координат.

На рисунке 19 группа банков, расположенных в первом и четвертом квадрантах, представляет наибольший интерес. Это крупные (I) и небольшие (IV) кредитные организации, имеющие высокий риск отзыва лицензий.

Распределение количества кредитных организаций по второй главной компоненте близко к нормальному (рисунок 20).

В ходе исследования была построена диаграмма расстояния от дискретных значений факторов ликвидированных Банком России кредитных организаций до их математического ожидания. Рас-

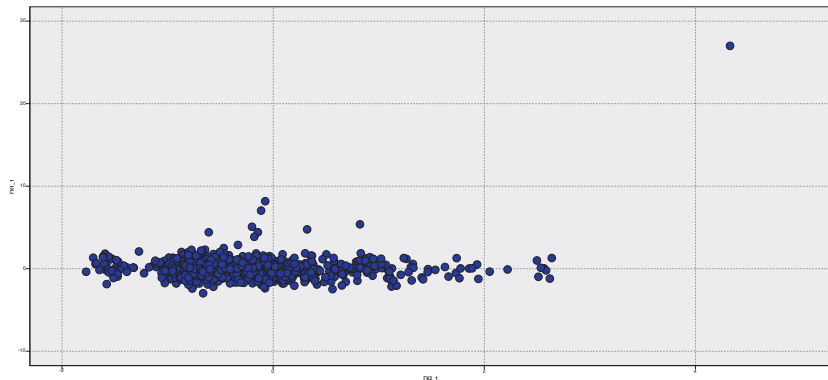


Рис. 19. Распределение банков по первой и второй главным компонентам

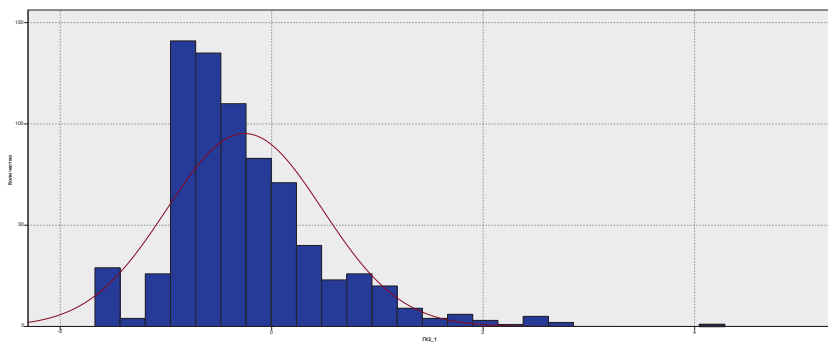


Рис. 20. Распределение количества банков по второй главной компоненте

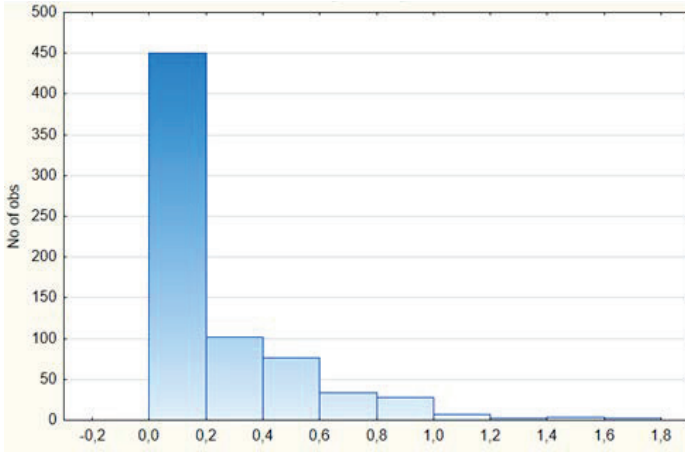
сма­три­вае­мое рас­сто­я­ние яв­ля­ет­ся срав­ни­тель­но ма­лым от­но­ситель­но ана­логич­ного рас­сто­я­ния до зна­чений фак­то­ров бла­го­на­деж­ных бан­ков (со­от­вет­ст­вую­щие диа­грам­мы при­ве­де­ны на ри­сунке 21).

На ри­сунке 21 по­ка­за­но, что боль­шая часть бла­го­на­деж­ных бан­ков ле­жит от ма­те­ма­ти­че­ского ожи­да­ния лик­види­ро­ван­ных на рас­сто­я­нии в диа­па­зоне от 0,0 до 0,25; что же ка­са­ет­ся рас­сто­я­ния от лик­види­ро­ван­ных кре­дит­ных ор­га­ни­за­ций до их ма­те­ма­ти­че­ского ожи­да­ния, то в боль­шин­стве слу­чаев оно ле­жит в диа­па­зоне от 0,0 до 0,15.

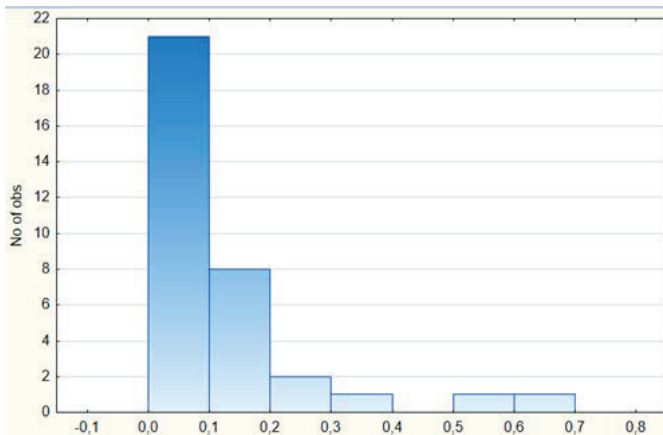
Уста­нов­лен­ный факт сви­де­тель­ст­вует о том, что дей­ст­вую­щие кре­дит­ные ор­га­ни­за­ции, на­хо­дя­щие­ся на от­но­ситель­но ма­лом ев-

Глава 2

клиндовом расстоянии от математического ожидания кредитных организаций, ликвидированных заданный момент времени, несут потенциальные риски вовлечения в противоправную деятельность, отмывание денег и финансирование терроризма.



а) «плохие» банки



б) «хорошие» банки

Рис. 21. Диаграммы расстояний

Глава 3

ИНТЕЛЛЕКТУАЛЬНЫЕ ТЕХНОЛОГИИ ПРОГНОЗИРОВАНИЯ СОСТОЯНИЙ ФИНАНСОВЫХ РЫНКОВ

3.1. Традиционные модели машинного обучения, используемые для прогнозирования рыночных трендов

Прогнозирование точных значений цен финансовых инструментов большинством исследователей и практиков представляется невозможным. В работе [68] показано, что корреляции между временными рядами цен не являются ни экономически, ни статистически значимыми. В большинстве работ, связанных с прогнозированием цен, обсуждается *гипотеза эффективного рынка (Efficient Market Hypothesis, EMH)*, предложенная в работе [69]. Согласно этой гипотезе, вся существенная информация немедленно и в полной мере отражается на рыночных ценах активов. На эффективном рынке мгновенно происходит коррекция цен, которые становятся справедливыми, не оставляя участникам рынка арбитражных возможностей. При этом участники рынка однородны в своих установках и однородно интерпретируют поступающую информацию, мгновенно корректируя свои решения при поступлении новой информации.

Различают три формы эффективности рынка:

- слабую форму эффективности, при которой стоимость рыночного актива полностью отражает прошлую информацию, касающуюся данного актива (общедоступную в данный момент времени информацию о прошлых значениях цен и объемов торгов активом);

Глава 3

- среднюю форму эффективности, когда стоимость рыночного актива полностью отражает не только прошлую, но и публичную информацию (текущую информацию, которая становится общедоступной в настоящий момент времени, предоставленная в финансовых новостях, отчетах компаний, выступлениях аналитиков и т. п.);
- сильную форму эффективности, если стоимость рыночного актива полностью отражает всю информацию — прошлую, публичную и внутреннюю (инсайдерскую, известную узкому кругу лиц в силу служебного положения или иных обстоятельств).

Математически гипотеза эффективного рынка означает, что соответствующие случайные процессы, определяющие поведение цен, являются марковскими, т. е. их будущие значения не зависят от значений в предыдущие моменты времени, и поэтому будущие цены активов невозможно предсказать, основываясь только на прошлых значениях цен.

Многие трейдеры, брокеры, финансовые аналитики, индивидуальные инвесторы и другие участники рынка убеждены в том, что они могут (интуитивно либо с помощью тех или иных методов) прогнозировать рыночные тенденции и с помощью этого извлекать прибыль. Помимо интуитивных прогнозов, для прогнозирования тенденций разработано множество методов и моделей.

Традиционные подходы к прогнозированию рыночных цен можно в целом разделить на два типа:

- фундаментальный анализ, основанный на оценке «внутренней» стоимости компании-эмитента с использованием ее финансовых показателей, а также макроэкономических индикаторов [70, 71, 72, 73, 74, 75, 76];
- технический анализ, представляющий собой набор инструментов, полностью игнорирующих гипотезу эффективного рынка, предполагающих, что история повторяется и поведение рынка определяется прошлыми изменениями цен в аналогичных обстоятельствах [77, 78, 79, 80, 81].

Как правило, фундаментальный анализ применяют при горизонте прогнозирования, не меньшем одного года, при более коротких интервалах прогноза применяют технический анализ.

В техническом анализе в последние годы используют индикаторы и осцилляторы не со стандартными (рекомендованными) параметрами, а с регулярно подстраиваемыми по скользящей выборке на основе вероятности ошибки прогноза $P_a = N_a/N$, где N_a — количество верных сигналов тренда, N — общее количество сигналов тренда.

Общее отношение к техническому анализу сейчас, по-видимому, иллюстрируется работами [82, 83, 84].

В работе [82] к данным о динамике шести валют, котируемых в долларах США, за период с 1994 по 2014 г. применялись 7650 торговых правил технического анализа: авторы замечают, что часть из них на исторических данных показали прибыль, часть нет, и при этом некоторые правила срабатывают чаще, чем другие.

Авторы статьи [83] исследовали прогностическую способность 93 популярных индикаторов технического анализа для 54-летней выборки ценовых данных об акциях из S&P 500, и ни один из индикаторов не показал значимой прогностической способности.

В статье [84] к реальному набору 198 бразильских акций применялись методы тройного экрана, простых и экспоненциальных скользящих средних в сочетании с тактикой стоп-лоссов, при этом никакой значимой прогностической способности выявлено не было.

В последние годы появилось достаточно много исследований, основанных на методологии интеллектуального анализа данных и машинного обучения, в том числе использующих аппарат деревьев и лесов решений, нейронных сетей, нечеткой логики, генетических алгоритмов и т. д. [85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100, 101, 102, 103, 104, 105, 106, 107, 108, 109, 110, 111].

Все эти работы можно разделить на четыре группы:

- модели, прогнозирующие значения цены, например [87, 92, 95, 98];
- модели, прогнозирующие направление изменения цены: завтрашняя цена увеличится, уменьшится или не изменится [97, 100];
- модели, определяющие фазу рынка; при этом, как правило, прогнозируется направление тренда: вверх или вниз, боковики и «серые зоны» не рассматриваются [90, 91];
- модели, прогнозирующие прибыль инвестора [99, 101].

В работе [85] предлагается использовать индикатор технического анализа, основанный на нечеткой логике и использовании субъек-

тивной экспертной информации об уровне риска. Однако качество модели оценивается не по реальным рыночным данным, а с помощью многоалгоритмной имитационной модели.

В работе [86] рассматривается регрессионная модель волатильности российского валютного рынка (RUR/USD), используется декомпозиция волатильности на компоненты, характеризующие фрактальную структуру финансового ряда, с помощью регрессионного анализа подтверждается квазицикличность одной из компонент. При обсуждении возможности прогноза динамики волатильности, в том числе прогноза перехода рынка в нестабильное состояние, показано, что прогнозирование на интервал длиной в полтора характерных масштаба фактически не отличается от «бросания монеты» — память полностью пропадает, но при предсказании на интервал длиной не более 4–8 недель (при характерном масштабе в 32 недели) прогноз с помощью предложенной модели оказывается значимо лучше мартингального, хотя его достоверность и невелика (доля совпадений спрогнозированных и фактических трендов оказалась на уровне 0,500–0,515).

- о Машинное обучение — альтернатива классическим статистическим методам, которая показывает интересные результаты при решении задач, в которых определения и свойства изучаемых объектов плохо формализуемы:
 - распознавание образов и речи;
 - беспилотное вождение;
 - медицинская диагностика;
 - обнаружение мошенничества;
 - кросс-продажи;
 - кредитный скоринг;
 - прогнозирование оттока клиентов;
 - анализ финансовых рынков и др.

Суть методологии машинного обучения состоит в том, что на основании относительного больших наборов исходных данных строится модель, которая затем используется для прогнозирования.

Прогностическая способность (понимаемая как доля сбывшихся прогнозов) моделей, предсказывающих направление изменения цены с помощью машинного обучения, в моделях из работ [102, 103, 104, 105, 106] находится соответственно на уровне 56, 58, 62, 68 и 86%.

В моделях, прогнозирующих прибыль инвестора, известны только частные случаи. Так, например, в работе [101] для двух тайваньских акций за один год показана 54%-ная и 128%-ная доходность. Однако качество модели на больших объемах данных о ценах разных инструментов в разное время не проверялось.

В работах, посвященных определению фазы рынка, также описываются частные случаи прибыльных стратегий.

Достаточно полный, хотя и не совсем новый (2009 г.) обзор методов машинного обучения в применении к прогнозированию динамики цен приведен в работе [112].

В целом, модели, основанные на машинном обучении, показывают себя эффективнее, чем традиционные, в частности, чем традиционные модели технического анализа и авторегрессии.

Среди известных моделей прогнозирования движения цен и состояния рынка, основанных на машинном обучении, наилучшие результаты показывают следующие классы моделей:

- усиленные деревья решений (с усилением *ADABOOST*, *XGBOOST*);
- леса решений;
- многослойные однонаправленные нейронные сети, основанные на алгоритме обратного распространения (*MLP BPN*);
- комбинированные модели (нейронные сети с нечеткой логикой и генетическими алгоритмами, комбинации нейронных сетей и деревьев решений и т. п.);
- многошаговые модели, в которых, например, на первом шаге отбираются или генерируются релевантные факторные признаки, а на последующих шагах строится прогноз на признаках, полученных на предыдущих шагах.

В работе [113] описаны результаты применения усиленных деревьев к решению задачи прогнозирования состояния рынка. На основании данных о ценовой динамике 100 акций из индекса S&P 500 за период с 2007 по 2017 г., в которых реальными трейдерами были размечены периоды трендов и боковиков, была построена модель машинного обучения (на основе усиленных деревьев решений *XGBOOST*). Однако несмотря на высокие метрики качества модели ($AUC=0,860$, $Accuracy=0,788$, $Precision=0,812$, $Recall=0,877$, $F1\ Score=0,843$) результаты оказываются неприменимы на практике ввиду слишком частых сигналов переключений.

При этом важным оказывается выбор входных параметров: помимо данных о ценах и объемах, качество может улучшиться за счет введения параметров, синтезированных из исходных данных (например, технических индикаторов), или параметров, внешних по отношению к ценовым рядам (например, макроэкономических индикаторов).

При этом, по-видимому, пока нейронные сети раскрыли не весь потенциал в анализе финансовых рынков: в существующих моделях нейронные сети применяются к прогнозированию и классификации финансовых временных рядов, тогда как наилучшие результаты нейронные сети показывают в задачах классификации неструктурированных данных, например, изображений и текстов.

Поэтому, в частности, возможен серьезный прогресс в использовании неструктурированных данных для прогноза, прежде всего текстов: финансовых новостей и аналитических прогнозов. Однако пока публикаций на эту тему, в которых были бы получены результаты, лучшие чем полученные с помощью известных алгоритмов анализа рядов цен и объемов, нет.

3.2. Алгоритм прогнозирования моментов разворотов рыночных трендов на основе скользящих окон

В данном разделе описан предложенный авторами алгоритм классификации состояний рынка, основанный на использовании методики скользящих окон.

Исходными данными являются файлы, название которых содержит категориальный признак *Stock* – идентификатор акции, а сами файлы содержат для каждой даты из определенного периода времени значения цены данной акции (на момент начала и конца, минимальное и максимальное), а также объем сделок:

- *Date* – дата;
- *Open* – цена открытия;
- *Close* – цена закрытия;
- *High* – максимальная цена;
- *Low* – минимальная цена;
- *Volume* – объем торгов.

Эти данные размечаются экспертами с помощью специально разработанного инструмента, при работе с которым эксперт вводит свой идентификатор, после этого получает возможность видеть значе-

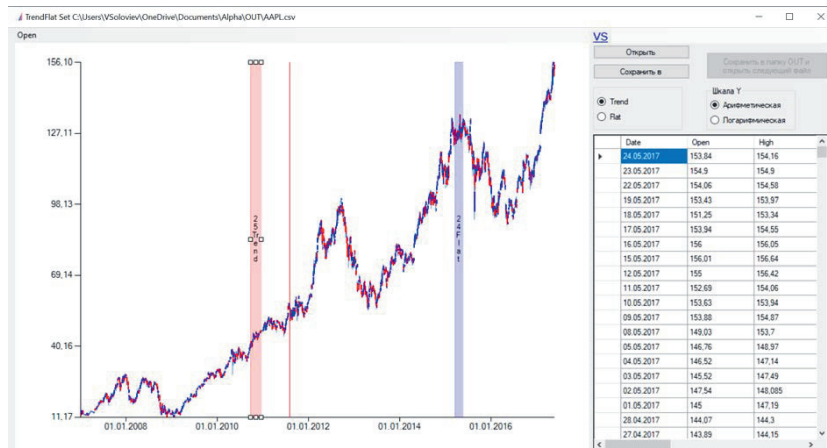


Рис. 22. Инструмент разметки данных о трендах

ния всех признаков, а также их графическую интерпретацию (рисунок 22).

При этом масштаб времени (и соответствующий ему минимальный шаг по времени) при проведении разметки данных может варьироваться в широких пределах, также возможно отображение графика как в арифметической, так и в логарифмической шкале, а также его приближение, удаление и прокрутка.

Результатом каждой процедуры разметки является определение экспертом состояния рынка финансового инструмента (тренд, боковик, «серая зона») на каждый момент времени внутри заданного временного периода («окна разметки»). В случае, если данный день определен как часть тренда, признак *Phase* для этого дня устанавливается равным значению *Trend*, если данный день определен как часть боковика, признак *Phase* для этого дня устанавливается равным значению *Flat*, если же данный день никак не идентифицирован, он считается частью «серой зоны» и признак *Phase* для этого дня оставляется пустым.

Результатом работы инструмента является создание файла метаданных, хранящего экспертные оценки состояния рынка в формате CSV (значения, разделенные запятыми).

- *Expert* – идентификатор эксперта (категориальный признак);
- *Date* – дата;

- *Open* – цена открытия;
- *Close* – цена закрытия;
- *High* – максимальная цена;
- *Low* – минимальная цена;
- *Volume* – объем торгов;
- *Phase* – фаза рынка (результативный категориальный признак).

Категориальный признак *Stock* – идентификатор акции – содержится в имени файла.

Для автоматизированного сбора размеченных данных используется специализированный скрипт на языке Python, позволяющий в автоматическом режиме создать набор данных для применения алгоритмов машинного обучения. Этот скрипт создает файл, в котором данные сортируются вначале по экспертам, затем по акциям, после этого по датам.

Данный файл впоследствии используется для обучения алгоритмов классификации, используемых при идентификации фаз рынка, а также для оценки качества построенных моделей.

Авторами был сформирован набор размеченных данных о состоянии рынка финансовых инструментов в различные периоды времени, включающий по согласованному с заказчиком перечню 600 финансовых инструментов за период с 2006 по 2017 г. ряды дневных значений цен открытия и закрытия, максимальной и минимальной цены, а также объема торгов, в которых экспертами были размечены окна трендов и боковых.

В качестве обучающей выборки использовались данные по датам до 11.10.2014 г., в качестве контрольной выборки – после 11.10.2014 г.

Общая схема алгоритма определения состояния рынка с использованием методологии скользящих окон выглядит следующим образом:

- первая модель (*Trend or Flat*) определяет состояние рынка на конкретном участке разметки между двумя заданными точками (тренд или боковик);
- вторая модель (*Changing Points*) определяет критические точки, в которых выявление одно состояние рынка сменяется другим;
- на обучающей выборке происходит последовательное обучение двух моделей, вначале *Changing Points*, затем *Trend or Flat*;
- на тестовой выборке (как и в продуктивной модели) первый день считается критической точкой, далее ежедневно запускается мо-

дель *Trend or Flat*, определяющая, какая тенденция наблюдается от последней критической точки до текущего дня, затем — модель *Changing Points*, определяющая, нет ли критической точки в текущий день.

Определение критических точек происходит на основании вычисления коэффициентов наклона регрессии цены закрытия по последним 5 дням, в связи с чем модель выдает прогноз с опозданием не менее чем на пять дней. При этом по знаку соответствующего коэффициента регрессии можно определить направление тренда (вверх или вниз).

При оценке качества моделей использовались следующие характеристики:

- *True Negative* (TN) — количество правильно распознанных боковиков;
- *True Positive* (TP) — количество правильно распознанных трендов;
- *False Negative* (FN) — количество ошибок первого рода, т. е. неправильно распознанных боковиков; в случае использования данной модели на практике ошибка первого рода означает, что в боковике будет принято решение оставаться в открытой позиции;
- *False Positive* (FP) — количество ошибок второго рода, т. е. неправильно распознанных трендов; в случае использования данной модели на практике ошибка второго рода означает, что в тренде будет принято решение открыть позицию;
- $Accuracy = \frac{TP+TN}{TP+FP+TN+FN}$ — доля правильно распознанных трендов и боковиков;
- $Precision = \frac{TP}{TP+FP}$ — мера точности, характеризующая, сколько полученных в результате использования модели классификации трендов на самом деле являются трендами;
- $Recall = \frac{TP}{TP+FN}$ — мера полноты, характеризующая способность классификатора «угадывать» как можно большее число трендов из ожидаемых (ложные прогнозы трендов не влияют на эту метрику);
- $F1\ Score = \frac{2(Precision \times Recall)}{Precision + Recall}$ — среднее гармоническое точности и полноты;
- Кривая *ROC* (Receiver Operating Characteristic) — линия, которая отображает изменение отношения доли верно классифи-

цированных трендов в их общем количестве к доле бокови-ков, неправильно отнесенных к трендам, при варьировании порога решающего правила;

- *AUC* (Area Under ROC Curve) – площадь под кривой ROC; чем выше показатель AUC, тем качественнее классификатор, при этом значение 0,5 демонстрирует непригодность выбранного метода классификации (это значение соответствует случайному гаданию на основе подбрасывания монеты), значение, меньшее 0,5, свидетельствует о том, что классификатор действует с точностью до наоборот: если положительные назвать отрицательными и наоборот, классификатор будет работать лучше.

Помимо стандартных метрик машинного обучения, рассчитывались также следующие метрики:

- *Freq* – частота переключений, которая показывает, во сколько раз количество реальных смен состояний рынка больше, чем количество предсказанных смен состояний рынка;
- *Delay* – задержка предсказания, т. е. доля участка данных, на которой система распознает истинное состояние рынка на данном участке;
- *Profit* – доходность, показывающая, на сколько процентов вырастет первоначальная сумма, если каждый раз в начале и в конце тренда занимать соответствующую позицию;
- *TimeInPos* – время в позиции, т. е. общая длина распознанных трендов.

Визуализация результатов производится с помощью средств пакета Plotly.

На графике исходных данных о ценах и объемах исходная разметка трендов и боковилов выделяется соответственно зеленой и розовой подложкой. Прогнозируемые участки восходящих и нисходящих трендов выделяется соответственно жирными зелеными и жирными красными линиями, а прогнозируемые боковики – жирными синими линиями.

Значение показателей *AUC* и *Accuracy* на уровне 0,58 означает, что состояние рынка правильно распознается для 58% дней тестовой выборки. Учитывая, что моменты начала и конца трендов и боковилов определяются нечетко, значение данного показателя представляется в целом удовлетворительным.

Значение показателя задержки (*Delay*) на уровне 16% означает, что для определения состояния рынка система должна увидеть его начальный участок длиной, в среднем составляющей 16% от общей длины участка с постоянной фазой рынка.

Значение показателя частоты переключений (*Freq*) на уровне –58% означает, что система определяет на 58% больше моментов переключений, чем есть в реальности. Большое количество «ложных срабатываний» связано с отсутствием консенсуса экспертов.

Средняя доходность торговой стратегии, определяемой классификатором, составляет 28% при среднем времени нахождения в позиции, равном 288 дням.

Примеры результатов расчетов представлены на рисунках 23–28.

3.3. Применение сверточных нейронных сетей для решения задач классификации и прогнозирования

На сегодняшний день сверточные нейронные сети (или convolutional neural network, CNN) представляют собой один из популярнейших, а следовательно, самых разрабатываемых методов глубокого обучения. Значительный вклад в развитие интереса к DCNN (Deep CNN) был внесен на ImageNet Big Scale Visual Recognition Challenge (ILSVRC) 2012 [114]. На соревновании Алекс Крижевски [115] использовал DCNN для классификации примерно 1,2 млн изображений на 1000 классов с рекордными результатами. С тех пор DCNN доминируют в последующих версиях ILSVRC или, если быть точнее, в части классификации изображений [116, 117, 118].

Классификация изображений, иными словами, задача категоризации изображений в один из нескольких predetermined классов, является фундаментальной проблемой в компьютерном зрении. Она формирует основу для таких задач компьютерного зрения как локализация, обнаружение и сегментация [119]. Хотя данная задача решается людьми на интуитивном уровне, она же становится сложнейшей для компьютера. До методов глубокого обучения для решения проблемы классификации использовался двухэтапный подход. Помеченные вручную признаки сначала извлекаются из изображений с использованием дескрипторов, далее они служат входными данными в обучаемом классификаторе. Главным препятствием для этого подхода было то, что точность задачи классификации в значительной степени зависит от разработанности этапа извлечения признаков. Последнее признавалось сложной задачей [120].

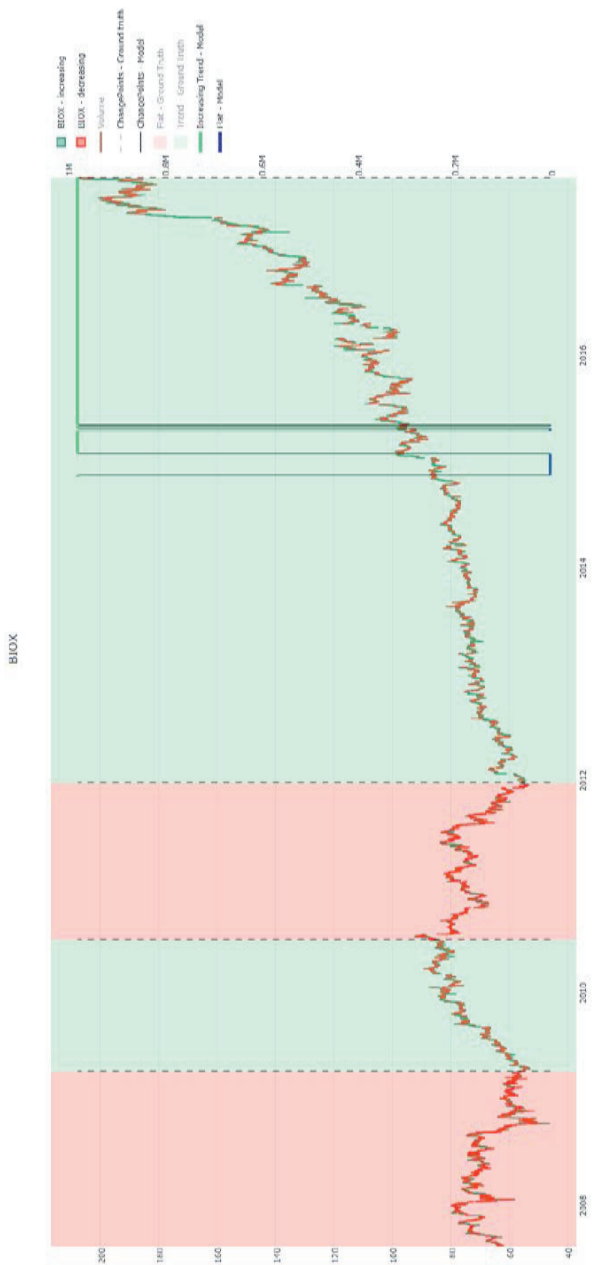


Рис. 23. Прогноз разворотов трендов (BIOX)

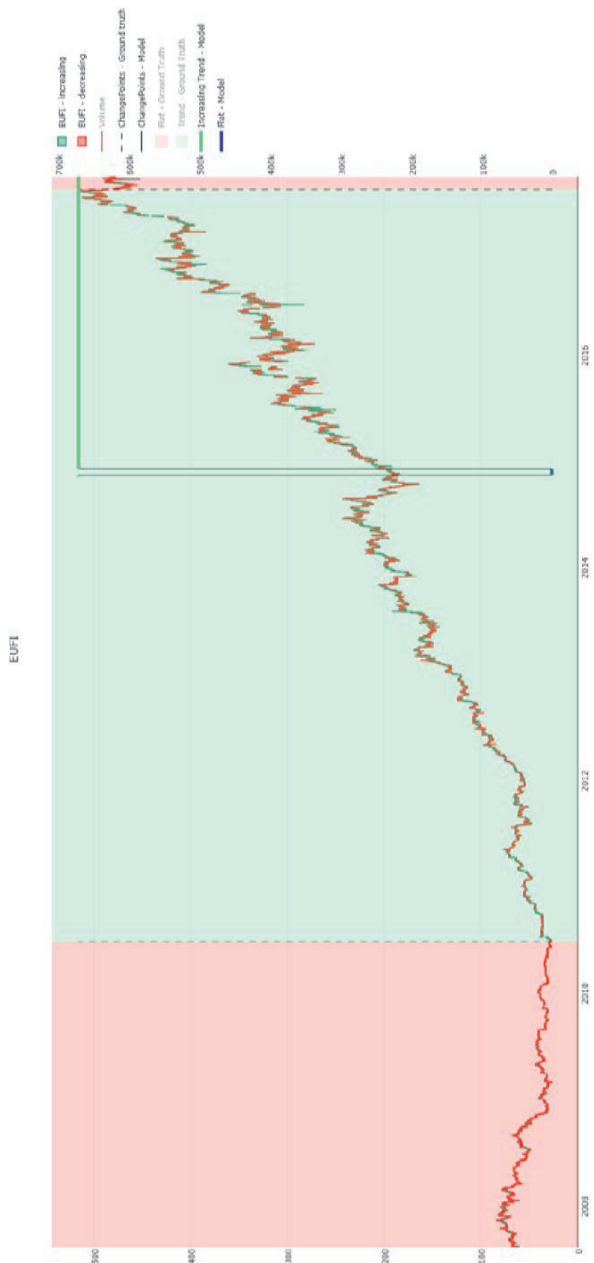


Рис. 24. Прогноз разворотов трендов (EUFi)

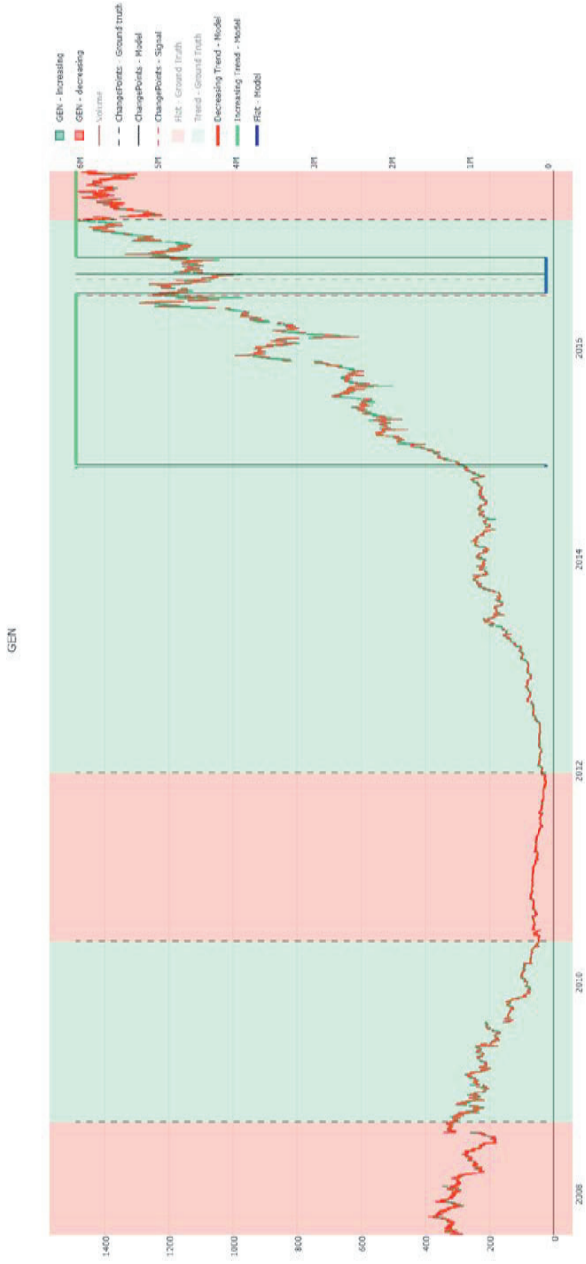


Рис. 25. Прогноз разворотов трендов (GEN)

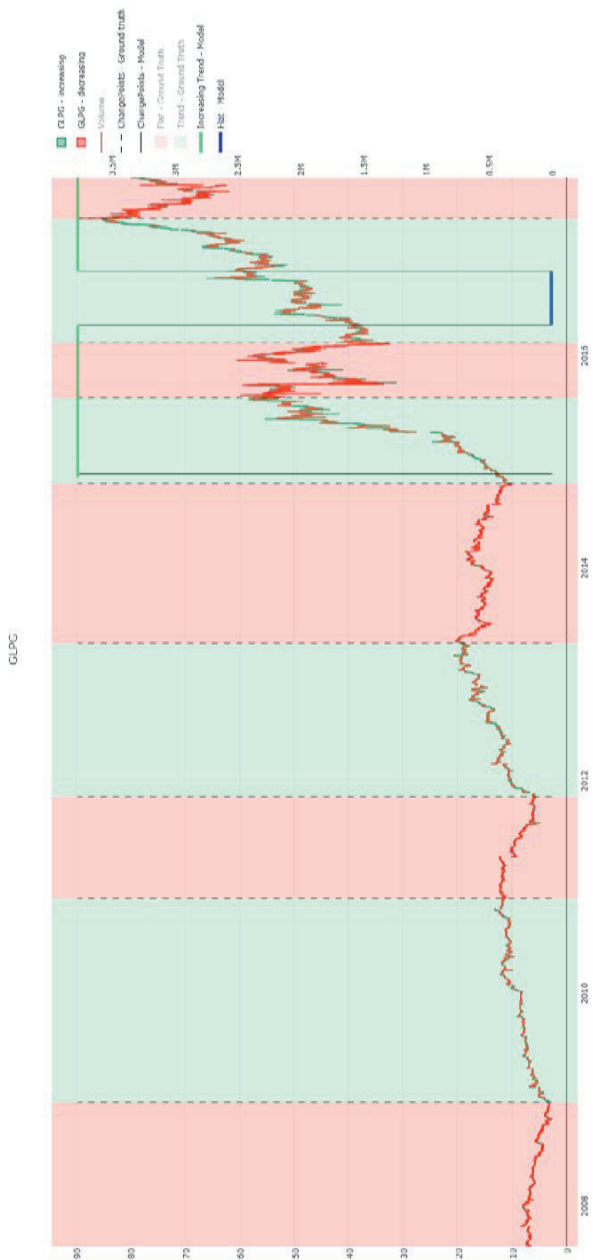


Рис. 26. Прогноз разворотов трендов (GLPG)

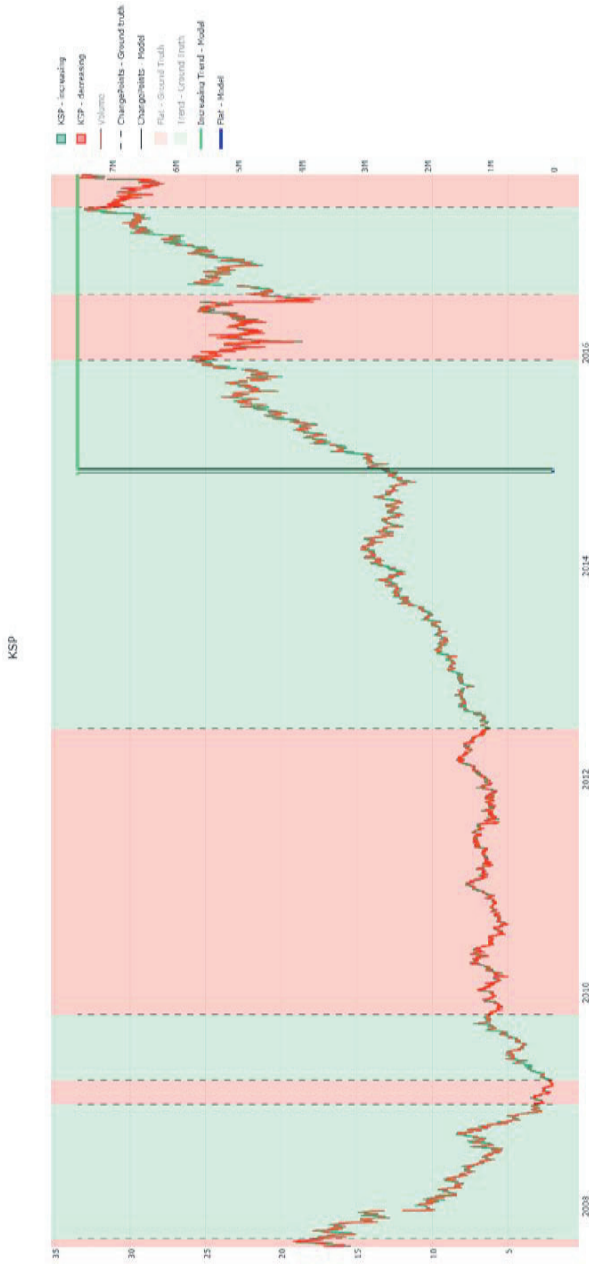


Рис. 27. Прогноз разворотов трендов (KSP)

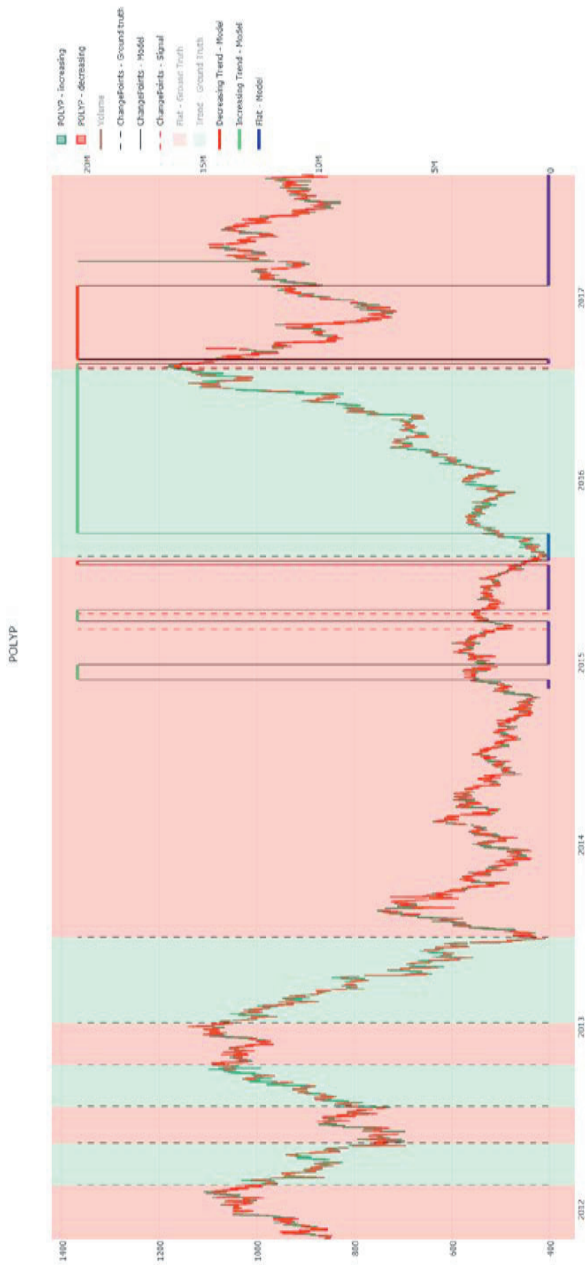


Рис. 28. Прогноз разворотов трендов (POLYP)

Эта проблема разрешилась благодаря алгоритмам глубокого обучения, которые используют множество уровней нелинейной обработки информации для извлечения и преобразования признаков, также как для анализа и классификации паттернов. Среди них сверточные сети стали ведущей архитектурой для большинства задач распознавания, классификации и обнаружения изображений [121]. Несмотря на некоторые ранние успехи (которые будут обсуждаться далее), глубокие CNN (DCNN) оказались в центре внимания в результате возрождения глубокого обучения [122, 123, 124], которое подпитывалось развитием графических процессоров, большими наборами данных и лучшими алгоритмами [115, 116, 117, 125]. Несколько достижений, такие как первая реализация GPU [126] и первое применение пулинга для DCNN [127] способствовали относительно недавнему развитию популярности CNN.

Более того, были предприняты попытки улучшения других аспектов DCNN:

- архитектура сети [128, 129, 130, 131];
- нелинейные активационные функции [132, 133];
- компоненты, касающиеся обучения с учителем [134, 135];
- механизмы регуляризации [129, 136];
- методы оптимизации [115, 137].

Данные работы определили направление для последующих исследований, сформулировав задачи, стоящие перед ними. Среди них, проблемы геометрических искажений [130], как правило медленные и большие модели [115, 116], открытие составительных примеров [138]. Все они привели к еще большему количеству исследований, посвященных классификации изображений с помощью DCNN.

Ранее были опубликованы общие обзоры глубокого обучения [120, 139, 140, 141], обзоры, посвященные глубокому изучению визуального понимания [142], обзоры последних достижений в CNN [143], и систематика DCNN для задач компьютерного зрения [144, 145]. Основываясь на них, данный обзор предназначен для понимания развития технологий и архитектуры CNN от их предшественников до современных систем глубокого обучения. Он также определяет краткое понимание их будущего и предоставляет несколько интересных неотложных направлений.

Сверточный слой

Сверточные слои служат для извлечения признаков, и, соответственно, изучают особенности входных изображений, которые представляют собой матрицы некоторого размера и глубины. Нейроны в сверточных слоях собраны в карты признаков. Каждый нейрон в карте признаков имеет поле восприятия, которое движется по предыдущему слою, позволяя ему свертываться с фильтром через нелинейную функцию активации, набором обученных весов соответствующих размеров. Таким образом заполняется карта признаков. Все нейроны в карте признаков имеют одинаковый фильтр; однако различные карты признаков в одном сверточном слое имеют разные фильтры, так что в каждом слое могут быть извлечены несколько признаков [119, 120, 146]. Нелинейные функции активации позволяют извлекать нелинейные особенности. Традиционно использовались сигмоидные и гиперболические касательные функции; недавно стали популярны усеченные линейные преобразования [147]. Их популярность и успех открыли область исследований, которая фокусируется на разработке и применении новых функций активации DCNN для улучшения нескольких характеристик производительности DCNN.

Пулинг (слой объединения)

Слои объединения используются для уменьшения пространственного разрешения карт признаков и, таким образом, достижение пространственной инвариантности к искажениям и сдвигам входа. Первоначально общепринятой практикой было использование слоев, в которых группа нейронов входа (поле восприятия) уплотняется до ее среднего значения, и это значение переходит следующий слой (average pooling) [119, 148, 149]. Однако в более поздних моделях [114, 115, 116, 132, 150, 151] на следующий слой переходит максимальное значение поля восприятия (max pooling) [126]. Формально max pooling выбирает наибольший элемент в каждом поле восприятия.

Полносвязные слои

Обычно несколько сверток и пулингов следуют друг за другом, чтобы при перемещении по сети извлекать больше абстрактных представлений признаков. Полносвязные слои, которые следуют за этими слоями, интерпретируют эти представления признаков и выполня-

ют функцию высокоуровневых рассуждений [115, 116, 135]. Для задач классификации на последнем этапе DCNN используется оператор softmax [114, 115, 127, 132, 150]. Хотя первые успехи были достигнуты благодаря использованию радиально-базисных функций (RBF) как классификатора над свертками [119], Tang [133] обнаружил, что замена softmax на метод опорных векторов (SVM) приводит к повышению точности классификации (подробнее далее). Альтернативой может служить простой линейный классификатор [127]. Сравнение производительности различных классификаторов в DCNN по-прежнему требует дальнейшего изучения и, следовательно, становится интересным направлением исследований.

Биология вдохновила несколько методов искусственного интеллекта, среди которых искусственные нейронные сети (ANN), эволюционные алгоритмы и клеточный автомат [152]. Однако, пожалуй, самая большая история успеха среди них – это сверточные нейронные сети или convolutional neural network (CNN) [153]. История этих сетей началась с нейробиологических экспериментов, проведенных Дэвидом Хьюбелем и Торстеном Визелем, нобелевскими лауреатами 1981 г. Они получили премию за работу, проведенную в 1959 г. [154]. Формально премия выдана за «работы, касающиеся принципов переработки информации в нейронных структурах и механизмов деятельности головного мозга».

В ходе эксперимента были обнаружены следующие эффекты:

- 1) нейроны ранних стадий первичной зрительной коры реагируют на точно ориентированные образцы света (например, полосы), но игнорируют более сложные структуры входного стимула, которые привели к реакциям нейронов более поздних стадий;
- 2) нейроны первичной зрительной коры образуют иерархию, состоящую из простых клеток, которые имеют локальные восприимчивые поля, и комплексных клеток, которые были инвариантными к сдвинутому или искаженному стимулам.

Одним из результатов стала модель зрительной системы со следующими свойствами [155]:

- соседние нейроны обрабатывают сигналы с соседних областей сетчатки;
- нейроны образуют иерархическую структуру, где каждый следующий уровень выделяет все более и более высокоуровневые признаки;

- нейроны организованы в так называемые колонки – вычислительные блоки, которые трансформируют и передают информацию от уровня к уровню.

Эти работы послужили ранним вдохновением для моделирования автоматизированных систем зрения, основанных на характеристиках центральной нервной системы.

В 1979 г. были предложены первые модели нейронной сети, основанные на идеях Хьюбеля и Визеля [156]: когнитрон и неокогнитрон. Архитектура последнего состояла из простых (сегодня это, в сущности, свертки) и комплексных (пулинг) ячеек, объединенных иерархически. Благодаря этой архитектуре сеть оказалась успешной при распознавании простых входных шаблонов, независимо от сдвига положения или значительных искажений в форме входного шаблона [157, 158]. Можно считать, что неокогнитрон заложил основу для развития CNN. Фактически, CNN были получены из неокогнитрона, и, следовательно, они имеют аналогичную архитектуру [120].

В 1989 г. Ликан с соавторами [147, 148] предложили первые многослойные CNN и успешно применили эти крупномасштабные сети к реальным проблемам классификации изображений: рукописные цифры и почтовые индексы.

Эти первоначальные CNN напоминают неокогнитрон. Однако ключевое различие заключалось в том, что они обучались с учителем, используя back propagation (алгоритм обратного распространения ошибки), в отличие от эвристического алгоритма в режиме без учителя, используемого их предшественником. Это позволило им больше полагаться на автоматическое обучение, а не на ручную подготовку для извлечения признаков [147, 148, 159], которые показали себя чрезвычайно сложными. Данные работы сформировали важную составляющую многих популярных DCNN [114, 115, 116, 117].

История развития алгоритма обратного распространения ошибки может быть представлена следующими значащими работами: [160, 161] – современная форма алгоритма (без упоминания ANN); [162, 163] – первое известное применение алгоритма для ANN; [73, 164]. Работа [97] популяризовала алгоритм, продемонстрировав, что с его помощью внутренние скрытые нейроны ANN могут обучаться так, чтобы представлять собой важные признаки в области задач.

В 1998 г. была опубликована статья [120], в которой представлена LeNet-5 – CNN для задачи индивидуальной классификации символов для распознавания документов. В работе подчеркивались основ-

ные преимущества CNN по сравнению с традиционными ANN: они требуют меньше свободных параметров (из-за распределения весов) и рассматривают пространственную топологию входных данных, что позволяет им иметь дело с изменчивостью 2D-фигур. В дополнение Лекан представил популярный набор данных MNIST (Modified National Institute of Standards and Technology) из 70000 рукописных цифр, который с тех пор широко использовался для нескольких задач компьютерного зрения, в частности, для классификации изображений и проблем распознавания.

Вплоть до недавнего времени эта архитектура была стандартным шаблоном для построения сверточных сетей: свертка чередуется с пулингом несколько раз, затем несколько полносвязных слоев. Основные строительные блоки — свертки 5×5 со сдвигом 1 и пулинг 2×2 со сдвигом 2. Свертки играют роль детекторов признаков, а пулинг используют для уменьшения размерности, эксплуатируя тот факт, что изображения обладают свойством локальной коррелированности пикселей — соседние пиксели, как правило, не сильно отличаются друг от друга. Таким образом, если из нескольких соседних получить какой-либо агрегат, то потери информации будут незначительными.

В конце 1990-х и начале 2000-х гг. исследования нейронных сетей стали менее популярны. Широко распространено мнение, что их изучение было недопустимо из-за проблем с популярным алгоритмом оптимизации — градиентным спуском, разновидность которого используется в *back propagation*. В частности, одной из проблем считалось явление, известное как «плохие локальные минимумы» (*poor local minima*).

В то время, как CNN не пользовались популярностью, Simard и другим [165] удалось добиться самого известного результата классификации в наборе данных MNIST, улучшив предыдущие наилучшие результаты, полученные SVM [166]. Ссылаясь на преимущества, упомянутые в [120], и используя CNN для визуальных задач, они расширили размер и качество набора данных MNIST и предложили использовать простые *software loop* для операций свертки. Эти циклы использовали свойство *backpropagation*, которое позволяет ANN выражаться модульным способом, а это допускало модульную отладку программного обеспечения. Еще в работе [120] было доказано, что увеличение размера набора данных с помощью искусственно созданных аффинных преобразований приводит к улучшению про-

изводительности сети. Симард с соавторами [165] смогли улучшить качество увеличенной части набора данных, используя упругие деформации изображения. Эта работа была частью серии нескольких приложений оптического распознавания символов, использующих сверточные сети. В частности, Microsoft использовала CNN для английских рукописных цифр [165, 167], распознавания арабского рукописного текста [168] и распознавания рукописных символов в Восточной Азии [169].

Поскольку исследования нейронных сетей замедлились в конце 1990-х и начале 2000-х гг., развитие CNN также было затруднено, но оно возродилось в 2006 г. [122, 123, 124, 170]. Описанные в этих исследованиях преимущества скорости и точности предварительной подготовки без учителя (unsupervised pre-training) вдохновили Ранзато и др. [127] использовать архитектуру, подобную DCNN, обученную без учителя, чтобы изучать иерархические разреженные признаки, нечувствительные к малым сдвигам и искажениям. Их подход, который ввел *max pooling*, достиг результатов, очень близких к современному уровню для MNIST и CALTECH-101 (набор данных Калифорнийского технологического института [171, 172]).

Считая обсужденные методы предварительной обработки сложными и ограниченными, Weston и др. [173] представили более простой способ глубокого обучения путем слияния нелинейных алгоритмов включения с глубокими многослойными архитектурами (в том числе DCNN), прошедшими обучение с учителем. Полученная схема глубокого обучения с частичным привлечением учителя привела к появлению конкурентоспособных коэффициентов ошибок (*error rate*) в наборе данных MNIST по сравнению с другими техниками полуавтоматического обучения [174, 175].

Несмотря на это, в последнее время обучение с учителем стало ведущей парадигмой для обучения DCNN. Однако обучение с частичным привлечением учителя более биологически правдоподобно. Например, если рассмотреть, как дети узнают об их среде или, точнее, как они учатся распознавать или классифицировать объекты. Они, как правило, снабжаются несколькими примерами своими опекунами, аналогичными обучению с частичным привлечением учителя, и они используют это для обобщения на неизвестных объектах. Таким образом, чтобы приблизить наши текущие модели к природе, предполагается, что будущие DCNN вернуться к использованию полуавтоматических схем.

Алгоритмы глубокого обучения, которые используют в настоящее время, доступны с 1980-х гг., однако их обучение требует дорогостоящих вычислений из-за недостаточной мощности оборудования, доступного до 2006 г. Кроме того, выполнение программы свертки делают DCNN значительно медленнее по сравнению со стандартными нейронными сетями того же размера. Чтобы преодолеть эти ограничения, в [59] предложили три новых метода ускорения DCNN: разворачивание свертки, использование основных программных подпрограмм линейной алгебры и использование графических процессоров. Хотя графические процессоры уже были применены к ANN [176, 177], эта работа – первая реализация DCNN с использованием графических процессоров. Со временем это стало важным аспектом большинства удостоенных наград или самых современных DCNN [114, 115, 116, 128, 131, 135, 149, 150, 154]. Хотя разработка усовершенствованного аппаратного обеспечения для облегчения вычисления DCNN по-прежнему остается открытой областью исследований, в последние годы она превратилась в источник дохода, что привело к применению коммерчески доступного оборудования, или смещению фокуса на алгоритмическое развитие во многих научных работах. Ожидается, что неизбежные достижения в области аппаратного и программного обеспечения будут сосредоточены на развертывании DCNN на мобильных устройствах.

В 2010 г. Шерер с соавторами [178] эмпирически показали, что операция *max pooling* может привести к улучшению обобщения и более быстрой конвергенции по сравнению с операцией подвыборки (*SubSampling*).

Подробный теоретический анализ *max pooling* и *average pooling*, дополненный эмпирическими оценками, был приведен в [179]. Авторы пришли к выводу, что производительность любого из методов зависит от данных и их особенностей, и что для проблемы классификации использование одной стратегии пулинга не оптимально. Несмотря на это, *max pooling* по-прежнему остается ключевым компонентом нескольких современных DCNN [114, 115, 149, 150].

С 2006 г. DCNN успешно применялась для решения самых разных задач. Однако они по-прежнему мало принимались сообществом компьютерного зрения и машинного обучения. Это изменилось после ILSVRC 2012, когда DCNN в режиме обучения без учителя достиг рекордных результатов классификации по подмножеству набора данных ImageNet [114]. Эта сеть произвела революцию в области

компьютерного зрения, и в результате DCNN с тех пор стали ведущей архитектурой для большинства визуальных задач.

В AlexNet были реализованы несколько новых и необычных приемов. Вместо того, чтобы использовать традиционные сигмоидные или гиперболические функции активации, авторы вдохновились работой [180] и использовали ReLU [145], что позволило значительно ускорить время обучения. Поскольку их сеть была слишком большой, чтобы вписаться в один графический процессор, они распорстраняли ее на два графических процессора, расположенных в параллельной конфигурации, что было похоже на многоколоночные DCNN, предложенные Ciresan, Meier и Schmidhuber [181].

Сеть состояла из пяти сверточных слоев, три из которых сопровождались пулингом, и тремя полностью связанными слоями. Некоторые части слоя работали на одном графическом процессоре, а другие находились на втором графическом процессоре. Графические процессоры взаимодействовали друг с другом только на определенных уровнях.

Чтобы преодолеть переобучение, авторы использовали метод регуляризации, известный как Dropout [135]. Результатом стал сильный эффект регуляризации, который значительно сократил переобучение [182].

Более того, переобучение было уменьшено за счет дополнения данных, популярной процедуры для искусственного увеличения набора данных. В частности, они создали больше изображений, применяя сдвиги и горизонтальные отражения к изображениям, изменяя интенсивность их цветовых каналов и выполняя анализ основных компонентов (РСА) по значениям пикселей, что привело к повышению эффективности классификации. AlexNet широко использовалась для различных целей с момента ее разработки. Множество исследований использовали ее для сравнения с своими сетями или в качестве базовой модели для тестирования новых алгоритмов.

С 2012 г. DCNN доминировали в задачах классификации изображений, в частности ILSVRC. Фактически, они были триумфальными в каждой проблеме классификации ImageNet: [115, 116, 150, 183].

Более глубокие модели, как правило, более точны и, таким образом, дают лучшие эмпирические результаты; однако, по мере увеличения глубины, увеличиваются и вычислительные затраты. Были предприняты несколько попыток повысить точность классифика-

ции DCNN, изменив их архитектуру для повышения производительности без потери вычислительной нагрузки, налагаемой на такие модели. В частности, модели [115, 116, 150, 182] сосредоточены на более глубоких или более широких сетях для повышения точности, с несколькими трюками, начиная от уменьшения размеров и заканчивая остаточным обучением, для обработки связанного с ним вычислительного напряжения, размещенного в более глубоких сетях. Это привело к классической инженерной дилемме между более глубокими моделями, которые являются более точными, но вычислительно дорогими, и более мелкие модели, которые легче и дешевле в обучении, но не дают одинаково точной классификации. В итоге, несмотря на то, что было предпринято несколько попыток решить проблему сохранения точности с уменьшением вычислительных затрат, она остается открытой задачей для DCNN.

Несмотря на многообещающие результаты классификации изображений, полученные глубокими сверточными сетями, все еще есть проблемы, которые необходимо решить. Рассмотрим некоторые из таких вызовов, а также отдельные тенденции в недавних работах.

Несмотря на эмпирические успехи DCNN, теоретические доказательства того, почему они преуспевают, отсутствуют. С этой целью Маллат [184] доказал трансляционную инвариантность и устойчивость к деформации признаков, извлеченных из нескольких сверточных сетей. Вятовски и Бёлскей [185] провели математический анализ признаков, извлеченных DCNN, и теоретически установили стабильность деформации и вертикальную трансляционную инвариантность. Были предложены другие методы визуализации, все с намерением понять внутренние механизмы DCNN [186, 187, 188]. Дальнейший прогресс зависит как от теоретических доказательств, так и от практических исследований, которые приводят к улучшению понимания сверточных сетей.

Хотя DCNN устойчивы к мелкомасштабным деформациям [189], их конечные результаты не являются геометрически инвариантными [129, 149, 190]. В частности, они чувствительны к глобальным сдвигам, вращениям и масштабированию [129]. Чтобы устранить отклонения в сдвигах, Ли и др. [191] предложили вероятностный max pooling, в то время как MOP (Multi-scale Orderless Pooling) в [129] показала устойчивость к некоторым геометрическим отклонениям. Метод TI-пулинга (transformation-invariant pooling), представленный в [192] эффективно обрабатывает вращения и изменения масштаба

и, таким образом, преобразовывает трансформационную инвариантность в архитектуре DCNN. Модуль пространственного трансформатора (spatial transformer module), предложенный в [193] изучает трансляцию, масштаб, поворот и инвариантность к деформации. Таким образом, интересным направлением является исследование того, необходимы ли дальнейшие фундаментальные изменения для архитектуры DCNN с целью повышения их универсальной устойчивости. Более того, несмотря на большое количество изображений, доступных в современных наборах данных, таких как ImageNet, все еще возможно, что наши текущие наборы данных не подходят для решения задач инвариантности, с которыми мы сейчас сталкиваемся. Таким образом, еще одно перспективное направление — собирать или генерировать новые данные, которые не обязательно приводят к большему набору данных, подобно той тенденции, которая наблюдается в последние несколько лет, но будут способствовать изучению возможностей DCNN, что способствует созданию более надежных моделей.

Среди открытых проблем, пожалуй, самая главная заключается в том, что точность классификации DCNN и классификаторов в целом не надежна, что доказывают состязательные примеры. Это небольшие, но преднамеренные возмущения, применяемые к изображениям с целью введения в заблуждение или обмана системы распознавания или классификации. Когда такие возмущения используются для изменения изображения, люди легко могут правильно классифицировать изображение [194, 195], тогда как классификаторы относят изображение к другому классу. Начиная с первоначального открытия этого явления [137], несколько работ подтвердили уязвимость DCNN к таким изображениям и предложили некоторые жизнеспособные контрмеры для их смягчения [134, 137, 193, 195, 196, 197, 198, 199, 200, 201].

До сих пор наиболее перспективные попытки решить эту проблему были сосредоточены на:

- различных методах обучения: состязательное обучение [193] и дистилляция [199];
- генеративных методах предварительной обработки: шумоподавляющие автокодировщики (denoising autoencoders) [194];
- изменениях архитектуры DCNN: сделать ее более нелинейной или «наказывать» необычные сигналы [134, 195].

Кроме того, была введена теоретическая основа для исследования устойчивости сети. Как следствие, установлены фундаментальные ограничения на устойчивость отдельных классификаторов относительно меры различимости между классами [202, 203]. Более того, Бастани с соавторами [204] недавно предложили две статистические меры, *adversarial frequency* and *adversarial severity*, для измерения устойчивости на основе формализованной концепции точечной устойчивости, закодированной как проблема ограниченной задачи оптимизации. Интересно, что сети RBF (радиально-базисных функций) были по своей природе невосприимчивы к состязательным примерам [193]. Их объединение со стандартной архитектурой DCNN – это один из многих подходов, которые требуют дальнейшего изучения.

В дополнение к состязательным примерам, в [193] обнаружили, что можно сгенерировать изображения, которые полностью неизвестны для людей, но современные DCNN классифицируют и узнают их с чрезвычайно высокой уверенностью. Эти изображения были сгенерированы с использованием эволюционных алгоритмов и оптимизации на основе градиента и называются обманчивыми изображениями. Обращаясь к этим изображениям как к классу мусора, в [127] показали, что они влияют не только на глубокие сети (это исследование включало DCNN), но и на мелкие классификаторы. Работа [134] назвала эти образы бессмысленными и начала работу, показывающую, что изменения архитектуры DCNN являются жизнеспособным решением для преодоления этого недостатка. Тем не менее, как и состязательные примеры, обманчивые изображения остаются задачей, требующей дальнейшего внимания.

Состязательные и обманчивые изображения подчеркивают огромный разрыв между возможностями зрения человека и систем компьютерного зрения. Это открыло совершенно новую область исследований, в которой основное внимание уделяется созданию таких изображений и разработке систем, которые устойчивы к ним. Естественно, поскольку DCNN стали ведущей архитектурой для визуальных задач и с учетом того факта, что они не застрахованы как от состязательных, так и обманчивых образов, исследование устойчивости сверточных сетей является важной проблемой.

Несмотря на то, что DCNN превзошли показатели человека на наборах изображений с одним объектом на них (MNIST и ImageNet). Изображения реального мира обычно содержат несколько классов,

которые относятся к различным объектам [206]. Способность правильно описывать семантическое содержание изображения в правильно сформированных предложениях на естественном языке является сложной задачей [207], которая находится на пересечении компьютерного зрения и обработки естественного языка. Считается, что решение может быть в объединении DCNN с RNN. В работе [205] представлена структура DCNN-RNN, в которой DCNN извлекает семантические представления из изображений, в то время как RNN моделирует отношения между классами и меткой. В [208] использовали DCNN для классификации изображений, а RNN для моделирования предложений. Потом объединили их в единую сеть, которую использовали для генерации описаний изображений на английском языке. Другим перспективным направлением является подготовка этих комбинированных архитектур с использованием обучения с подкреплением, и хотя системы, сочетающие глубокое обучение и обучение с подкреплением, все еще находятся на раннем этапе, они уже дали некоторые исключительные результаты классификации изображений [209].

Действующие DCNN в основном используют обучение с учителем, поэтому они не могут использовать огромное количество немеченых данных, доступных в Интернете, хранящихся в облачных системах или даже снятых мобильными устройствами. Вдобавок, человеческое обучение естественно не контролируется, и поэтому ожидается, что будущие модели DCNN будут пытаться имитировать человеческий мозг в этом направлении. Недавние попытки в этом направлении включают работы: [210, 211, 212, 213, 214].

DCNN требуют нескольких гиперпараметров, например, количество эпох, для обучения модели и скорости обучения. Но их определение требует тщательной настройки, которая часто основывается на опыте эксперта, эмпирических правилах или на чрезмерно изоширенных поисковых методах. Используя автоматизированную байесовскую оптимизацию, в [215] смогли найти лучшие гиперпараметры быстрее эксперта-человека, и получили отличные результаты классификации изображений. Однако этот метод занимает много времени и не очень хорошо масштабируется для крупных моделей [143]. Соответственно, необходимы альтернативы по этим направлениям. Возможным решением может быть использование эволюционных алгоритмов, таких как метод роя частиц [216], ставший популярной

методикой оптимизации для проведения поиска гиперпараметров, а затем интеграции результатов с DCNN. Возможно, эти два популярных, биологически вдохновленных метода, будут успешны, работая в тандеме. Еще одним интересным, но сложным направлением является использование дискриминационных и выразительных классификационных способностей DCNN в роботизированных системах. Успехи на этом пути описаны в [217, 218, 219].

Еще одна тенденция, которая набирает популярность, — это факторизация сверток для повышения эффективности вычислений. Этот метод был популяризирован моделью [175], в которой было установлено, что свертки могут быть факторизованы с последующими свертками. Например, они обнаружили, что 3×1 свертки, за которыми следуют 1×3 свертки, привели к 33%-ному сокращению вычислений по сравнению с использованием одного фильтра 3×3 , имеющего такой же эффективный размер поля восприятия. В [220] предложили архитектуру Xception, в которой они заменяют исходные строительные блоки глубинными отделяемыми свертками (глубинная свертка с последующей точечной сверткой). Глубинные свертки были также использованы в [221]. Повышение эффективности использования такого типа факторизации, особенно в связи с развертыванием мобильных DCNN, имеет важное значение и, вероятно, будет широко использоваться в будущих моделях.

Данный обзор анализирует большинство значительных достижений в отношении архитектур CNN, механизмов регуляризации, методов оптимизации. Несмотря на некоторые успехи в других областях, DCNN характеризуется заметным прогрессом в решении задачи классификации, устанавливая современный стандарт по нескольким сложным наборам данных и доминируя в многочисленных соревнованиях, связанных с классификацией изображений. Фактически, на всех известных наборах данных с одним объектом (классом) на изображении сверточные сети показали себя лучше, чем человек. Современный рост DCNN привел исследователей к изучению качества их классификации, устойчивости и вычисляемых характеристик, что привело к формированию направления будущих исследований.

Таким образом, сверточные сети — эффективный инструмент, применяемый для классификации изображений, достаточно проработанный и перспективный для применения в отдельных областях экономики или бизнеса, в частности, в финансах. Хотя пока

в этой сфере нет публикаций, заслуживших внимания исследователей, сверточные нейронные сети – это потенциально эффективный способ распознавания фаз рынка.

Авторами был построен алгоритм предсказания будущих моментов смены состояний рынка на основе сверточной нейронной сети, показавший почти 20%-ную доходность в случае 75-дневного горизонта прогнозирования.

Однако существенным недостатком алгоритма является то, что позиции открываются примерно в 6 раз чаще, чем усредненным экспертом, что повышает транзакционные издержки.

Чтобы лучше понять причины ошибок, проанализируем матрицу сопряженности, получившуюся в результате симуляции торгов (рисунок 29).

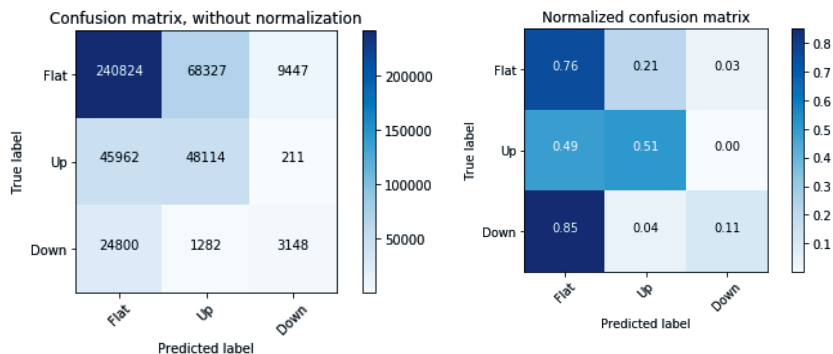


Рис. 29. Матрица сопряженности и ее нормализованная версия

Основной причиной не вполне удовлетворительной точности предсказаний являются противоречия в исходной разметке данных. Также следует учитывать дисбаланс в отношении класса нисходящих трендов, что привело относительно более низкой точности их распознавания моделью TF. В целом, однако, разработанная математическая модель и алгоритм машинного обучения для предсказания будущих моментов смены состояний рынка с использованием сверточной нейронной сети является рабочим, хотя и требующим дополнительной калибровки.

На рисунке 30 приведена одна из иллюстраций результатов симуляции торгов.

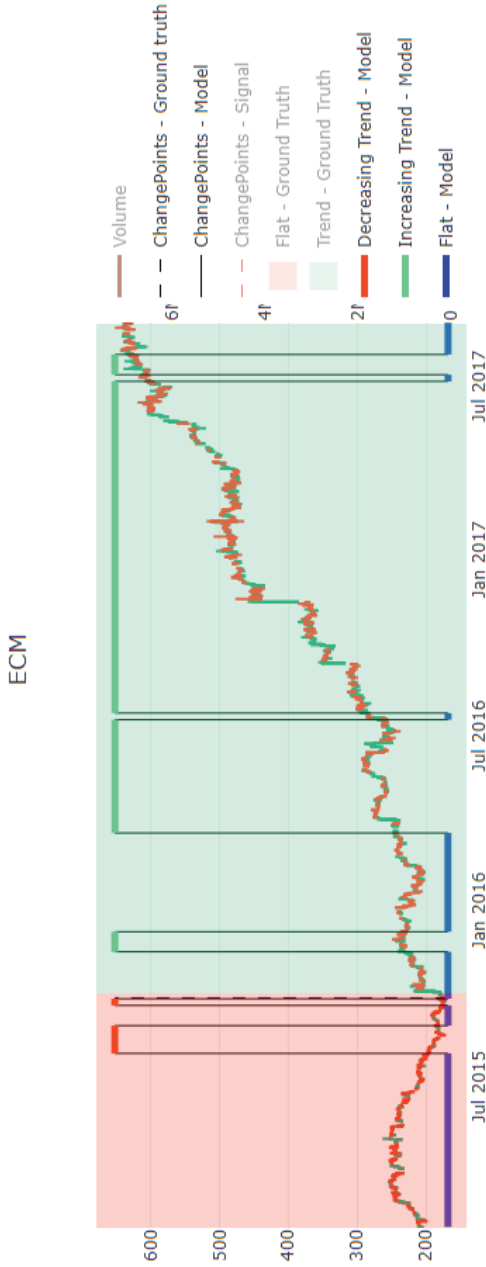


Рис. 30. Иллюстрация результатов симуляции торгов

3.4. Применение генеративно-сопоставительных нейронных сетей для решения задач классификации и прогнозирования

Одной из особенностей классических искусственных нейронных сетей прямого распространения является то, что будучи универсальными классификаторами, они неспособны породить образцы наборов данных, принадлежащих распознаваемым классам объектов. Глубокие многослойные перцептроны, например, способны выделять довольно высокоуровневые признаки в, например, изображениях, что позволяет им реализовывать сложную классификацию изображений, которые сеть не видела на этапе обучения. Но все равно, сгенерировать изображение, которое сеть не видела при обучении, т. е. это задача другого класса. Решить ее и призваны генеративные модели, в частности – ставшие весьма популярными за короткий период своего существования, генеративно-сопоставительные сети (порождающие соперничающие сети, generative adversarial networks, GANs).

Идея генеративно-сопоставительных сетей была предложена Я. Гудфеллоу из Университета Монреаля [222]. В течении буквально пары лет этот метод нашел свое применение в задачах семантической сегментации изображений, анализа медицинской информации, распознавании материалов, анализе временных рядов [223, 224, 225, 226, 227, 228, 229, 230, 231, 232, 233, 234, 235, 236, 237, 238, 239, 240, 241, 242].

Принципиальная схема генеративно-сопоставительной сети представлена на рисунке 31.

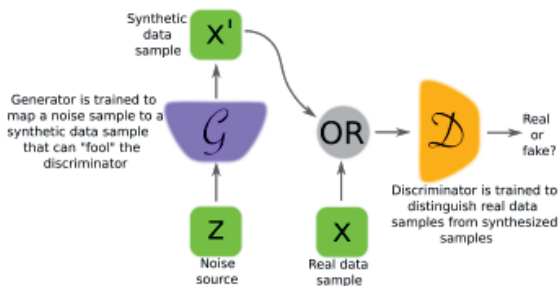


Рис. 31. Принципиальная схема генеративно-сопоставительной сети.

Здесь X – реальный объект предметной области (напр., рисунок); Z – источник случайного шума; G – нейронная сеть генератора; X' – искусственно сгенерированные данные, мимикрирующие реальные; OR – случайный выбор одного из двух входов; D – нейронная сеть дискриминатора

Основная идея генеративно-сопоставительных сетей — обучение одновременно двух сетей — генератора и дискриминатора. Генератор принимает на вход случайный вектор (источник энтропии, иногда интерпретируется как пространство скрытых переменных, *latent space*) и генерирует некоторый образ. На вход дискриминатора подается либо созданный генератором искусственный образ, либо объект реальной обучающей выборки. Его задача — отличить искусственный образ от реального. По идее, в процессе обучения генератор обучается распределению исходной выборки и начинает создавать образы все более близкие к реальным в то время, как дискриминатор становится все более точным в распознавании реальных образов.

GAN не первая генеративная модель, однако Гудфеллоу с соавторами предложили оригинальный метод сопоставительного обучения таких моделей, когда две сети «соревнуются» в решении противоположных задач.

Исследователи выделяют несколько вариаций изначальной идеи генеративно-сопоставительных сетей.

1. Полносвязные GAN — изначальная предложенная архитектура, где генератором и дискриминатором служат многослойные сети прямого распространения [243, 244, 245].
2. Сверточные GAN (*deep convolutional GAN, DCGAN*) — используют многослойные сверточные сети. Являются логичным развитием идеи GAN в применении к задачам синтеза изображений. Недостатком сверточных генеративных сопоставительных сетей является более долгий процесс обучения модели [246].
3. Условные GAN (*conditional GAN, CGAN*) — архитектура, в которой как генератору, так и дискриминатору на вход дополнительно подается вектор, указывающий на класс объекта. Такие сети могут генерировать условное распределение выборки, с указанием конкретного класса. Таким образом, они используются для моделирования мультимодальных распределений [247].
4. Сопоставительные автокодировщики (*Adversarial Autoencoders, AAE*). Автокодировщики — это сети, состоящие из двух частей — кодировщика и декодера, которые обучаются детерминистическому отображению из пространства данных в пространство скрытых переменных (обычно гораздо меньшей размерности) и обратному отображению. Сопоставительное обучение используется для оптимизации, схожей с вариационными автокодировщиками и обыч-

но используется для придания пространству скрытых переменных осмысленной организации в терминах предметной области.

5. GAN с моделями вывода (ALI, BiGAN) – расширяют функциональность GAN механизмом вывода скрытых переменных из объекта предметной области. обычные GAN могут сгенерировать правдоподобный объект из случайного вектора в пространстве скрытых переменных (latent space). Две независимо предложенные архитектуры состязательного обучения выводу (adversarial learning inference) и двунаправленных GAN (Bidirectional GAN) предоставляют механизм нахождения обратного преобразования, что может быть полезно для задач выделения признаков. Однако на сегодняшний день, надежность таких методов серьезно ограничена.

На рисунке 32 представлена принципиальная схема двунаправленной GAN.

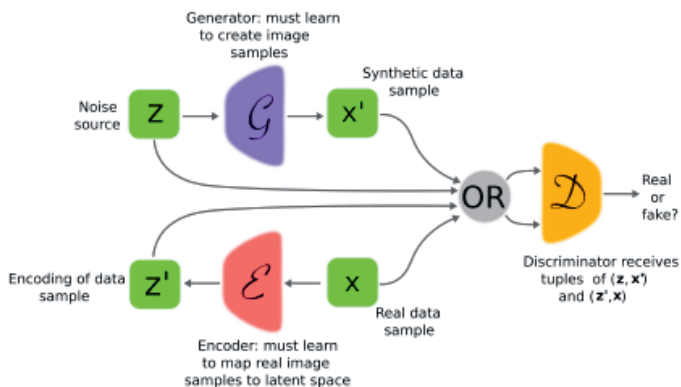


Рис. 32. Принципиальная схема двунаправленной GAN.

Здесь E – нейронная сеть кодировщика; Z' – представление реального объекта в пространстве скрытых переменных; OR – случайный выбор пары (X, Z') либо (X', Z)

Поиск новых областей применения генеративно-состязательных сетей является в данный момент очень активной областью исследования. Они уже хорошо себя зарекомендовали в таких областях, как классификация изображений, генерация изображений по текстовому описанию, преобразование изображений, повышение разре-

шения. И в 2018 г. генеративно-состязательные сети остаются активно развивающейся методикой машинного обучения.

Однако за время своего существования генеративно-состязательные сети показали существенные проблемы в использовании, такие как коллапс модели (ситуация при которой генератор выдает один объект при любых значениях входов), нестабильность обучения, отсутствие общепринятых подходов к оценке эффективности.

Искусственные нейронные сети доказали свою эффективность в решении многих задач, однако эта эффективность в большой мере зависит от умения подбирать правильную архитектуру сети под каждую конкретную задачу анализа данных.

При проектировании архитектуры сети эксперт сталкивается с необходимостью принять множество решений, как количественных, так и качественных, от реализации которых напрямую зависит производительность результирующей модели. Среди них такие как: выбор количества слоев, количества нейронов в каждом слое, выбор функции активации, использование рекуррентных и сверточных слоев. В процессе такого проектирования задачей эксперта является балансирование между вариативностью модели и склонностью к переобучению. Существует еще и внешние факторы, которые необходимо брать в расчет: располагаемая вычислительная мощность, временные рамки решения задачи и т. д.

Пространство возможных нейронных сетей, в котором ведется поиск, огромно. При проектировании нейронных сетей специалисты пользуются определенными эвристическими правилами и инструментами диагностики, однако полноценной методологией такой поиск назвать сложно, это скорее творческий процесс.

Весьма естественным направлением исследований в области искусственного интеллекта является построение методов и инструментов сокращения человеческого участия в построении систем машинного обучения, автоматизация данного процесса. В настоящее время существует два принципиально отличающихся подхода: автоматизация машинного обучения (AutoML) и нейроэволюция (evolutionary artificial neural networks, EANN).

Дальнейшие исследования привели к созданию метода NEAT (neuroevolution through augmenting topologies), существенно оптимизированного по использованию вычислительных ресурсов. Совсем недавно этот метод был адаптирован для эволюционного поиска структур глубокий нейронных сетей.

На сегодняшний день эволюционное программирование дает возможность создавать сети, по производительности сравнимые с лучшими в своем классе при нулевом участии человека в процессе проектирования и обучения для задач классификации изображений и задач распознавания речи.

В настоящее время ведутся активные работы с использованием нейроэволюции в таких областях как: предсказание временных рядов ансамблевыми моделями, предсказание энергопотребления компьютерными кластерами, построение межпланетных траекторий, распознавание языка речи, предсказание цен на нефть.

Главным недостатком нейроэволюционного подхода является весьма высокие требования к вычислительной мощности для поддержки данного процесса до сходимости. Развитие глубоких нейронных сетей и распространение больших данных еще выше поднимает эту планку. Поэтому, в настоящее время мы наблюдаем спад интереса к эволюционному программированию искусственных нейронных сетей: время программиста и специалиста в области анализа данных сейчас стоит дешевле, нежели требуемое процессорное время.

Однако большие корпорации, располагающие достаточными вычислительными мощностями, могут позволить себе эксперименты с нейроэволюцией. По-видимому, с ростом производительности вычислительной техники интерес к эволюционным методам в машинном обучении будет возвращаться (возможно итерационно).

Авторами был разработан алгоритм предсказания будущих моментов смены состояний рынка на основе генеративно-состязательной нейронной сети.

Диагностика обучения генеративно-состязательных сетей показала несколько вариантов динамики обучения. Проиллюстрируем их на характерных случаях из нашего пула.

Первый случай можно назвать успешным обучением (рисунок 33).

Валидационная точность растет и приближается асимптотически к точности на обучающей выборке, затем начинает снижаться за счет переобучения. Такая картина наблюдалась нами на сбалансированных не самых вариативных моделях. Стоит заметить, что, как видно на рисунке 33, точность приближается к значению 1,0 практически вплотную. Однако, это не обязательно говорит нам об исключительном качестве модели. Необходимо вспомнить параметры набора данных, а именно, в этом наборе данных 99,84% всех наблюдений составляют отрицательные случаи. Это классический

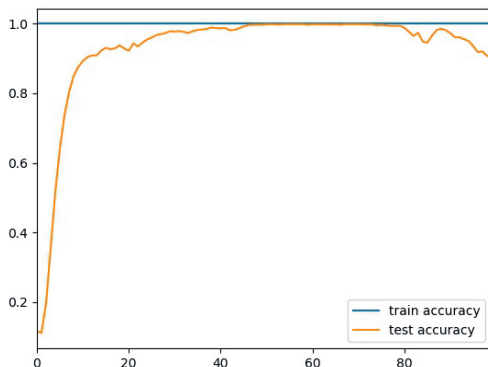


Рис. 33. Схема разметки 10–2, Вариант сети 2–2, нормальное обучение

пример проблемы смещенных классов, выражающийся, в частности, в том, что применяя примитивную модель, всегда предсказывающую 0, мы получим точность равную 99,84%. Естественно, этот показатель не является репрезентативным.

Второй вариант наблюдается в случае, когда дискриминатор по вариативности значительно превосходит генератор (рисунок 34). В таком случае, несмотря на состязательность обучения, слабовариативный генератор не способен оказать значительного давления на дискриминатор. Вследствие этого процесса, точность распознавания образцов дискриминаторов быстро падает и не способна восстановиться, что хорошо видно на рисунке 34.

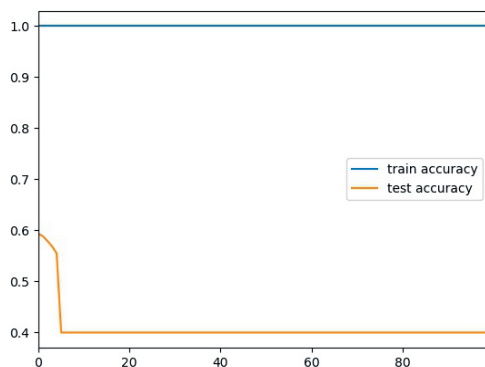


Рис. 34. Схема разметки 10–0, Вариант сети 0–3, коллапс модели

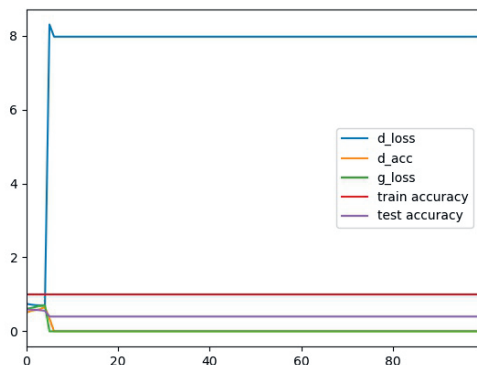


Рис. 35. Динамика остальных параметров при коллапсе модели

Довольно распространено неустойчивое поведение модели, характеризующееся сложной динамикой тестовой ошибки (рисунок 35), когда за период обучения динамика основных показателей состязательной сети несколько раз меняет свое направление. Это говорит о расхождении процесса обучения в принципе, и о неустойчивом характере состязания, что не позволяет ожидать от таких моделей надежных предсказаний.

Следует отметить, что, как и во многих моделях машинного обучения, в ГСС заметна тенденция к переобучению. Однако парадоксальным образом она больше проявляется в менее вариативных моделях (рисунок 37). Это можно объяснить тем, при малой доле

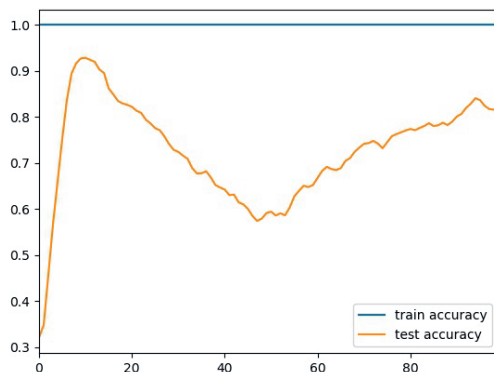


Рис. 36. Схема разметки 10–10. Вариант сети 1–3, неустойчивое обучение

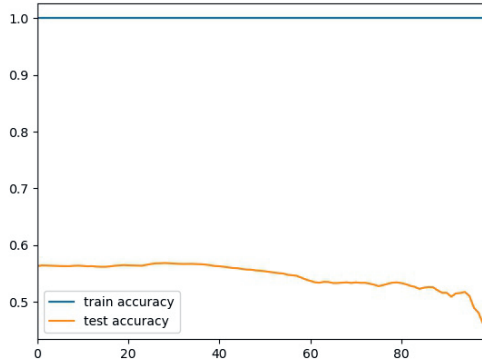


Рис. 37. Схема разметки 30–0. Вариант сети 0–1, переобучение

вариации генератора вся сеть пребывает в меньшем пространстве возможных образов, что эффективно снижает размерность набора данных, что губительно сказывается на динамике предсказательной способности модели в целом.

Для анализа результатов в более-менее приемлемом случае обучения взглянем на график, показывающий динамику ошибки генератора и дискриминатора (рисунок 38). На этом графике нас интересует две линии — динамика ошибки генератора и соотношение ошибки дискриминатора с пороговым значением 0,5.

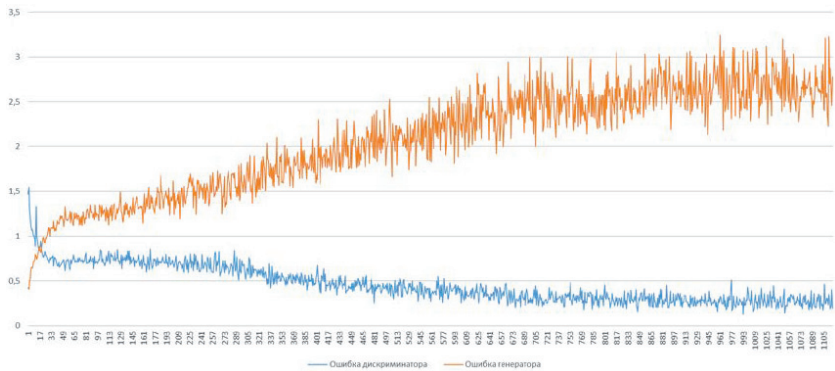


Рис. 38. Значение ошибки дискриминатора и генератора в каждой эпохе обучения

В идеальном случае, при обучения генеративных состязательных сетей мы должны видеть плавное снижение ошибки генератора со стремлением к нулю сверху, а также стремление ошибки дискриминатора значение 0,5 снизу. В классическом подходе нас интересует конечное значение ошибки генератора, так как состязательное обучение используется для тренировки генератора для воспроизведения правдоподобных образцов.

В нашем случае задача обратна классической, мы используем генератор для обучения дискриминатора способности распознать заданный шаблон. Соответственно, нас не удовлетворяет классическая динамика стремления ошибки дискриминатора к 0,5, т. е. ситуация случайного дискриминирования при низкой ошибке генератора. Наша задача сводится к попытке заставить дискриминатор снижать ошибку неопределенно долго. Одновременно с этим мы хотим иметь как можно более низкую ошибку генератора, так как дискриминировать образцы, далекие от натуральных нецелесообразно для данной задачи.

По графику данного обучения мы видим, что с увеличением количества шагов ошибка дискриминатора плавно уменьшается, и в конце обучения переходит пороговое значение.

Это, казалось бы, говорит об успешности обучения. Однако в то же самое время ошибка генератора растет неопределенно, т. е. обучение генератора расходится само по себе.

В таком случае о результативности обучения говорить не приходится, так как дискриминатор обучается на неправдоподобных шаблонах.

Однако при значениях ошибки дискриминатора, меньших 0,5, он оказывает обучающее давление на генератор, в чем, собственно, и заключается состязательность обучения.

В данном примере мы видим продолжительную динамику дискриминатора, что может дать нам надежду на то, что хорошо обученный дискриминатор позволит снизиться ошибки генератора.

В результате такого обучения тренд определяется очень «шумно» — слишком часто точки сменяют класс с трендовых на нетрендовые (рисунок 39).

Положительные отклонения в качестве модели, если и присутствуют, то незначительные, что может служить основанием для вывода о том, что среди протестированных нет модели, способной давать надежные прогнозы со статистически значимой точностью выше шанса.



Рис. 39. Результат обучения CGAN

3.5. Применение систем, основанных на обучении с подкреплением, для решения задач классификации и прогнозирования

Обучение с подкреплением (Reinforcement Learning, RL) является активно развивающейся областью машинного обучения, которая отличается от обычных методов тем, что вместо получения указаний совершать конкретные действия, агент сам взаимодействует с неким окружением, оценивает произошедшее и затем выбирает оптимальные для себя действия. Агент должен самостоятельно выбирать, какие действия следует предпринять, чтобы максимизировать не только сиюминутное вознаграждение, но и вознаграждения в будущем, которые могут быть получены в результате выбора текущего действия. Эти концепции исследования, признания важности отложенных наград и обучения методом «проб и ошибок» являются ключевыми в обучении с подкреплением и отличают их от других методов машинного обучения.

В 1998 г. Ричард С. Саттон и Эндрю Дж. Барто написали книгу, в которой содержится всеобъемлющее введение в теорию, историю и приложения известных на тот момент методов обучения с подкреплением. Осенью 2018 г. вышло второе издание [248].

Концепция финансового рынка естественным образом укладывается в задачу обучения с подкреплением. Участники рынка (в терминах обучения с подкреплением являющиеся агентами) совершают некоторые действия на финансовом рынке (в окружении) с целью

максимизации своей выгоды (награды). В связи с этим попытки использования RL-агентов для автоматизации действий на финансовом рынке совершались относительно давно и часто [249, 250, 251, 252, 253, 254, 255, 256, 257, 258].

Обучение с подкреплением не является обучением с учителем, поскольку агенту явно не предоставляются примеры хорошего и плохого поведения. Это делает обучение с подкреплением более подходящим для интерактивных задач, поскольку часто бывает трудно описать оптимальное поведение для таких типов проблем, и гораздо легче позволить агенту самому взаимодействовать с окружением. Обучение с подкреплением представляет собой общую задачу, которая может быть решена с помощью различных методов, и любая такая задача называется задачей обучения с подкреплением [180, 259].

Таким образом, задача обучения с подкреплением сводится к задаче взаимодействия с окружением для достижения поставленной цели. Объект, в процессе своего обучения взаимодействующий с окружением, а также являющийся в данном контексте лицом, принимающим решения (ЛПР), называется *агентом* (Agent). То, с чем взаимодействует агент, называется *окружением* (Environment). В ходе постоянного взаимодействия агента с окружением окружение на выбранное агентом действие отвечает новым *состоянием* (State, иногда Observation). Кроме состояния окружение на каждое *действие* (Action) агента порождает *награду* (Reward), которая является некоторым числовым значением, которое агент пытается научиться максимизировать за время своего обучения.

Взаимодействие агента с окружением разбито на последовательность дискретных *шагов* (Steps) во времени $t=0, 1, 2, \dots$. На каждом шаге агент получает от окружения состояние $S_t \in S$, где S – это множество всех допустимых состояний. Затем на основании полученного состояния агент выбирает действие $A_t \in A(S_t)$, где $A(S_t)$ – это множество всех допустимых действий из состояния S_t . После совершения выбранного действия, на следующем временном шаге $t+1$, агент получает соответствующую выбранному действию численную награду $R_{t+1} \in R$, где R – множество всех допустимых наград, и новое состояние S_{t+1} . На каждом шаге агент производит сопоставление текущего состояния и набора вероятностей выбора доступных действий, называемое *стратегией* (Policy) агента π . $\pi(a|s)$ вероятность выбора действия $A_t = a$ в состоянии $S_t = s$. Различные методы обучения с подкреплением определяют конкретный способ, с помощью которого

агент выстраивает свою стратегию на основании полученного опыта взаимодействия с окружением. Целью агента является построение такой стратегии, которая приводит к максимизации *кумулятивно-го* значения награды. Графически задача обучения с подкреплением изображена на рисунке 40.

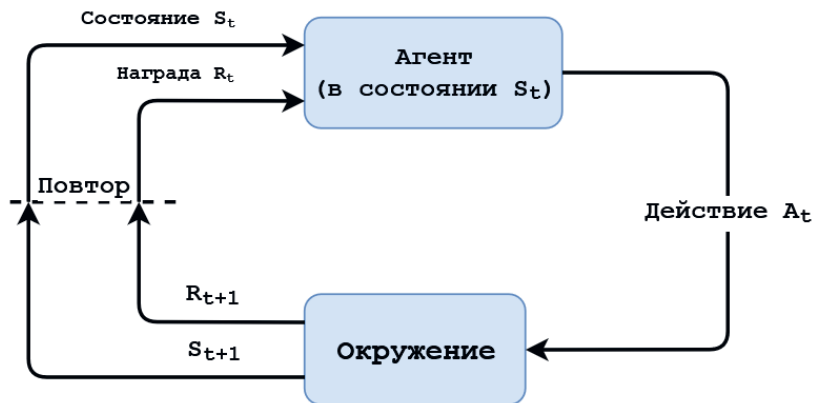


Рис. 40. Общий вид задачи обучения с подкреплением

Одним из основных отличий RL-задач является использование наград вместо функций потерь для формализации целей. Это является более естественным и гибким механизмом, что на практике позволяет описывать более обширный круг задач. К примеру, в задаче обучения модели человека бегу на соревновании, прошедшем в ходе конференции NIPS 2017, наградой служило пройденное расстояние в метрах [260]. В различных компьютерных играх наградой агенту часто выступает количество заработанных им очков [261, 262]. Или, например, в задаче удержания роботизированной рукой пишущей ручки награда может принимать всего два значения: 0, если заданное положение достигнуто, и -1 в противном случае [263].

Награды следует выдавать агенту таким образом, чтобы они не содержали в себе явным образом сведений о том, как агенту необходимо достигать поставленной цели. Также следует убедиться, что награды соответствуют общей цели, а не ее подцелям. Например, применительно к игре в шахматы, награды не должны отражать такие подцели как взятие большего количества вражеских фигур или занятие центра доски. В противном случае агент может выработать стратегию

достижения отдельных подцелей без достижения общей цели: завладеть большим количеством вражеских фигур, но проиграть партию. Таким образом, награды должны соответствовать тому, *что* должно быть выполнено агентом, а не *как*.

Если агент после шага t получает последовательно награды R_{t+1} , R_{t+2} , ..., то агенту необходимо максимизировать кумулятивную общую награду G_t , которую можно представить в виде суммы последующих наград до некоторого терминального шага T по формуле:

$$G_t = \sum_{i=0}^T R_{t+i+1}.$$

Однако данное представление функции награды возможно лишь при существовании в решаемой задаче последнего временного шага и приводит к разбиению всего процесса взаимодействия агента с окружением на *эпизоды* (Episodes). Эпизод – это последовательность действий от начального состояния до терминального. После терминального состояния происходит переход обратно в начальное состояние или выбор начального состояния из некоторого распределения. Задачи, в которых возможно выделить эпизоды, называются эпизодическими.

Тем не менее, многие задачи не являются эпизодическими, в них взаимодействие агента с окружением может продолжаться бесконечно. В таком случае они называются *континуальными*, и тогда $t = \infty$. Тогда сумма наград может достигать бесконечности, и на каждом шаге агент получает несравнимо маленькое значение. Для решения данной проблемы вводится *коэффициент дисконтирования*, и тогда агент вырабатывает стратегию максимизации ожидаемой кумулятивной дисконтированной награды согласно формуле:

$$G_t = \sum_{i=0}^{\infty} \gamma^i R_{t+i+1}.$$

Коэффициент $\gamma \in [0; 1)$. Если, то агент отдает предпочтение сиюминутным наградам. При $\gamma \rightarrow 1$ агент становится более «дальновидным», и для него становятся одинаково ценными как сиюминутные награды, так и отложенные во времени. При соблюдении условия, что награды ограничены, ряд сходится к конечному значению.

Определение состояния в задаче обучения с подкреплением является крайне важным этапом для ее успешного решения. Если состояние не содержит в себе достаточно информации, агент не сможет осуществить полностью обоснованный выбор действия и использовать все ранее накопленные знания о текущей ситуации. С другой

стороны, следует избегать ситуации, когда в состоянии заложено больше информации, чем агенту положено знать в текущий момент времени. Таким образом, агент не должен знать все важное для него во все моменты времени, но ожидается, что он может вспомнить все важное из своего опыта [180].

Математически это эквивалентно *марковскому свойству* (Markov Property), которое обычно используется в стохастических процессах. Можно сказать, что при фиксированном текущем значении будущие значения не зависят от прошлых значений процесса. Применительно к задачам обучения с подкреплением это означает, что награда и состояние на шаге $t+1$ должны зависеть только от состояния и выбранного действия на шаге t , а не от всей истории состояний и действий:

$$\begin{aligned}\mathbb{P}(R_{t+1} = r, S_{t+1} = s' \mid S_0, A_0, R_1, \dots, R_t, S_t, A_t) \\ = \mathbb{P}(R_{t+1} = r, S_{t+1} = s' \mid S_t, A_t).\end{aligned}$$

Марковское свойство для задач обучения с подкреплением важно, поскольку оно позволяет рассчитать следующее состояние и ожидаемую награду на основании только текущего состояния и действия [180, 264].

Задача обучения с подкреплением, удовлетворяющая марковскому свойству, может рассматриваться как *марковский процесс принятия решений* (Markov Decision Process, MDP). Для конечного MDP, т. е. такого MDP, у которого множества состояний и действий конечны, вероятность перехода из состояния s в состояние s' при выборе действия a вычисляется как:

$$p_a(s, s') = \mathbb{P}(S_{t+1} = s' \mid S_t = s, A_t = a).$$

При этом ожидаемая награда при переходе из состояния s в s' с вероятностью $p_a(s, s')$ рассчитывается по формуле:

$$r_a(s, s') = \mathbb{E}(R_{t+1} \mid S_{t+1} = s', S_t = s, A_t = a).$$

Агент постепенно получает все большие награды на каждом шаге, активно обучаясь с подкреплением. В каждом состоянии агент выбирает действие, которое приводит к другому состоянию и соответствующей награде. Поведение агента можно описать простым вопросом: «Какое действие необходимо выполнить в каждом состоянии?». Поведение транслируется в стратегию агента, т. е. сопоставление, определяющее, какое действие необходимо выполнить на каждом ша-

ге. Стратегия может быть детерминированной, когда выполнение действия гарантированно $\pi(S) = A_i$, или вероятностной, когда рассматривается вероятность выполнения действия $\pi(A_i|S) = p_i, i \in [0; 1]$.

Для оценки и улучшения стратегии вводятся *функции ценности*, которые присваивают каждому состоянию (или паре «состояние–действие») некоторое числовое значение.

Функция ценности (состояния) ((State) Value Function) оценивает стратегию в заданном состоянии s следующим образом:

$$V_{\pi}(s) = \mathbb{E}_{\pi}(G_t | S_t = s) = \mathbb{E}_{\pi}(\sum_{i=0}^{\infty} \gamma^i R_{t+i+1} | S_t = s).$$

Таким образом, функция ценности равна ожидаемой награде при условии, что агент продолжит на каждом будущем шаге придерживаться стратегии π . Для задач относительно небольшой размерности функция $V_{\pi}(s)$ может быть вычислена напрямую через вероятности переходов и ожидаемых наград в случае наличия этой модели переходов. Если модель переходов неизвестна, то функцию можно вычислить через взаимодействие с окружением методом Монте-Карло (Monte Carlo Method) при условии, что число находжений в каждом состоянии будет стремиться к бесконечности.

В случае вероятностной стратегии получается *уравнение Беллмана* (Bellman Equation) для $V_{\pi}(s)$:

$$V_{\pi}(s) = \sum_a \pi(a | s) \sum_{s'} p_a(s, s') [r_a(s, s') + \gamma V_{\pi}(s')].$$

Функция качества (Q -функция, Quality Function, Q-function) или *функция ценности действия* (Action Value Function) оценивает стратегию в заданном состоянии s для действия a следующим образом:

$$\begin{aligned} Q_{\pi}(s, a) &= \mathbb{E}_{\pi}(G_t | S_t = s, A_t = a) \\ &= \mathbb{E}_{\pi}(\sum_{i=0}^{\infty} \gamma^i R_{t+i+1} | S_t = s, A_t = a). \end{aligned}$$

Таким образом, функция ценности $Q_{\pi}(s, a)$ равна ожидаемой награде при условии, что агент в состоянии s совершит действие a , а затем продолжит на каждом будущем шаге придерживаться стратегии π .

Решением задачи обучения с подкреплением является нахождение такой стратегии, следуя которой, агент получит наибольшую кумулятивную дисконтированную награду. Оптимальной стратегией π^* называется стратегия, которая по сравнению с остальными стратегиями принесет агенту наибольшую ожидаемую награду.

При рассмотрении оптимальной стратегии π^* значение функции ценности $V_{\pi^*}(s)$ равно значению Q -функции $Q_{\pi^*}(s, a)$ при выборе оптимального действия:

$$V_{\pi^*}(s) = Q_{\pi^*}(s, a) = \max_{\pi} Q_{\pi}(s, a).$$

Для задач с большой размерностью пространств или, что важнее, для задач с континуальными пространствами состояний или действий невозможно посетить каждое состояние достаточное количество раз, чтобы рассчитать оценку значения функции. В связи с этим используется аппроксимация функций ценности и качества.

Некоторое время назад *RL*-задачи успешно решались табличным методом. Таблица естественным образом складывалась из пар состояние–действие. Эти значения в таблице обновлялись после каждого шага алгоритма, превращая сложные задачи в большие деревья решений, часто неприменимые на практике из-за ограничений по памяти и времени. В частности, когда задача имеет пространство состояний большой размерности, данная проблема носит название проклятия размерности [265].

Сейчас на смену таблицам пришли современные методы машинного обучения. Особый успех имеют различные конфигурации искусственных нейронных сетей [266]. Несмотря на то, что современные методы победили проклятие размерности, перед ними возникают новые трудности, такие как переобучение, вопрос инициализации параметров и другие.

На данный момент для вычисления оптимальной стратегии одним из наиболее популярных методов является *TD-обучение* (Temporal Difference Learning, TD-Learning) [248], которое объединяет в себе идеи динамического программирования (Dynamic Programming) [267] и методов Монте-Карло. *TD*-алгоритмы способны обучаться непосредственно на опыте взаимодействия агента с окружением, без какой-либо его модели. Однако в отличие от методов Монте-Карло, где обновление функции ценности происходит по достижению терминального состояния, *TD*-обучение не требует завершения эпизода и производит обновление функции ценности на каждом шаге, что может быть использовано в задачах, где вообще отсутствует разбиение на эпизоды. Ценой за возможность обновлять оценки на каждом шаге служит то, что для обновления используются другие оценочные значения, как и в динамическом программировании. В связи с тем, что *TD*-методы используют каждый шаг для обучения, нет не-

обходимости игнорировать или дисконтировать какие-либо эпизоды, как в методах Монте-Карло.

Наиболее простой разновидностью TD -методов является алгоритм $TD(0)$, который можно представить следующей формулой обновления функции ценности:

$$V(S_t) \leftarrow V(S_t) + \alpha[R_{t+1} + \gamma V(S_{t+1}) - V(S_t)],$$

где α – скорость обучения (Learning Rate).

Дальнейшим развитием метода является алгоритм $TD(\lambda)$. В нем, в отличие от $TD(0)$, значения обновляются не только у предыдущего состояния, а у всех состояний в *следе приемлемости* (Eligibility Trace). След приемлемости – это специальная временная структура, в которую записываются факты наступления событий, например, посещение какого-либо состояния или совершение какого-либо действия [248]. Значение λ выбирается из отрезка $[0; 1]$, что соответствует *нахождению баланса между смещением и дисперсией* (Bias and Variance Tradeoff). При $\lambda = 0$ алгоритм является одношаговым $TD(0)$, который обладает меньшей дисперсией, но большим смещением из-за оценки V_π последующего состояния. При $\lambda = 1$ алгоритм становится эквивалентным методу Монте-Карло. В этом случае след приемлемости заканчивается терминальным состоянием и содержит большое количество значений, что приводит к большей дисперсии, но избавляет от смещенности оценки.

Двумя особыми разновидностями TD -методов являются $SARSA$ и Q -обучение.

$SARSA$ (акроним от State _{t} , Action _{t} , Reward _{$t+1$} , State _{$t+1$} , Action _{$t+1$}) представляет собой TD -алгоритм с интегрированной оценкой качества стратегий (on-policy), что означает постоянное оценивание $Q_\pi(s, a)$ для стратегии поведения агента π и одновременное изменение π в соответствии со значением $Q_\pi(s, a)$:

$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \alpha[R_{t+1} + \gamma Q(S_{t+1}, A_{t+1}) - Q(S_t, A_t)].$$

Существует также разновидность алгоритма $SARSA(\lambda)$, в котором используются следы приемлемости.

Q -обучение (Q-learning) является TD -алгоритмом с разделенной оценкой качества стратегий (off-policy), что означает, что функция $Q_\pi(s, a)$ по-прежнему аппроксимирует оптимальную $Q_{\pi^*}(s, a)$, однако применяемая стратегия выбора действий может не совпадать с выбираемой стратегией [268]. Как и для $SARSA$, в Q -обучении воз-

можно применение следов приемлемости, такие алгоритмы носят название $Q(\lambda)$. Наиболее простой одношаговый алгоритм Q -обучения выражается следующей формулой:

$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \alpha [R_{t+1} + \gamma \max_a Q(S_{t+1}, a) - Q(S_t, A_t)].$$

Наиболее успешной попыткой объединить традиционное обучение с подкреплением и машинное обучение можно считать алгоритм *DQN* (Deep Q-Network), который был придуман в лаборатории Google DeepMind [261, 269]. Он заключается в том, что Q -функция аппроксимируется глубокой искусственной нейронной сетью $Q(s, a, \Theta_i)$, где Θ_i — это параметры (веса) нейросети на i -й итерации, а в качестве функции ошибки для тренировки сети выступает среднеквадратическая ошибка между текущей оценкой Q -функции и ее рассчитанным целевым значением. Затем на его основе были разработаны такие алгоритмы, как *Double DQN* (DDQN) [270], *Prioritized DDQN* [271], *Dueling DDQN* [272], *Noisy DQN* [273], *Distributional DQN* [274], *Rainbow* [275].

Авторами был разработан алгоритм для предсказания будущих моментов смены состояний рынка на основе обучения с подкреплением.

В наших экспериментах каждый раз мы сравнивали эффективность выработанной агентом стратегии с двумя другими стратегиями.

Первая стратегия заключалась в выборе действия на каждом шаге *случайным образом* (Random Strategy) [276]. Статистики по полученной прибыли (Profit) и длине занимаемых позиций (Trade Duration) на тестовой подвыборке приведены в таблице 7. А общая прибыль данной стратегии, полученная по всем финансовым инструментам из тестовой подвыборки, составила 12329,43 USD.

Из таблицы 7 видно, что несмотря на то, что стратегия случайного выбора действий позволяет получить общую положительную прибыль, среднее значение прибыли за одну сделку находится в окрестности нуля при значении среднеквадратического отклонения около 34 USD. Также интересным замечанием может быть то, что средняя длина занимаемой позиции составляет менее недели.

В качестве типичного примера стратегии случайного выбора действий на рисунке 41 приведены сгенерированные сигналы входов и выходов в позиции за 2015 г. (мы приводим только фрагмент за 2015 г., а не весь временной период, по причине огромного количества сгенерированных сигналов и невозможности их корректного изображения статичным рисунком на бумаге) для акций промышлен-

Таблица 7
Результаты Random Strategy

Statistic	Profit	Trade Duration
count	76220,000000	76220,0000000
mean	0,161761	3,862792
std	34,270007	3,404495
min	-2275,000000	1,000000
25%	-1,000000	1,000000
50%	0,000000	3,000000
75%	1,000000	5,000000
max	1169,000000	37,000000



Рис. 41. Результаты стратегии случайного выбора для TCFP за 2015 г.

ленной группы Thales (TCFP), выпускающей информационные системы для авиакосмической и военной отрасли.

Маркерами синего цвета обозначены сгенерированные сигналы входа (указывающая вверх стрелка) и выхода (указывающая вниз стрелка) в длинные позиции. Маркерами фиолетового цвета обозначены сгенерированные сигналы входа (указывающая вверх стрелка) и выхода (указывающая вниз стрелка) в короткие позиции.

Второй стратегией, с которой производилось сравнение, была стратегия покупки и удержания (Buy and Hold, B & H) [277, 278]. Как следует из названия, данная стратегия заключается в занятии длинной позиции в начале и ее закрытии в конце отведенного пери-

ода. Стратегия В & Н строится на представлении о том, что глобальный долгосрочный тренд, развивающийся на финансовом рынке, всегда является восходящим. Хорошая торговая стратегия должна по крайней мере превосходить стратегию В & Н.

Статистики стратегии В & Н по полученной прибыли и длине занимаемых позиций на тестовой подвыборке приведены в таблице 8. Общая прибыль, полученная при помощи данной стратегии по всем финансовым инструментам тестовой подвыборки, составила 51573,27 USD, что примерно в четыре раза больше, чем прибыль, полученная по стратегии случайного выбора действий.

Таблица 8
Результаты В & Н

Statistic	Profit	Trade Duration
count	180,000000	180,000000
mean	286,518141	2249,961111
std	910,849000	500,327003
min	-2639,266771	94,000000
25%	4,570060	2376,000000
50%	37,164215	2390,000000
75%	122,107815	2405,000000
max	5405,000000	2426,000000

Из таблицы 8 следует, что средняя прибыль за одну совершенную сделку уже равна не нулю, как при стратегии случайного выбора действий, а 287 USD. Однако значение среднего квадратического отклонения является очень большим и составляет около 911 USD, что свидетельствует о том, что В & Н стратегия не является универсальной и на многих финансовых инструментах из тестовой подвыборки привела к отрицательным значениям прибыли. Несовпадение минимального и максимального значения длины занимаемой позиции объясняется тем, что для некоторых финансовых инструментов история наблюдений была доступна за более короткий период времени, чем для остальных.

Несмотря на большое количество проведенных экспериментов, агенту так и не удалось превзойти результат стратегии В & Н. Даже

при установке комиссии за открытие и закрытие позиций в настройках симулятора равной нулю, наибольший размер прибыли, который агент смог заработать на тестовой подвыборке, равен 37892,38 USD. Средняя прибыль, получаемая на одном финансовом инструменте равна около 23,32 USD при значении среднеквадратического отклонения равном 253,91 USD, что отражено в таблице 9.

Таблица 9
Результат лучшей стратегии агента

Statistic	Profit	Trade Duration
count	1625,000000	1625,000000
mean	23,318385	227,068308
std	253,906495	179,743405
min	-3081,000000	50,000000
25%	-4,830000	98,000000
50%	1,770000	173,000000
75%	17,000000	299,000000
max	2960,000000	1362,000000

Однако стоит отметить, что получаемая агентом прибыль зачастую превосходила прибыль, полученную с помощью стратегии случайного выбора действий, в 2–3 раза. Полученные результаты согласуются с результатами, полученными авторами [251], где агент также не смог превзойти В & Н стратегию на данных индекса Russell 2000.

Также стоит отметить, что агент добивается положительных результатов, выстраивая свою стратегию почти только на занятии длинных позиций. В качестве иллюстрации данного факта на рисунке 42 приведены сгенерированные стратегией агента сигналы входов и выходов в позиции для акций промышленной группы Thales (TCFP).

Кроме этого, вырабатываемая агентом стратегия является сильно зависимой от значений гиперпараметров модели. Небольшое изменение гиперпараметра в ту или иную сторону влечет за собой кардинальное изменение в стратегии агента, вплоть до получения отрицательной прибыли порядка – 15000 USD.

Таким образом, модель является нестабильной, что подтверждается графиком кривой обучения, приведенном на рисунке 43, на ко-

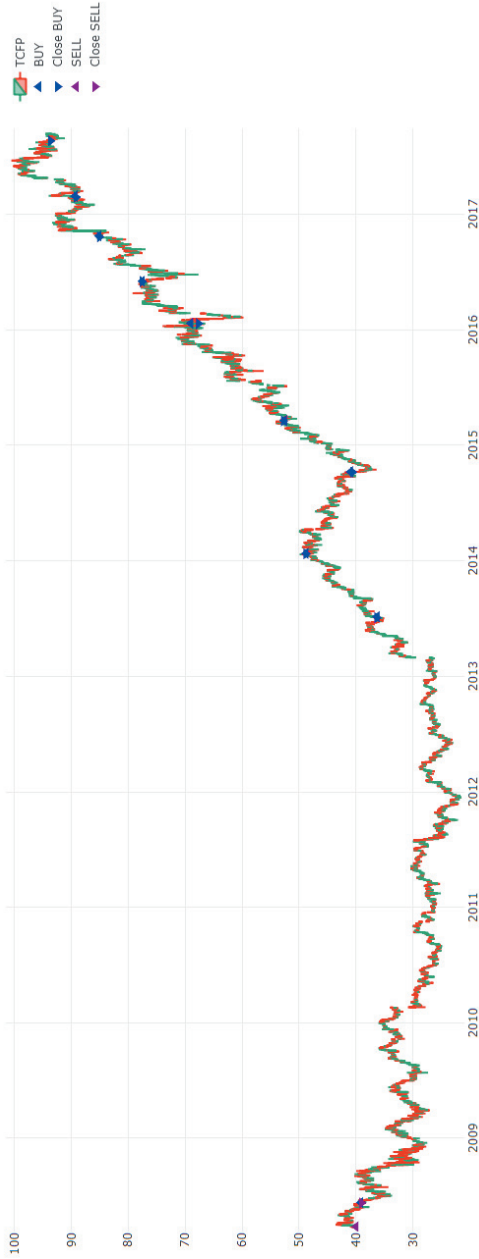


Рис. 42. Результаты наилучшей стратегии агента для TCFR

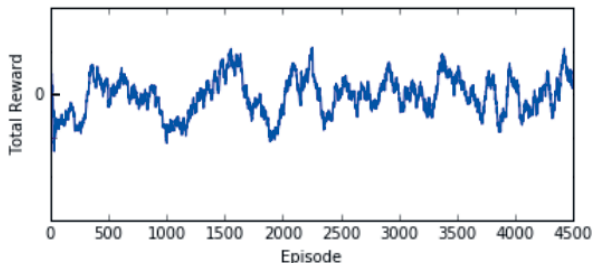


Рис. 43. График усредненной по 100 последним шагам прибыли в процессе обучения агента

тором видно, что усредненная по 100 последним шагам обучения прибыль, которая является функцией награды для агента, не сходится к какому-либо положительному значению, а флуктуирует в районе нуля. Более наглядно данный процесс можно изобразить, увеличив в несколько раз число эпизодов обучения агента (рисунок 44).

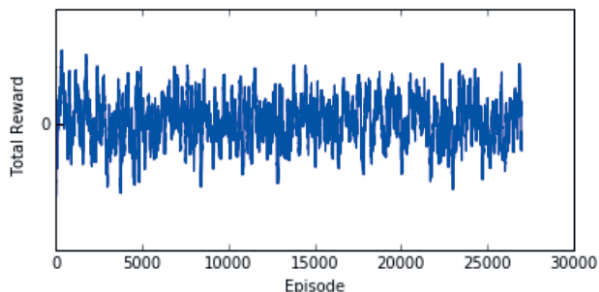


Рис. 44. График усредненной по 100 последним шагам прибыли в процессе обучения агента при увеличении эпизодов обучения

Глава 4

ЦИФРОВЫЕ ПЛАТФОРМЫ И МАРКЕТПЛЕЙСЫ

4.1. Цифровые платформы как тренд корпоративной автоматизации

Развитие систем корпоративной автоматизации претерпело несколько этапов в своем развитии. На начальном этапе (начавшегося в 1970-х годах) роль информационных технологий (ИТ) в организациях сводилась к автоматизации планирования поставок материалов, необходимых для производства. Класс систем, предназначенных для таких расчетов, назывался MRP (Material Requirements Planning), а сами системы устанавливались на больших электронно-вычислительных машинах (ЭВМ), под которые выделялись отдельные помещения. Это было время вычислительных центров на предприятиях. Следующий этап в корпоративной автоматизации начался в связи с появлением персональных компьютеров и локальных вычислительных сетей в конце 1980-х годов. Именно на этом этапе сложилась существующая сетевая структура информационных систем масштаба предприятий. Персональные компьютеры позволили вовлечь в процесс работы с электронной информацией не только операторов ЭВМ, но и всех сотрудников. Это привело к появлению многочисленных программ для автоматизации бухгалтерского и складского учета, офисной работы и т. п.

Хаотичное развитие автоматизации в организациях сделало востребованным появлением единых многофункциональных программных решений, получивших название ERP (Enterprise Resource Plan-

ning) систем. Компании, которые продвигали решения в этом классе систем (SAP, Oracle и др.), фактически стали международными монополистами на рынке корпоративной автоматизации. Но эпоха ERP систем была дважды на грани окончания. Сначала это случилось в начале нового столетия, когда выяснилось, что для организаций важна не только автоматизация учета и планирования внутренних ресурсов, но необходимо автоматизировать работу с внешними контрагентами: с клиентами, с поставщиками, с партнерами. Начали появляться новые классы систем управления отношениями с клиентами CRM (Customer Relationship Management), управления цепочками поставок SCM (Supply Chain Management) и т. п. Впрочем, поставщики ERP систем быстро сориентировались и включили новые функциональности в свои продукты, хотя и потеряли часть рынка корпоративной автоматизации. Gartner даже ввел термин для такого класса систем – ERP II.

Второй кризис ERP подхода к автоматизации организаций начался во второй половине первого десятилетия текущего века, когда выяснилось, что функциональность отдельных подсистем ERP решений все чаще не устраивала ИТ-службы, и необходимо использовать другие решения. В результате этого в информационной экосистеме, по крайней мере крупных предприятий, число приложений от разных поставщиков росло, что опять как и в 1990-е годы превращало ИТ ландшафт в «зоопарк». Впрочем и здесь решение было найдено – получили распространения так называемые сервисные шины предприятия, которые были призваны интегрировать различные приложения за счет обмена микросервисами. Но, судя по всему, сегодня наступает очередной кризис ERP систем, и возможно новый этап в корпоративной автоматизации, связанный с появлением цифровых платформ, которые вообще разрушают цельность информационных систем организаций.

Конец 2000-х гг. был ознаменован широким распространением мобильных устройств, подключенных к сети Интернет, как среди населения, так и в деятельности организаций. Доля мобильных устройств сегодня существенно превышает долю всей компьютерной техники. Так по данным Gartner мировое производство вычислительного оборудования в 2017 г. составило 665 млрд долл., при этом объем продаж только смартфонов составил 479 млрд долл. – 72%! Такой взрывной рост мобильности связан с успехами в области радиосвязи, позволившие владельцам мобильных устройств пользоваться программными сервисами без установки громоздких приложений

на оборудование. Если раньше внедрение ИТ было в основном сосредоточено на корпоративном рынке, то с появлением смартфонов, нетбуков и планшетов, информационные технологии пришли в каждый дом. Стремительное развитие электронных сервисов для населения не только позволило качеству таких сервисов догнать корпоративные ИТ, но и перегнать их: сегодня, например, почтовые сервисы и сервисы социальных коммуникаций гораздо более развиты, чем аналогичные сервисы на предприятиях.

Тотальная цифровизация корпоративного рынка и населения привело к своего рода фазовому переходу, когда на рынке стали создаваться совершенно новые цифровые сервисы, которые объединяли предприятия целой отрасли и их клиентов. Это явление получило название цифровой агрегации, а информационные системы, используемые для такой агрегации, получили название цифровых платформ. Так, например, сервис Booking.com объединил информационные сервисы большинства отелей мира на единой веб-платформе, предоставив клиентам возможность выбора лучшего отеля. Фактически Booking.com заменил часть функциональности CRM систем всех отелей сразу, став своего рода посредником между отелями и их будущими постояльцами. Однако появление таких цифровых посредников приводит и к появлению совершенно новых сервисов, которых раньше вообще не было — в случае Booking.com таким сервисом стало рейтингование отелей самими клиентами — именно этот сервис предоставляет интерес для тех, кто ищет подходящий отель. Можно сказать, что цифровизация ведет к существенному изменению существующих рынков, именно такое изменение и принято сегодня называть цифровой трансформацией [279].

В отличие от корпоративных информационных систем, в которых управление внутренними ресурсами, даже несмотря на новый подход ERP/II, все-таки является доминирующим, цифровые платформы предназначены исключительно для коммуникаций различных участников. Специалисты Московской школы управления Сколково дают такое определение цифровым платформам: «Цифровая платформа — сложная информационная система, обеспечивающая специфический способ выполнения определенной функции и открытая для использования клиентами и партнерами, включая разработчиков приложений, мерчантов и агентов. Платформа может быть использована напрямую или же через приложения, созданные на ее основе ее владельцем или третьими лицами» [280]. Создание цифровых плат-

форм фактически меняет экономику [281], отбирая у организаций те или иные функциональности. Так, например, цифровые агрегаторы такси заменили своим сервисом многочисленные колл-центры, которые были у таксопарков. Компания Амазон стала международным провайдером услуг по хранению данных и одновременно маркетплейсом для интернет-торговли. Уже развернулась нешуточная конкурентная борьба на рынке цифровой агрегации, где соперничают как международные гиганты из США и Китая, так и локальные игроки.

Цифровые платформы в корпорациях

Но роль цифровых платформ заключается не только в цифровой агрегации отдельных сервисов организаций. Сегодня выясняется, что платформенный подход необходимо использовать и на уровне самого предприятия. Первыми это поняли банки, которые оказались в авангарде цифровой трансформации экономики. Поскольку в основном услуги финансовых организаций носят информационный характер, именно финансовые сервисы стали предоставляться клиентам на условиях самообслуживания, через использование тех самых мобильных устройств и компьютеров. Если еще 10 лет назад личный кабинет в сети Интернет имели только особо продвинутые клиенты банков, то сегодня практически каждый человек, имеющий смартфон, пользуется возможностью провести оплату или перевод денег через мобильное устройство. Перевод сервисов на самообслуживание приводит к сокращению персонала финансовой организации и к снижению маржинальности (напр., услуги по переводу денег сегодня многие банки оказывают бесплатно), что делает финансовый рынок еще более конкурентным. В свою очередь конкуренция приводит к тому, что банки ищут все новые и новые сервисы, которые они могут предложить своим клиентам, причем уже не только финансовые: продажа авиабилетов, недвижимости, скидок в магазинах и т. п. тоже предоставляются сегодня банками. В такой же ситуации находятся и операторы связи: стоимость их услуг в России практически не повысилась, несмотря на двукратный рост валюты (а оборудование их в основном импортное), и они вынуждены сегодня предлагать клиентам медийные сервисы, различные услуги доставки, сервисы системной интеграции и даже клининговые услуги.

Конкурентный «шторм», в который неожиданно попали сегодня финансовые организации и операторы телекоммуникацион-

ных услуг, выявил и недостатки развития прошлых этапов корпоративной автоматизации — оказалось, что использование единой жесткой системы (как это предполагает ERP подход) мешает адаптации организации к быстро меняющемуся рынку. Сервисные шины предприятия тоже не помогают, поскольку они поддерживают все алгоритмы интеграции приложений, их также сложно быстро перестроить под новые сервисы как и единую ERP систему. Именно поэтому преимущества получают те организации, которые свою информационную систему сразу затачивают под новые сервисы. Феномен успеха банка Тинькофф связан именно с тем, что у этого банка не было старой и сложной системы, которую трудно перестраивать под новые требования бизнеса, связанные с предоставлением цифровых сервисов. Сегодня многие банки начинают пересматривать свое отношение к системам автоматизации, понимая, что помимо надежности и производительности они должны обеспечивать легкую адаптацию новых функций. И здесь как раз очень кстати может пригодиться платформенный подход, который предполагает не замену старого ПО, а его интеграцию с единой цифровой платформой, обеспечивающей едиными классификаторами, справочниками и внутренними сервисами.

Цифровые платформы сегодня внедряются не только в сугубо цифровой сфере, но и в индустриальных отраслях, в рамках общего тренда Индустрии 4.0 и внедрения Интернета вещей IoT (Internet of Things). В частности один из лидеров машиностроения компания Siemens предложила открытую операционную систему MindSphere, которая позволяет подключать разнообразное оборудование, имеющее выход в Интернет. Со своей программной платформой (Predix), позволяющей собирать и анализировать данные с промышленного оборудования, вышла на рынок и компания General Electric. Известная на рынке строительной техники компания Caterpillar предложила систему управления оборудованием под названием Cat Connect. Система Cat Connect позволяет управлять всем оборудованием на строительной площадке как единым целым, повышая эффективность всего процесса строительства.

Цифровые платформы в государственных организациях

В настоящее время все больше становятся востребованными информационные системы, выходящие за рамки отдельных организа-

Цифровые платформы и маркетплейсы

ций, которые получили название – информационные платформы. По всей видимости, и в области управления кадрами государственной гражданской службы настало время создание единой цифровой платформы человеческого интеллектуального капитала. Для начала в такую платформу стоит включать не всех госслужащих, а только тех, которые проявил способности в области трансформации органов власти – в части внедрения новых цифровых технологий, в области управления (включая проектами). Можно было бы говорить о, своего рода, глобальном «кадровом резерве» государственной службы, куда бы попадали служащие с магистерским образованием и выше, либо служащие, которые уже проявили себя в управлении и информатизации.

Цифровая платформа кадров государственной гражданской службы (ЦПКГГС) не должна подменять собой кадровые программы, установленные в различных организациях госслужбы, но должна интегрировать их с точки зрения единых моделей компетенций, общего трекинга карьеры государственного гражданского служащего. На рисунке 45 показано, что ЦПКГГС будет общей платформы для всех организаций в государственной гражданской службе. При этом в качестве классификаторов в той или иной организации будут использоваться лишь часть общей национальной модели компетенций, а интеграция персональных данных госслужащих позволит отслеживать карьерный путь служащего, независимо от коли-

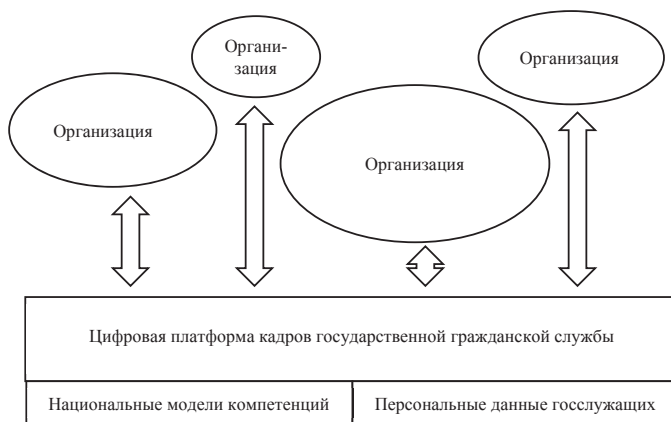


Рис. 45. Цифровая платформа государственной гражданской службы как интеграционное решение

чества его переходов с одной работы на другую. Более того, единая цифровая платформа позволит создавать отдельные сервисы для госслужащих, включая образовательные услуги и сервисы профессиональных сообществ, которые позволят более точно измерять и повышать эксклюзивные компетенции сотрудников. Можно сказать, что внесение сотрудника в ЦПКГГС является своего рода «путевкой» в карьерном росте, он становится «видимым» для других организаций, у него появляются возможности роста даже в том случае, если в его организации такого роста пока быть не может. В будущем можно будет интегрировать ЦПКГГС с едиными базами данных образовательных учреждений. Использование аналитических инструментов позволит выявлять особенности в работе с кадрами в госслужбе в целом, принимать стратегические решения о ее развитии.

4.2. Цифровые платформы в управлении цепями поставок

Сферы и области применения технологии блокчейна в современной экономике постоянно расширяются. Крупнейшие транснациональные корпорации, в том числе сервисные, торговые и логистические, уделяют приоритетное внимание цифровым технологиям и осуществляют капитальные вложения в соответствующие разработки, в том числе в создание системы блокчейна и ее внедрение при построении цепи поставок.

Целью данного исследования было выявление основных эффективных сфер применения блокчейна в логистической деятельности, определение возможностей применения системы распределенного реестра при управлении цепями поставок и обоснование необходимости легализации термина «блокчейн» на национальном и международном уровнях.

К таким компаниям можно отнести производителя и поставщика аппаратного и программного обеспечения IBM, торговую сеть Walmart, лидера интернет-торговли Amazon, транснациональные компании-производители Unilever и Nestlé, транспортно-экспедиционную компанию UPS и др.

Одним из первых блокчейн в торговле применил мировой лидер розничной торговли компания Walmart. В 2016 г. она запустила пилотный проект по использованию технологии блокчейна при поставках свинины из Китая. Кроме самой Walmart, в реализации проекта приняли участие Университет Циньхуа в Пекине и корпорация

IBM. Сотрудничество было направлено на совершенствование системы отслеживания товаров, перевозимых из Китая. Следующим примером применения технологии блокчейна торговой компанией Walmart можно назвать поставку манго из Мексики.

В 2018 г. базирующаяся во Франции международная розничная торговая сетевая компания Carrefour объявила о расширении использования технологии блокчейна [282]. Покупателям будет предложена возможность отслеживать всю цепочку движения товаров животноводства и овощей. Считывая QR-код, размещенной на продукции, с помощью смартфона потребитель сможет узнать все подробности происхождения товара, его транспортирования и хранения.

Проанализировав практику применения блокчейна в различных сферах деятельности на основе данных, представленных на сайте CHAINSTEP (консалтинговая компания, осуществляющая разработку и информационную поддержку блокчейн-проектов), можно сделать вывод, что именно управление цепями поставок и логистика — сферы деятельности, где блокчейн имеет наибольший потенциал для развития. Так, по данным на 10 марта 2018 г., из 453 кейсов внедрения блокчейн технологии, на блок «управление цепями поставок, транспорт, логистика» приходится 61 проект. При этом наибольшее число проектов все-таки пока приходится на сферу финансов — 116 [283].

Ряд транснациональных компаний (IBM, Walmart, Nestlé, Dole, Tyson Foods, Kroger и др.) договорились в 2017 г. о сотрудничестве, направленном на изучение, использование и дальнейшее развитие технологии блокчейна для отслеживания движения продуктов питания на протяжении всей цепочки поставок.

В настоящее время осуществляется также разработка и апробирование торговой блокчейн-платформы для мировой логистической индустрии, разработанной компанией IBM совместно с мировым лидером в сфере контейнерных перевозок, датской компанией Maersk. Цель — создание цифровой торговой платформы, построенной на открытых стандартах и предназначенной для использования всей глобальной экосистемой логистики.

Блокчейн при управлении цепями поставок в большей степени применяется в следующих основных функциональных областях: документирование логистических операций, заключение договоров и обеспечение исполнения прочих многосторонних соглашений, отслеживание грузов, финансовое обеспечение логистического про-

цесса и т. п. В 2017–2018 гг. пилотные проекты применения технологии блокчейна были запущены в области поставок продуктов питания, в первую очередь, таких групп товаров как овощи и фрукты, продукция животноводства, рыба и морепродукты. По мнению автора, эта тенденция сохранится в ближайшие два года, но одновременно будет расширяться сфера эффективного применения технологии блокчейн при управлении цепями поставок.

Проанализируем преимущества, которые предоставляет блокчейн для участников цепи поставок.

Технология блокчейна положительно влияет на решение ключевых задач управления цепями поставок, в том числе на надежность, устойчивость и гибкость цепи поставок [284].

Анализируя применение технологии блокчейна при управлении цепями поставок, можно выделить следующие преимущества:

- снижение логистических издержек во всей цепочке поставок;
- сокращение системных рисков в операционной деятельности [285];
- повышение сохранности товаров, снижение уровня потерь в процессе доставки и хранения товаров;
- обеспечение прозрачности и достоверности информации о производителях (поставщиках) товаров и процессе товародвижения;
- обеспечение полной анонимности всех сделок в случае реализации закрытых решений [286];
- достижение высокой степени гибкости цепочки поставок;
- высокая степень защиты информационных потоков внутри логистической системы;
- обеспечение неограниченного времени хранения разрешительных и сопроводительных цифровых документов, включая сертификаты, лицензии, подтверждения уплаты акцизов и др. [287];
- повышение скорости таможенной обработки товаров при осуществлении международной торговли;
- обеспечение более тесных взаимоотношений и координации деятельности всех участников цепи поставок;
- обеспечение прав потребителей за счет предоставления полной и неоспоримой информации о происхождении товаров, поступивших в розничную сеть;
- снижение доли «серого» импорта;

Цифровые платформы и маркетплейсы

- рост возможностей для развития «справедливой» торговли (fair trade) и «зеленой» логистики;
- уменьшение числа заболеваний, связанных с потреблением некачественной, зараженной пищевой продукции;
- применение стратегии динамического ценообразования в сфере услуг, которая может быть выстроена на основе реальных данных, к примеру с учетом реальной скорости доставки, климатических условий и т. д.

Для субъектов рынка, которые незаконным образом производят товары, блокчейн станет препятствием для дальнейшей недобросовестной предпринимательской деятельности.

Применение технологии блокчейна позволит также упростить процедуру государственного администрирования цепи поставок и будет способствовать реализации государственной политики, направленной на снижение уровня контрафактной продукции и повышение качества товаров, поступающих в розничную сеть. К примеру, по данным агентства «Интерфакс», в 2018 г. «в России будет введена система полного поштучного учета алкогольной продукции в ЕГАИС (единая государственная автоматизированная информационная система учета алкоголя) <...> Система поштучного учета использует механизмы технологии блокчейна: движение каждой единицы продукции создает цепочку блоков неизменных транзакций, хранящихся в дублирующих центрах обработки данных в Росалкогольрегулировании и Гознаке» [288].

В исследовании, проведенном компанией DHL [289], подчеркивается, что развитие мировой логистической системы связано с развитием экономики совместного потребления (sharing economy). В рамках совместного потребления именно применение технологии блокчейна и внедрение смарт-контрактов позволяет обеспечить прозрачность сделок.

Результаты исследования показывают, что зарубежные и российские эксперты схоже оценивают препятствия для внедрения технологии блокчейна. Основными препятствиями они считают:

- невозможность оценить экономическую эффективность внедрения системы блокчейна в деятельность организаций, а также непосредственно в процесс организации товародвижения;
- непонимание сущности блокчейна и особенностей его внедрения лицами, принимающими решения;

Глава 4

- необходимость реинжиниринга бизнес-процессов;
- необходимость дополнительных инвестиций в проектирование новых цепей поставок;
- недостаточное число квалифицированных кадров в сфере блокчейна;
- необходимость интеграции внедряемой технологии с существующими технологическими платформами.

Дальнейшие исследования в сфере использования блокчейна при управлении цепями поставок будут носить междисциплинарный характер и охватывать следующие области: информационные технологии, менеджмент, логистику, мировую и региональную экономику. Особое внимание также следует уделить «токенизации» объектов гражданского права [290] и регулированию договорных отношений в процессе осуществления поставки товаров.

На государственном уровне одним из значимых вопросов, который требует проведения дополнительного исследования, является применение блокчейна государственными, в первую очередь таможенными, органами.

Развитию блокчейна в России будет способствовать формирование системы нормативно-правового регулирования в этой области. Необходимо решение вопроса юридической значимости записей, подтвержденных (задокументированных) с помощью технологии блокчейн. Должна быть решена проблема легализации блокчейн-реестров, определена ответственность субъектов рынков за включение в распределенный реестр неполной и/или недостоверной информации, а также четко прописаны вопросы контроля операций, совершенных на основе блокчейна. Должны быть определены субъекты, ответственные за хранение информации. Следует определить механизм защиты информации, составляющей коммерческую тайну, персональную информацию и т. п. Также должен быть решен вопрос соотношения действующих правовых норм, к примеру, в отношении хранения, удаления и модификации информации, с условиями и возможностями, которые представляет внедрение технологии блокчейна в предпринимательскую деятельность.

Остается открытым и вопрос о заинтересованности самих субъектов предпринимательской деятельности использовать технологию блокчейна и становиться полностью «прозрачными» в части управления цепями поставок.

Предприниматели и органы власти постоянно находят новые пути и способы применения блокчейн-технологии в современной экономической системе. Блокчейн в управлении цепями поставок – технология, которая позволяет проводить транзакции между равноправными участниками единой сети в процессе управления материальными, информационными, финансовыми и сервисными потоками.

За последние два года ключевые проекты применения технологии блокчейна были реализованы в области поставок продуктов питания (овощи и фрукты, продукция животноводства, рыба и морепродукты). В ближайшие годы сфера эффективного применения технологии блокчейна при управлении цепями поставок расширится. Будут охвачены как поставки продовольственных, так и непродовольственных товаров, в первую очередь, это будет касаться товаров, по которым наблюдается высокая доля контрафактной и некачественной продукции.

Дальнейшие исследования в сфере блокчейна в управлении цепями поставок будут носить междисциплинарный характер и охватывать такие области научных исследований как информационные технологии, менеджмент, логистика и управление цепями поставок, финансы, право.

4.3. Межотраслевой маркетплейс для участников создания новых высокотехнологичных продуктов

Развитие промышленности и производительных сил показало несостоятельность монопроизводств применительно к выпуску высокотехнологичной продукции. Следует понимать, что в современном мире любое высокотехнологичное изделие, будь то автомобиль или компьютер, это по сути скомплексированный (собранный) под неким брендом набор комплектующих от разных производителей, требующий высококвалифицированного сервисного обслуживания на всех этапах его жизненного цикла. При этом значительная часть современных высокотехнологичных конкурентных продуктов все чаще создается на стыке различных, зачастую не взаимосвязанных, областей знаний, технологий, производств и т. д, что требует расширения и повышения эффективности производственной кооперации.

Вместе с тем, в нашей стране исторически сложилось так, что многие производственные цепочки развивались в условиях ведомствен-

ных и отраслевых ограничений. В ряде случаев это создает труднопреодолимые барьеры в проектировании, изготовлении и эксплуатации технически сложных изделий. В связи с этим, одним из наиболее проблемных вопросов в выпуске современной высокотехнологичной продукции выступает межотраслевая разобщенность, в том числе в вопросах распространения и использования информации о товарных предложениях комплектующих, имеющих широкий спектр применения, а также конструкторских, технологических, производственных, сервисных, финансовых и иных возможностях потенциальных участников производственной кооперации. При этом очевидно, что чем шире выбор комплектующих, тем больше возможностей по производству новых технически сложных продуктов (изделий) и их сервисному обслуживанию. А чем шире сервисная сеть и доступ к комплектующим, тем выше конкурентоспособность технически сложного продукта (далее ТСП).

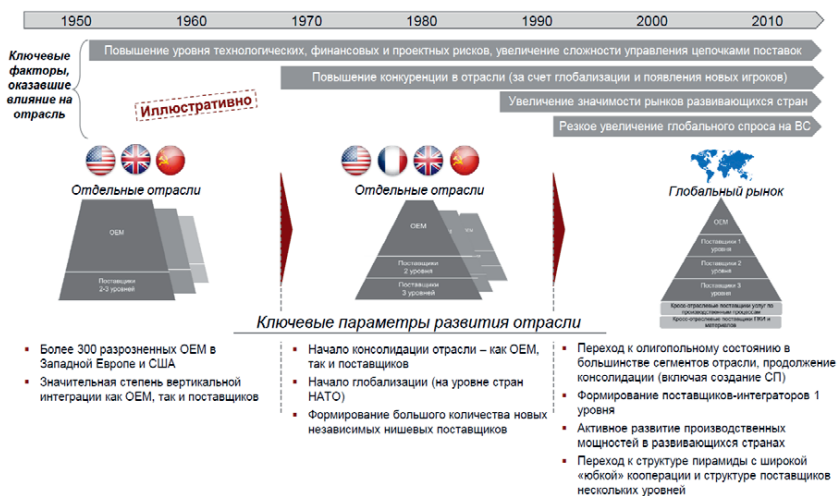


Рис. 46. Динамика развития кооперационных связей

Не менее значимой является проблема централизованного обобщения и систематизации (каталогизации) информации о конструкторских, технологических и других возможностях выпуска компонентов и комплектующих, используемых в процессе создания новых высокотехнологичных конкурентных продуктов. Помимо этого,

у большинства инициаторов выпуска новых высокотехнологичных продуктов ключевым вопросом является не только поиск и подбор необходимых компонентов и комплектующих будущего изделия, а в первую очередь выстраивание эффективных коммуникаций между производителями, сервисниками, логистами и инвесторами. Без публичной, самонаполняемой, структурированной и систематизированной базы данных, обобщающей информацию о существующих и потенциальных проектах производства ТСП, производителей (разработчиках) комплектующих, поставщиках сервисных услуг, логистах и инвесторах способных и готовых к взаимодействию выстроить производственные цепочки крайне затруднительно. Однако, на сегодняшний день такой базы данных не существует, а деятельность по поиску необходимых контрагентов и установлению кооперационных связей зачастую носит субъективный характер и не всегда охватывает весь спектр предложений.

Увязка решения проблемы выпуска ТСП с поиском поставщиков комплектующих, логистических и сервисных услуг, а также инвесторов могла бы позволить оптимизировать организацию производства высокотехнологичных продуктов. При этом эффективность организационной деятельности может быть достигнута благодаря сетевой интеграции производственных, конструкторских, технологических, логистических финансовых и иных возможностей на единой независимой онлайн-площадке.

Таким образом, решению указанных проблем мог бы способствовать специализированный информационный инструмент – маркетплейс⁹, позволяющий на этапе разработки и производства сложного высокотехнологичного продукта осуществлять оперативный поиск подходящих комплектующих либо на основе задекларированных возможностей напрямую адресовать техзадание их производителям. При этом инициатор производства нового изделия также мог бы на основе использования единой платформы параллельно осуществлять онлайн-поиск инвестора, выстраивать логистику и сис-

9 Маркетплейс (*англ.* online marketplace, online e-commerce marketplace) – платформа электронной коммерции, предоставляющая информацию о продукте или услуге третьих лиц, чьи операции обрабатываются оператором маркетплейса. Маркетплейс представляет собой форму многоканальной электронной коммерции и позволяет оптимизировать процесс предоставления продуктов или услуг. URL: <https://ru.wikipedia.org/wiki/Маркетплейс>.

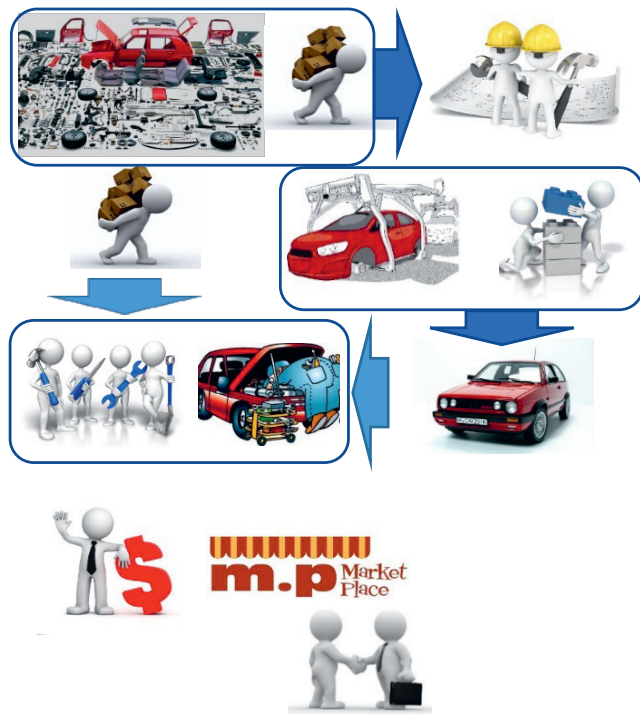


Рис. 47. Маркетплейс

тему сервисного обслуживания будущего продукта на всех этапах его жизненного цикла. Функциональность платформы повысилась бы, если все пользователи в зависимости от изменения жизненной ситуации и производственных потребностей (возможностей) могли бы менять свою роль, представив себя, например, вместо инициатора производства ТСП в качестве поставщика комплектующих либо сервисных услуг и т. д. Следует отметить, что с учетом данной возможности рассматриваемая онлайн-площадка могла бы позволить выстраивать отношения не только в рамках полного цикла организации производства нового продукта, но и между отдельными его участниками уже существующей другой производственной цепочки при решении частных задач (модернизации ТСП, замены комплектующих, расширения сервисного обслуживания, изменения логистики или доп. инвестирования и т. п.). Это помогло бы сделать

Цифровые платформы и маркетплейсы

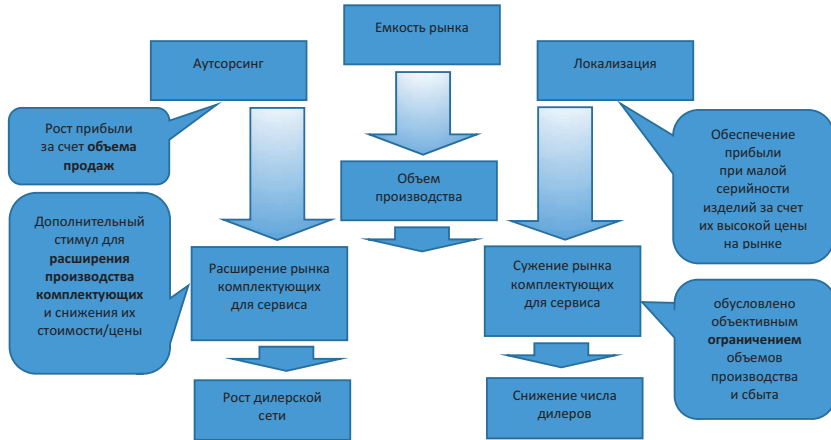


Рис. 48. Зависимости экономических показателей при использовании разных подходов в организации производства

существующие производственные процессы более гибкими и устойчивыми к внешним и внутренним проблемам. Это наиболее важно при импортозамещении и диверсификации оборонных производств.

Также важную роль в создании подобного рода площадки будет играть маркетинговая составляющая всех бизнес-процессов. Ключевым параметром в этом случае будет выступать репутация, формируемая автоматически (т. е. объективно и прозрачно) на основе результатов проведенных сделок, отзывов контрагентов, характеристик представленного предложения и др. параметров. При этом репутация будет выступать важнейшей характеристикой для инвестора, рассматривающего варианты объектов инвестиций. Помимо этого, репутация, а также иные характеристики пользователей платформы могут стать объектом востребованных аналитических исследований, публикация которых на портале будет способствовать продвижению площадки и бизнеса ее пользователей, формированию собственного комьюнити и создаст дополнительный источник дохода для владельцев маркетплейса.

Проект маркетплейса может строиться на следующих базовых положениях.

Любой технически сложный продукт (ТСП) — это результат промышленной кооперации, который получен в ходе сборки комплектующих и материалов, производимых на других предприятиях.

Чем шире выбор комплектующих, тем больше возможностей по производству новых технически сложных продуктов (изделий) и их сервисному обслуживанию.

Чем шире сервисная сеть и проще доступ к комплектующим, тем выше конкурентоспособность ТСП.

Чем прозрачней и надежней связи промышленной кооперации, тем проще поиск объекта инвестирования.

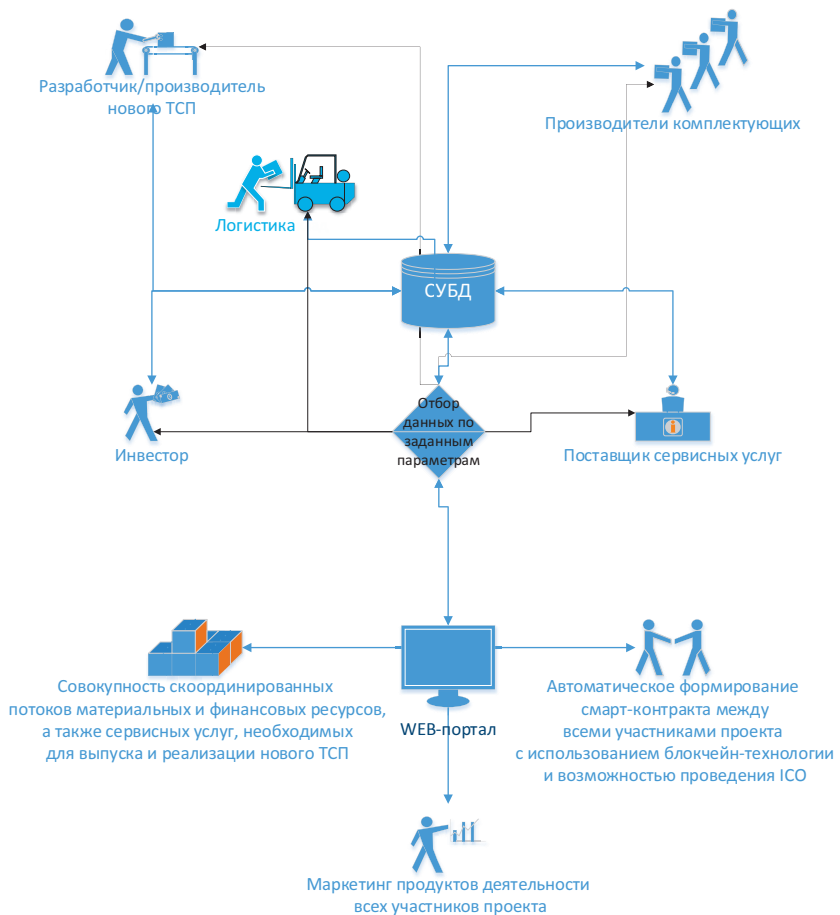


Рис. 49. Схема реализации и функционирования маркетплейса (вариант)

Цифровые платформы и маркетплейсы

Цели и задачи создания маркетплейса могут заключаться в создании интернет-платформы, позволяющей в режиме онлайн на основе рыночных механизмов формировать кооперационные связи между: производителями (разработчиками) ТСП, производителями комплектующих, сервисными службами, логистическими предприятиями и инвесторами.

Для достижения этой цели необходимо решить следующие технические задачи:

- разработать дружелюбный интерфейс интернет-платформы (маркетплейса);
- разработать или адаптировать существующую блокчейн-платформу, обеспечивающую реализацию описанной бизнес-логики;
- разработать алгоритмы наполнения, управления и хранения базы данных, формируемой при функционировании маркетплейса.

Детальное описание маркетплейса

Функционал межотраслевого маркетплейса с интерактивным представлением:

- проектов технически сложных продуктов (изделий);
- комплектующих к ним и конструкторско-технологических возможностей, производящих их промышленных предприятий;
- сервисных услуг;
- логистических схем и финансовых инструментов, необходимых для создания новых высокотехнологичных продуктов и завязывания кооперационных связей.

Общая концепция реализации проекта может быть аналогична в части касающейся Яндекс.Маркет, Авито, Amazon, eBay и т. п. на основе использования интуитивно понятного дружелюбного интерфейса. Одним из ключевых отличий от указанных платформ будет являться то, что пользователь может одновременно представлять и выбирать в качестве товарных предложений не только готовые изделия, но и компетенции на онлайн-бирже заказов и приглашений по участию в проектах по выпуску готовых изделий, комплектующих, сервисному обслуживанию, логистике и инвестированию. Ключевым отличием от большинства существующих торговых и инвестиционных площадок могла бы стать возможность заключения соглашений между контрагентами с использованием смарт-контракта платформы блокчейна.

Отдельными самостоятельными разделами могут быть представлены личные страницы пользователей, форум (чат) между участниками проектов, арбитраж, аналитика и др.

Наполнение базы данных может быть осуществлено следующими способами:

- на добровольной основе с активным использованием маркетинговых инструментов, PR-продвижения и рекламы;
- с использованием встроенного парсера;
- директивно с использованием административного ресурса заинтересованных министерств и ведомств.

При наличии государственной поддержки возможно создание закрытого сегмента платформы для работы предприятий ОПК и опасных производств.

Следует отметить, что адаптация маркетплейса к российским реалиям ведения бизнеса и его правового регулирования, по взгляду автора, не представляет большой сложности. С учетом требований Федеральных законов РФ¹⁰ к базовой модели маркетплейса посредством инструментов администрирования либо добавления необходимых плагинов к движку сервиса можно добавлять необходимый функционал, либо устанавливать требуемые ограничения. Например, формирование закрытого сегмента для оборонно-промышленного комплекса.

Маркетплейс предполагает группировку зарегистрированных участников по следующим ролям, которые они могут менять по необходимости:

- производитель (разработчик) сложного высокотехнологичного продукта (изделия);
- производитель комплектующих для сложных высокотехнологичных продуктов (изделий);
- поставщик сервисных услуг;
- поставщик логистических услуг;
- инвестор.

10 Федеральный закон № 44-ФЗ от 05.04.2013 (с изм. 01.10.2018) «О контрактной системе в сфере закупок товаров, работ, услуг для обеспечения государственных и муниципальных нужд», № 223-ФЗ (изм. от 01.07.2018) «О закупках товаров, работ, услуг отдельными видами юридических лиц», № 135-ФЗ от 26.07.2006 г. «О защите конкуренции», № 63-ФЗ от 06.04.11 (ред. 02.07.13) «Об электронной подписи» и др.

Цифровые платформы и маркетплейсы



Разработчик/производитель
нового ТСП

Имеет возможность:

- Задекларировать создание нового ТСП с перечнем необходимых комплектующих и указанием их ТТХ
- Осуществить поиск и подбор по заданным параметрам необходимые комплектующие либо сделать заявку на их производство
- Подобрать сервисные службы, способные осуществлять поддержку ТСП на всех этапах его жизненного цикла
- Найти инвестора или источник финансирования выпуска ТСП, в т. ч. путем проведения ICO
- Автоматически сформировать смарт-контракт с контрагентами



Производители комплектующих

Имеет возможность:

- Представить образцы своей продукции, которая может выступить в качестве комплектующих к ТСП
- Задекларировать свои компетенции и возможности по разработке и производству комплектующих по определенным параметрам
- Осуществить поиск и подбор по заданным параметрам проекты по выпуску ТСП, при производстве которых могут быть востребованы производящиеся комплектующие либо компетенции по их выпуску
- Осуществить поиск сервисных служб, осуществляющих поддержку ТСП на всех этапах его жизненного цикла для поставки им своих комплектующих, которые могут быть использованы для ремонтных, регламентных и иных работ
- Найти инвестора или источник финансирования выпуска комплектующих к ТСП, в т. ч. путем проведения ICO
- Автоматически сформировать смарт-контракт с контрагентами



Поставщик сервисных услуг

Имеет возможность:

- Задекларировать готовность осуществления сервисного обслуживания ТСП на всех этапах его жизненного цикла
- Осуществить поиск и подбор по заданным параметрам производителя ТСП, нуждающегося в услугах сервисных служб
- Осуществить поиск по заданным параметрам комплектующих к ТСП, необходимым для сервисного обслуживания
- Найти партнера среди производителей комплектующих, готового поставлять свою продукцию для осуществления сервисного обслуживания ТСП
- Автоматически сформировать смарт-контракт с контрагентами



Инвестор

Имеет возможность:

- Задекларировать готовность инвестировать в производство ТСП, комплектующих либо сервис
- Осуществить поиск и подбор по заданным параметрам производителя ТСП, комплектующих либо сервисных служб, нуждающихся в инвестициях
- Инвестировать в ICO действующих проектов



Логистика

Имеет возможность:

- Задекларировать готовность осуществления доставки и хранения материальных ценностей
- Осуществить поиск и подбор по заданным параметрам производителя ТСП, комплектующих либо сервисной службы, нуждающейся в логистических услугах
- Автоматически сформировать смарт-контракт с контрагентами

Рис. 50. Распределение ролей и возможностей участников маркетплейса

Каждая категория зарегистрированных участников использует базовый функционал платформы, позволяющий найти необходимого контрагента по заданным характеристикам и уникальным параметрам искомого продукта (услуги), задекларированным компетенциям и производственным мощностям поставщика продукта (услуги), отраслевой, технологической, ведомственной и иной принадлежности участника или продукта (услуги), географическим параметрам, срокам исполнения, ценовым (финансовым) параметрам, рейтин-

гу, формируемому на основе отзывов других участников и др.; отправить контрагенту заявку с указанием точных параметров заказа, включая техническую документацию; автоматически сформировать на основе соответствующей бизнес-логики смарт-контракты между заинтересованными участниками проекта и обеспечить контроль их обязательного исполнения; при необходимости автоматически сформировать пакет документов, требующихся при выстраивании производственно-технологических цепочек (счетов-фактур, накладных и т. п.), предоставить возможность обмена сообщениями между контрагентами; при необходимости организовать арбитраж между участниками проекта.

Для реализации сделок, предполагающих поставки материальных ценностей и инвестирования, в рамках работы платформы может быть использована технология блокчейна и реализованы возможности смарт-контрактов. Основная идея применения данных технологий в рамках данного проекта это перевод сделок в цифровую среду, что упростит и ускорит выполнение обязанностей всех заинтересованных лиц, а также обеспечит определенный уровень надежности совершаемых сделок. Помимо этого, появляется возможность альтернативного финансирования проектов и их участников¹¹.

Следует отметить, что процесс адаптации технологии блокчейна к задачам маркетплейса определяет ряд требований, связанных с доступностью веб-портала и иных его сервисов для широкого круга пользователей. В частности, доступ к распределенному реестру должен осуществляться без использования сторонних приложений через общераспространенные браузеры. Немаловажным вопросом с точки зрения доступности, а также стоимости пользования ресурсом выступает организация вычислений в рамках достижения консенсуса и подтверждения транзакций в распределенном реестре. Наиболее распространенные блокчейн-платформы¹² основаны на использовании алгоритма Proof-of-Work, что создает необходимость майнинга. А данный процесс неизбежно обуславливает стоимость транзакции, соответственно участник маркетплейса вынужден в этом случае нести дополнительные издержки в рамках кооперации.

11 Данный вопрос требует дополнительного изучения и обоснования. Вместе с тем, реализация указанного функционала технически возможна и резко повышает привлекательность проекта в глазах потенциальных пользователей.

12 Эфириум, биткоин, а также их фреймворки.

С другой стороны, необходимость майнинга создает дополнительные затраты для оператора маркетплейса, что также негативно сказывается на его работе. С учетом данного обстоятельства наиболее целесообразным вариантом в этой ситуации будет блокчейн-решение исключающее майнинг.

Не менее важным моментом с точки зрения использования блокчейн технологии в работе маркетплейса является содержание и возможность корректировки смарт-контракта, а также механизмы контроля его исполнения.

С учетом особенностей такого явления как смарт-контракт следует учитывать, что предметом закрепляемых им договорных отношений могут выступать исключительно количественные параметры. С этой точки зрения, маркетплейс и возникающие на его основе отношения вполне укладываются в данную парадигму. Вместе с тем, сам смарт-контракт не гарантирует его физического исполнения. Ключевым звеном в этом плане может выступать служба доставки, через которую фактически может подтверждаться факт транзакции через отгрузку комплектующих. Однако, может оказаться не вполне решенным вопрос оценки качества доставленных материальных ценностей либо качества предоставленных услуг. Видимо в этом случае следует предусмотреть дополнительные инструменты, как встроенный инструмент арбитража, реализуемый группой подготовленных сотрудников. Данный вопрос может открыть новые перспективные направления исследований, в частности, в области применения искусственного интеллекта при решении данной непростой задачи. Также отдельным вопросом, требующим более глубокого исследования, может стать работа с базой данных и управление ей при функционировании маркетплейса, а также интеграция в сервис приложений, позволяющих сопрягать платформу маркетплейса с современными системами автоматического проектирования (САПР) и иными информационно-экономическими системами.

4.4. Интеллектуальная система мониторинга вовлеченности студентов на основе технологий интернета вещей

С 1970-х гг. информационные технологии активно используются для повышения качества образования. При этом давно обсуждаются преимущества, недостатки и возможности совершенствования традиционных аудиторных занятий.

Сегодняшние студенты «с рождения» используют различные вычислительные устройства, а в Интернете по большинству теоретических и практических учебных дисциплин студенты часто могут получить более полную, более точную и более актуальную информацию, чем рассказывают преподаватели на лекциях. При этом информация в Интернете часть и подается более эффективно и эффективно, чем в аудиториях.

Интересно ли студентам на занятиях? Успевают ли студенты за преподавателем или темп изложения является слишком быстрым либо слишком медленным? Доступно ли излагается материал? Насколько студенты вовлечены в учебный процесс на аудиторных занятиях? Эти вопросы в эпоху цифрового образования становятся важнейшими. Тем не менее, еще недавно контролировать уровень вовлеченности студентов было практически невозможно: например, только в московских учебных корпусах Финансового университета занятия идут каждый день с 8:30 по 22:00 более чем в 500 аудиториях.

С 1980-х гг. одной из ключевых тем в исследованиях стала тема анализа вовлеченности студентов в учебный процесс. В первую очередь, интерес к этой теме был обусловлен необходимостью снижения количества отчисляемых студентов. Проведенные опросы (см., напр.: [291]) показывали, что от 25 до 60% студентов постоянно скучают на занятиях и отвлекаются от учебного процесса. Поэтому анализ возможностей повышения уровня вовлеченности студентов уже достаточно давно является важной целью исследований в области оптимизации учебного процесса [292].

Сегодня управление уровнем вовлеченности студентов актуально и для традиционного аудиторного учебного процесса, и для массовых открытых онлайн-курсов, и для образовательных игр, симуляторов и тренажеров, интеллектуальных обучающих систем и т. д. [293, 294, 295, 296].

В настоящее время выделяются три формы вовлеченности: поведенческая, эмоциональная и когнитивная [297, 298, 299]. Под поведенческой вовлеченностью обычно понимается готовность студента участвовать в учебном процессе (например, посещать учебные занятия, выполнять учебные задания, выполнять и представлять учебные задания, следовать указаниям преподавателя). Эмоциональная вовлеченность описывает эмоциональное отношение студента к обучению (например, студент может хорошо справляться с заданиями, но ему при этом не обязательно интересно — у таких студентов

высокая поведенческая вовлеченность, но низкая эмоциональная вовлеченность. Когнитивная вовлеченность — это уровень использования студентами своих когнитивных способностей (внимания, памяти, творческого мышления и т. д.).

Необходимость повышения вовлеченности студентов привела к появлению методов измерения уровня вовлеченности, самыми популярными из которых сегодня относятся к трем категориям: самооценка своей вовлеченности самими студентами; внешнее наблюдение с использованием контрольных карт и последующим рейтингованием; автоматическое измерение уровня вовлеченности с использованием технических средств [300].

В российских исследованиях вовлеченности студентов доминирует использование самооценок для измерения поведенческой вовлеченности. Например, в работе [301] на основании анализа данных онлайн-опроса студентов трех вузов (1957 студентов национального исследовательского университета; 3938 студентов федерального университета и 521 студента регионального вуза) выделены четыре составляющих поведенческой вовлеченности: вовлеченность в работу на занятиях; вовлеченность в групповую работу; вовлеченность в учебную деятельность, выходящая за рамки требований преподавателя; пассивная вовлеченность.

В данной работе описывается опыт разработки и использования роботизированной интеллектуальной системы мониторинга вовлеченности студентов на основании анализа видеопотоков из аудиторий.

Информационные системы для автоматического измерения уровня вовлеченности с использованием технических средств используются на практике уже достаточно давно. Значительная часть популярных методик для автоматической оценки уровня вовлеченности основана на анализе скорости и точности выполнения студентами контрольных заданий [302, 303]. Например, случайные ответы на легкие вопросы или очень короткое время, затрачиваемое на выполнение заданий могут быть индикаторами слабой вовлеченности.

Другой класс популярных методик для автоматической оценки уровня вовлеченности основан на обработке данных, получаемых с различных электрофизиологических и нейрофизиологических датчиков [304, 305]. В нейронауке вовлеченность обычно измеряется через уровень возбуждения и уровень внимания, которые, в свою очередь, определенным образом зависят от данных электроэнцефа-

лограмм, электрокардиограмм, артериального давления, частоты сердечных сокращений, гальванической реакции кожи и т. д. Однако такие методики требуют обеспечения студентов специальными датчиками, что делает невозможным их масштабное применение для исследования вовлеченности большого количества студентов, например, в масштабах целого университета.

Третий класс методик автоматического распознавания вовлеченности, к которому относится и система, описываемая в данной работе, основан на использовании компьютерного зрения [306, 307, 308, 309, 310, 311]. Такие методики позволяют оценивать вовлеченность студента путем анализа положения и наклона головы, направления взгляда, положения тела, различных жестов и т. д.

Системы автоматического анализа вовлеченности, основанные на компьютерном зрении, могут быть использованы преподавателями в качестве обратной связи, позволяющей совершенствовать образовательные методики, в том числе в режиме реального времени. При этом возможно использование таких систем как в очном образовании, так и в дистанционном.

Администрация учебных заведений может использовать системы автоматического распознавания вовлеченности для корректирующих действий, направленных на снижение доли неуспевающих студентов, внося изменения в учебный процесс в реальном времени с целью предупреждения снижения уровня вовлеченности студентов.

Важнейшим преимуществом таких систем является то, что оценка уровня вовлеченности происходит ненавязчиво, без отвлечения внимания студентов на сам процесс измерения вовлеченности.

Использование методик автоматического распознавания вовлеченности позволяет получить большие объемы данных, в том числе изображений и видео, которые могут быть использованы для последующего обогащения кампусных информационных систем и интеллектуального анализа данных об учебном процессе и успеваемости студентов.

Цель работы — разработка облачного сервиса мониторинга вовлеченности студентов на занятиях на основании интеллектуального анализа видеопотоков, поступающих с видеокамер, размещенных в учебных аудиториях, и последующей агрегации усредненных данных по группам, курсам, направлениям подготовки, уровням образования, факультетам на интерактивных панелях визуализации.

При этом измерению подлежит *предполагаемая вовлеченность*, т. е. уровень вовлеченности студентов, оцениваемый внешними наблюдателями.

Система, которая на основании изображения студента определяет, является ли он в данный момент вовлеченным, строится на принципах машинного обучения. Вначале большое количество фотографий лиц студентов, сделанных видеокамерами в аудиториях, предъявляется экспертам, которые разделяют фотографии на два класса (вовлеченные студенты и невовлеченные студенты), затем на основании этих, размеченных экспертами, данных происходит обучение модели классификации, и в дальнейшем обученная модель классификации используется для того, чтобы предсказывать уровень вовлеченности студентов на фотографиях, которых ни эксперты, ни модель классификации ранее не видели.

В работе обсуждается методология разметки изображений студентов, построенная модель машинного обучения, а также подходы к визуализации усредненной информации на интерактивных панелях.

В отличие от известных результатов исследования, в которых вовлеченность измеряется на основании данных с видеокамер, размещенных на компьютерах и способных измерять уровень вовлеченности отдельных студентов на занятиях, проходящих в компьютерных классах, либо при дистанционной форме обучения (в том числе на массовых открытых онлайн-курсах), в данной работе предложена интеллектуальная система, в автоматическом режиме измеряющая вовлеченность студентов в масштабах целой образовательной организации. Это дает возможность ее использования и отдельными преподавателями для получения обратной связи об изменении уровня вовлеченности студентов во времени (как в течение занятия, так и в течение семестра).

Но самое главное, такая система может использоваться администрацией учебного заведения для получения оперативной обратной связи о динамике средней вовлеченности групп студентов в течение семестра. Это позволяет проводить сравнение изменения вовлеченности во времени по факультетам, курсам, направлениям подготовки, отдельным группам, отдельным студентам и преподавателям и оперативно проводить соответствующие корректирующие действия.

Кроме того, отличительной чертой системы, которая описывается в данной работе, является то, что в отличие от других извест-

ных систем данная система построена в форме облачного сервиса, который может быть использован для мониторинга вовлеченности сколь угодно больших групп студентов, эластично масштабируясь при изменении числа студентов. Такой сервис может быть использован одновременно несколькими образовательными организациями или даже в масштабах всей системы образования.

Объектами исследования выступают студенты, находящиеся на учебных занятиях, предметом исследования является динамика предполагаемой вовлеченности студентов в течение учебных занятий.

Для получения видеопотоков используются установленные в учебных аудиториях видеокамеры (предпочтительно использование поворотных видеокамер, меняющих направление фокуса с заданной периодичностью с помощью специального скрипта).

Сервис распознавания вовлеченности развернут в облаке *Microsoft Azure* и использует соответствующую подписку учебного заведения. Идентификация пользователей происходит на основе служб каталогов *Microsoft Azure Active Directory*, синхронизированных с локальными университетскими службами каталогов. Идентификация студентов происходит на основании фотографий из кампусной базы данных, а идентификация учебных дисциплин и преподавателей — на основании информации из локальной базы данных «Расписание».

Решение было разработано в Финансовом университете при Правительстве Российской Федерации с участием ООО «Зеробит»: команда преподавателей и студентов департамента анализа данных, принятия решений и финансовых технологий Финансового университета отвечала за разработку системы машинного обучения и системы визуализации результатов, сотрудники ООО «Зеробит» — за оптимизацию инфраструктурных решений, а также за различные алгоритмы обработки данных. Архитектура решения разрабатывалась совместно.

В настоящее время система проходит пилотную эксплуатацию в двух учебных корпусах Финансового университета, к ней подключены около 200 видеокамер, расположенных в аудиториях.

Архитектура облачного решения проиллюстрирована рисунком 51. В качестве устройств интернета вещей выступают датчики присутствия, освещенности и температуры, контроллеры отопления и освещения, а также поворотные видеокамеры, размещенные под потолком лекционных аудиторий, аудиторий для практических занятий и компьютерных классов.

Цифровые платформы и маркетплейсы

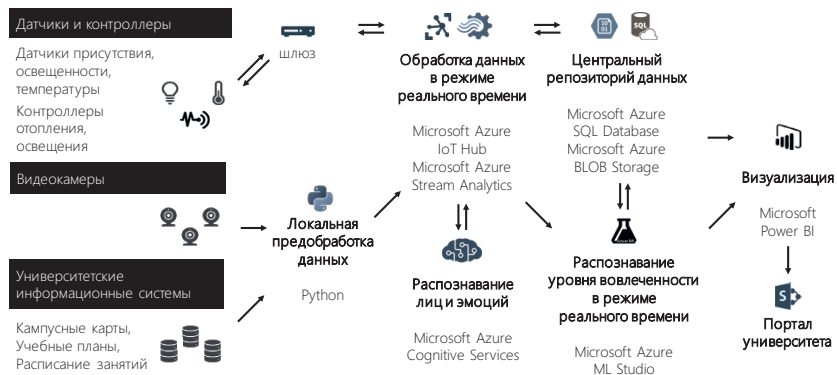


Рис. 51. Архитектура облачного сервиса мониторинга вовлеченности (подготовлено авторами)

Датчики и контроллеры через шлюз подключаются к концентратору интернета вещей *Microsoft Azure IoT Hub*, а видеопотоки, поступающие с видеокамер, перед отправкой в *IoT Hub* подвергаются локальной предварительной обработке: из видеопотоков с заданной периодичностью захватываются отдельные кадры, которые затем отправляются в концентратор интернета вещей. За локальную обработку отвечает один компьютер на каждой территории.

Данные о расписании занятий берутся в привязке к аудиториям, в которых размещены видеокамеры, из локальной информационной системы «Расписание» и включают в себя время начала и окончания занятия, номер аудитории, набор идентификаторов учебных групп (на практическом занятии это одна группа, а на лекции – несколько), идентификатор учебной дисциплины и идентификатор преподавателя.

При поступлении в *IoT Hub* изображения, содержащего снимок студентов, оно отправляется в *Microsoft Azure Cognitive Services* для распознавания лиц и эмоций студентов (предварительно в *Microsoft Azure Cognitive Services* заносятся фотографии студентов из кампусной базы данных), эмоций. Службы распознавания эмоций *Microsoft Azure Cognitive Services* возвращают в ответ для каждого лица, распознанного на изображении, распознанные значения возраста и пола; идентификатора студента; признаков наличия помады, очков, усов, бакенбардов, бороды; наклона и поворота головы; распознанных эмоций (*happiness* – счастье; *sadness* – печаль; *surprise* – удивление; *anger* –

гнев; *fear* – страх; *contempt* – презрение; *disgust* – отвращение; *neutral* – нейтральность), а также специальные точки, определяемые на лицах, – *facial landmarks* (рисунок 52).

Для каждого распознанного лица все названные признаки, полученные из служб распознавания эмоций *Microsoft Azure Cognitive Services*, а также время съемки кадра, идентификатор типа учебного занятия (лекция, практическое занятие, занятие в компьютерном классе), идентификатор учебной дисциплины и идентификатор преподавателя заносятся в базу данных *Microsoft Azure SQL Database*, а сами изображения сохраняются в хранилище *Microsoft Azure BLOB Storage*.

При появлении новой записи в таблице результатов распознавания лиц и эмоций базы данных *Microsoft Azure SQL Database* эта запись автоматически подается на вход веб-сервису, основанному на заранее обученной модели классификации в *Microsoft Azure Machine Learning Studio*, возвращающему для каждого лица вероятность его отнесения к классу вовлеченных студентов, и в таблице результатов распознавания лиц и эмоций базы данных заполняется соответствующее поле. Также в эту таблицу заносятся показания датчиков температуры и качества воздуха.

Обработка событий в режиме реального времени осуществляется при помощи службы *Microsoft Azure Stream Analytics*.

Еще один элемент системы – служба *Microsoft PowerBI* – с заданной периодичностью визуализирует в виде специальных отчетов сводные таблицы, основанные на результатах распознавания лиц и эмоций базы данных *Microsoft Azure SQL Database*, которые встраиваются в интерактивные панели на университетском портале.

Методология разметки изображений

Для разметки изображений экспертами, в ходе которой эксперты для каждого лица указывают, является ли данный студент, по их мнению, вовлеченным или нет, разработано специальное приложение, опубликованное в облаке *Microsoft Azure* (рисунок 53).

В качестве экспертов выступают преподаватели. Каждый преподаватель получает задание оценить вовлеченность для определенного количества распознанных лиц, при этом большая часть лиц автоматически выбирается с изображений, полученных с видеорекамера во время занятий, которые вел данный преподаватель (доля таких лиц является настраиваемым параметром, значение которого реко-

Цифровые платформы и маркетплейсы

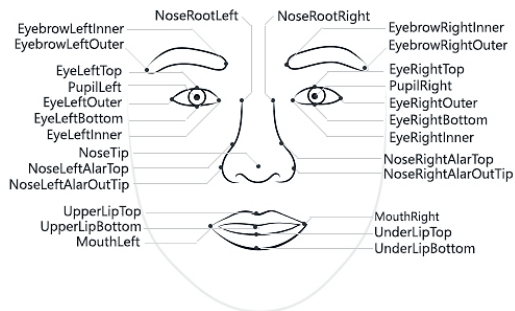


Рис. 52. Специальные точки (facial landmarks), определяемые на лицах службами *Microsoft Azure Cognitive Services* (источник: [312])

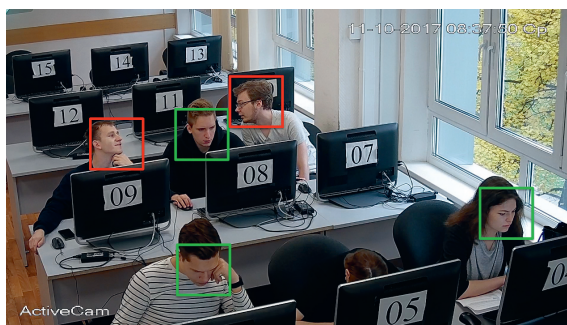


Рис. 53. Визуализация результатов разметки вовлеченности экспертами

мендуется устанавливать на уровне 90–95%), а оставшаяся часть лиц берется с кадров, сделанных во время занятий, которые вели другие преподаватели. Это необходимо для обеспечения адекватности оценок: если у какого-либо преподавателя оценки, сделанные им самим, не совпадают с оценками вовлеченности тех же студентов, сделанных другими преподавателями, это служит сигналом для предметного рассмотрения администратором разметки и возможного признания оценок неадекватными.

При этом у эксперта есть возможность не оценивать вовлеченность для отдельных фотографий в случаях, когда оценку вовлеченности студента провести невозможно или не нужно (например, когда для оценки эксперту представляется снимок, сделанный на перемене, либо если лица на изображении не видно).

Модель машинного обучения для классификации студентов по уровню вовлеченности

Наилучшие результаты классификации получены с помощью модели усиленных деревьев решений (*ADABOOST Two-Class Boosted Decision Tree*). Для этой модели были получены следующие метрики качества: доля правильных ответов (*Accuracy*) оказалась равна 84,8%, точность (*Precision*) равна 0.825, полнота (*Recall*) – 81,5%, *F*-мера (*F1 Score*) – 82,0%, площадь под *ROC*-кривой (*AUC*) – 91,2%. Данные результаты свидетельствуют о правомерности использования модели для распознавания уровня вовлеченности.

Среди факторов, оказывающих наибольшее положительное влияние на уровень вовлеченности, с помощью инструментария *Permutation Feature Importance* выделены следующие (в порядке убывания важности факторных признаков): наклон головы; распознанный возраст; поворот головы; уровень печали; уровень удивления; а также некоторые точки на лицах (*facial landmarks*).

Визуализация усредненной информации о динамике уровня индивидуальной и групповой вовлеченности студентов на интерактивных панелях

Интерактивная панель системы размещена на единственной странице, в верхнем меню которой пользователь системы выбирает период детализации отчетности: обзорная по семестрам или детальная по месяцам. Для каждого периода формируется детализированное меню:

Отображение информации о вовлеченности на любом уровне иерархии производится универсальным блоком отображения в виде дискретной цветовой шкалы для выбранного периода.

Изначально главная страница отображает верхние уровни – факультеты, информация о которых доступна пользователю (в модуле администрирования возможно разграничение доступа, например, сотрудникам ректората можно сделать доступной информацию о всех факультетах, сотрудникам деканата факультета – о всех направлениях подготовки и уровнях образования, которые реализуются на данном факультете, преподавателям и студентам – только об их занятиях; возможно поместить всю или часть информации в открытый доступ).

Дочерние элементы отображаются по требованию – клику на конкретный уровень. По умолчанию отображается текущий учебный

семестр, по желанию можно посмотреть статистику за предыдущие семестры. Клик по месяцу приводит к переходу на детализацию этого месяца. Каждый уровень раскрывается и сворачивается кликом.

В режиме детализации месяца дерево уровней выполнено в виде классического проводника; напротив каждой записи (факультет, направление подготовки, уровень образования, курс, группа, студент, преподаватель) визуализируется информация о вовлеченности для выбранного месяца с усредненными значениями, отображаемыми цветом.

Правая колонка показывает детальную информацию по выбранному пункту, как информационную (например, контактную информацию подразделения), так и статистическую аналитику.

Также в правой колонке отображаются снимки аудиторий, участвующей в контроле по выбранной дате, смена производится ротацией снимков дочерних элементов, входящих в состав данного уровня.

Фрагмент интерактивной панели визуализации представлен на рисунке 54.

Результаты пилотного использования облачного сервиса мониторинга вовлеченности студентов на занятиях на основании интеллектуального анализа видеопотоков демонстрируют достаточную степень адекватности для использования сервиса как администрацией образовательных организаций, так и преподавателями.

Измерение уровня вовлеченности студентов — одна из важнейших задач как для администрации образовательных организаций, так и для преподавателей.

При этом большинство инструментов, традиционно используемых для измерения уровня вовлеченности — самооценка вовлеченности самими студентами, внешнее наблюдение с использованием контрольных карт и последующим рейтингованием, а также оценка вовлеченности путем обработки данных, получаемых с различных электрофизиологических и нейрофизиологических датчиков, являются слишком сложными для измерения динамики вовлеченности всех студентов образовательной организации на постоянной основе.

В данной работе описан разработанный авторами облачный сервис мониторинга вовлеченности студентов на занятиях на основании интеллектуального анализа видеопотоков, поступающих с видеочамер, размещенных в учебных аудиториях, и последующей



Рис. 54. Фрагмент интерактивной панели

агрегации усредненных данных по группам, курсам, направлениям подготовки, уровням образования, факультетам на интерактивных панелях визуализации.

В результате пилотного использования сервиса выявлены ключевые факторы, определяющие вовлеченность, к которым относятся возраст, показатели положения, наклона и поворота головы, а также некоторые эмоции.

Данный облачный сервис не только может использоваться не только в различных университетах: путем незначительной трансформации данную архитектуру возможно использовать в банках, магазинах, предприятиях сферы обслуживания, финтех-компаниях, в которых важно измерение внимания или удовлетворенности клиентов.

Предварительные результаты данного раздела были опубликованы в статье [313].

4.5. Формализация экономических отношений в условиях развития автоматизации

Цифровые технологии преобразуют социальные и экономические отношения в сторону снижения прямого взаимодействия и личного контакта. Анонимность и одновременное расширение отношений до межкультурных (как в географической, так и возрастной плоскостях) приводит к глобальному снижению эффективности старых неформальных институтов. С другой стороны, формируемые в рамках межкультурной коммуникации сообщества, к примеру крипто-сообщество, взаимодействует на основе новых неформальных правил. Данные взаимодействия построены на основе высокой степени автоматизации и нуждаются в традиционных способах формализации в гораздо меньшей степени. Исходя из чего, можно допустить, что повышение степени автоматизации взаимодействия снижает необходимость его формализации в правовой форме, ибо уже являются формализованными.

В качестве преимуществ формализации институтов различные авторы [314] приводят приблизительно одни и те же доводы. Первым, из наиболее часто встречаемых, является довод о снижении транзакционных издержек, в частности расходов на анализ информации по контракту и агенту, контроль за исполнением контракта. Кроме того, формализации приписывают возможность решения проб-

лем безбилетника и дискриминации. Все эти преимущества все более эффективно реализуются через автоматизацию взаимодействия, без перехода в правовую формализацию.

По сути, несколько вариантов обеспечения эффективности применения правил, либо посредством правовой защиты, либо алгоритмически, с институциональной точки зрения, выявляют очень интересный феномен. Нам не удалось найти четкой классификации типов формализации институтов, но если отталкиваться от существующих, то большинство авторов склоняются к тому, что формализация происходит через применение правовых конструкций [315, 316, 317]. С другой стороны, к неформальным институтам, исходя, к примеру, из определения Д. Норта относятся непрописные «привычки, обычаи, традиции и условности».

В. Тамбовцев [318, с. 45] определяет неформальными институты, включающие принуждение к исполнению правил, осуществляемое гарантами, которые не заняты постоянно в данном качестве, т. е., если гарант профессионален и постоянно реализовывает эту функцию — институт формальный, если функция реализуется нерегулярно — институт неформальный.

А. Аузан также классифицирует институты исходя из характеристик гаранта. Правила, гарантом которых может выступить любой член сообщества — неформальные институты. Формальные институты — это правила со специализированным гарантом. Обычно этот гарант — государство. Ранее [319] автором данного раздела был сформулирован «парадокс алгоритмического гаранта». Мы рассматривали неформальные институты, функционирующий на основе неизменного алгоритма (к примеру, совокупности смарт-контрактов). В этой ситуации, исполнение обязательств обеспечивается самим алгоритмом и не требует какого-либо дополнительного гаранта, ибо таковым является сам алгоритм. Фактически получается, что лишь взаимодействие, не отраженное в алгоритме является неформальным, взаимодействие в рамках алгоритма — формальное. А. Аузан отмечает, что поддержание формальных институтов требует затрат на содержание гаранта и настройку его стимулов. И именно исходя из этого критерия можно учесть фактор алгоритмизации. Если взаимодействие переводится на алгоритмическую основу волей государства, которое гарантирует достоверность данным алгоритма, — то это формальный алгоритмический институт. Если же происходит взаимодействие на основе алгоритма поддерживаемого

обществом самостоятельно – неформальный. Но все равно остается открытым вопрос: есть ли необходимость формализации в нормах права отношений гарантируемых государственным алгоритмом. По мнению авторов – нет.

С развитием новых технологий алгоритмы начинают играть все большую роль в функционировании институтов, определяя тем самым характер взаимодействия индивидов. С другой стороны, технологический прогресс дает возможность реализации человеконезависимых юридически значимых действий. Последние являются, с одной стороны, объектом (например, искусственный интеллект или децентрализованные автономные организации) правового регулирования, а с другой – методом (например, смарт-контракты) контроля и защиты выполнения обязательств. В свою очередь, это приводит к необходимости переосмысления, основополагающих правовых концепций (например, правосубъектности), и одновременно – к возможности внедрения новых технологий в процессы нормотворчества и правоприменения.

Фиксация отставания нормотворчества от отношений, возникающих в связи с технологическим прогрессом, обычно [320] интерпретируется как организационный барьер, в то время, как базисом данной проблемы является именно технологическая составляющая. В тоже время, с использованием подходов дизайна алгоритмических механизмов [321] возможно разработать новую технологию цифрового нормотворчества, использующую оппортунистическое поведение агентов для создания автоматически актуализируемого законодательства, обладающего свойством самоисполняемости. Что по сути является симбиозом формализации институтов через автоматизацию и стандартной правовой формализации

Научные исследования алгоритмизации и программирования правовых норм ведутся достаточно давно. В 1974 г. А. Шебанов, проводя разработки в области правовой информатики, определил правовую информацию как «совокупность сведений о праве и всех процессах и явлениях, с ним связанных» [322] Еще одной фундаментальной работой в данном направлении является исследование В. Жукова «Правовая информация: ее сущность и типология» 1985 г. [323]. Тема алгоритмизации права нашла свое отражения и во множестве диссертаций последующих лет. Примечательно, что возможности формализации раскрыты именно в области криминалистики и следствия. Вместе с тем большинство существующих работ в данной области

в основном затрагивают формально-логическое моделирование. Применение алгоритмов в качестве формализующей основы для работы с текстами нормативно-правовых документов было наиболее полно раскрыто в работах Г. Иванова. Так, в статье «Принципы построения формально-алгоритмических моделей нормативно-правовых документов» автор акцентирует внимание на возможности применения методик алгоритмизации (автоматных и блок-схемные) как в процессе разработки новых документов, так и для анализа корректности уже существующих нормативно-правовых актов [324]. В более поздних работах Г. Иванов уделяет внимание процедурам моделирования регулятивных функций, где предпочтение отдается построению автоматных моделей нормативно-правовой деятельности [325].

В контексте планирования и прогнозирования правотворчества А. Мазуренко в работе «К вопросу о методах правотворческой политики» рассматривает метод социально-правового моделирования, который призван обеспечить поддержку выбора наилучшего решения в какой-либо ситуации. В данном контексте основными задачами прогнозирования становится предвидение всех свойств законодательства как сложной социальной системы правовых норм и институтов. Само по себе формирование моделей возможно благодаря переносу норм права в знаковую систему [326]. О. Гаврилов в работе «Информатизация правовой системы России» предложил, используя логические операции, проектировать модели логической структуры правовой нормы в виде точно установленной последовательности: гипотеза нормы, диспозиция, санкция. Формализация права, по его мнению, позволит систематизировать право, выявить характер взаимосвязей между различными документами и разрешить существующие противоречия [327].

В настоящее время в России поддержка LegalTech и автоматизации права осуществляется инновационным центром «Сколково». В центре внимания находятся: автоматизация юридических процессов, использование роботов, ботов в контрактном праве и правовой экспертизе, коммерческом арбитраже, судебной деятельности, возможности искусственного интеллекта в цифровизации права, смарт-контракты и блокчейн.

Вопросам совершенствования правовой системы также уделяется большое внимание за рубежом. Одной из первых работ, в которой поднимается идея децентрализация права, стала работа Роберта Д. Кутера «Decentralized Law for a Complex Economy: The Structural

Approach to Adjudicating the New Law Merchant». В отличие от ранее рассмотренных подходов, автор говорит о том, что правовые нормы должны «просачивается снизу», т. е. децентрализованные законы должны начинаться с неформальных обычаев и контрактов. Для анализа обычаев Роберт Д. Кутер предлагает использовать теорию игр, с последующим формированием моделей поведения агентов. Примечательно, что данный подход является достаточно перспективным, однако подлежит реализации только при достижении определенного уровня развития технологий, позволяющих формировать правовые нормы непосредственно субъектами, что и с точки зрения права требует некоторого переосмысления, основополагающих концепций.

Вопросам анализа и формального представления юридического языка посвящены и работы F. Bodard, M. Hella, Y. Pouillet, P. Stenne. В частности в работе «A Prototype ADP System to Assist Judicial Decision Making» [328] дано описание прототипа для содействия принятию судебных решений, основные функции которого с одной стороны, заключаются в формальном документальном анализе элементов описания события, и с другой стороны, оценке согласованности принимаемых решений ранее существующей практике. Clemens Beckstein, Reinhard Stolle, Gerhard Tobermann в работе «Meta-Programming for Generalized Horn Clause Logic» [329] обосновывают, что для соблюдения парадигмы логического программирования управляющая информация также должна быть выражена декларативно. Программа должна включать логическую теорию, которая определяет проблему и контролирует информацию, представляющую собой стратегию построения решения. Claude Thomasset и Louis-Claude Paquin в работе «Expert Systems in Law and the Representation of Legal Knowledge: Can we isolate it from the Why and the Who?» [330] поднимаются проблемы, на которые стоит обратить особое внимание при алгоритмизации и переводе норм в машиночитаемый вид — это проблемы лексики и формулировок. Для каждой юридической конструкции должно быть найдено несколько эквивалентных выражений, подтвержденных как юристами, так и обычными гражданами.

Необходимость сближения права и программирования является предметом исследования не только правоведов, но и исследователи в области информатики уже давно работают в этом направлении. В частности в 1988 г. Бертран Мейер, французский исследователь, работающий много лет в России, опубликовал книгу «Object-Oriented

Глава 4

Software Construction» [331], в которой подробно описывались основы «Проектирования по контракту» – метода, проектирования программного обеспечения по модели построения договора. Фактически, Б. Мейер предложил механизм совмещения программного кода и договоров различных форм. Десять лет спустя 1996 Ник Сабо (Nick Szabo) в «Smart Contracts: Building Blocks for Digital Markets» [332], дал определение понятию «смарт-контракт», описав его как компьютерный протокол, который самостоятельно проводит сделки и контролирует их исполнение с помощью математических алгоритмов.

Так была заложена основа целого ряда последующих исследований, направленных на адаптацию сформулированных предложений к современной правовой системе.

Верификацию норм предлагается осуществлять посредством алгебраических вычислений и символического моделирования с использованием промежуточного языка [333].

Помимо научных, теоретических положений в последние годы стали появляться и некоторые практические решения. Например, ROSS – юридическая исследовательская платформа искусственного интеллекта, частично основанная на технологии IBM Watson, представила новый продукт EVA, который не только способен сопоставить тесты юридических документов, но провести проверку качества договора на основе исторических данных. Или Legal Zoom – проект, который предлагает программное обеспечение для автоматического создания юридических документов или бизнес-форм. Автоматизация типовых юридических услуг позволяет хозяйствующим субъектам уже сейчас самостоятельно решать в режиме реального времени ряд сложных задач без консультаций юристов.

Таким образом, вопросы автоматизации права, алгоритмизации и формализации норм с переводом их в машиночитаемый вид не раз становились темой научных исследований во многих странах мира. Однако, если отталкиваться от тезиса, что повышение степени автоматизации взаимодействия снижает необходимость его формализации в праве, то интеграция инновационных технологий в процесс нормотворчества скорее стоит рассматривать не как улучшение существующего процесса, а как постепенное движение в сторону изменения преобладающего типа формализации, а именно переход от правового типа к алгоритмическому.

Глава 5

ТЕХНОЛОГИИ ВИКИ И ИХ ПРИМЕНЕНИЕ

5.1. Вики-технологии

Вики — это веб-сайт (вики-ресурс), содержимое которого пользователи могут самостоятельно изменять с помощью инструментов, предоставляемых самим сайтом. Обычно вики-ресурс является информационным ресурсом, предназначенным для обмена информацией или создания базы знаний. Самым известным вики-ресурсом является Википедия. Формирование контента вики производится с использованием вики-разметки — специального упрощенного языка разметки, который предназначен для непосредственного использования конечными пользователями (авторами контента) для оформления контента вики-ресурса в виде страниц веб-сайта. Страницы, оформленные с применением вики-разметки, преобразуются в HTML для просмотра в веб-браузере с помощью специального программного обеспечения — вики-движков.

Таким образом, основными элементами технологии вики являются:

- вики-разметка, которая используется для создания страниц, статей и других элементов вики-ресурса;
- вики-движок, который позволяет создавать вики-ресурс, работая с ним через веб-браузер;
- методология создания вики-ресурса, заключающаяся в способах организации информации на вики-ресурсе (как на уровне взаимосвязей страниц, так и на уровне отдельных страниц) и в спо-

собах взаимодействия пользователей (в первую очередь – авторов) вики-ресурса.

Существует три основных типа использования технологии вики:

- публичные вики, создающиеся сообществом читателей и часто используемые в качестве энциклопедии;
- корпоративные вики, используемые для хранения документации и создания базы знаний;
- личные вики, используемые в качестве заметок или блога.

Что такое вики?

Что же такое вики? Вики, или вики-ресурс, – это веб-сайт с набором страниц определенной тематики. Пользователи вики имеют возможность сообща изменять структуру и содержимое этих страниц, создавать новые страницы с помощью инструментов, предоставляемых самим сайтом. Также вики может создаваться и использоваться рабочими группами в качестве места для совместной работы над проектом. Для форматирования текста страниц и вставки различных объектов в текст, например, изображений или таблиц, используется вики-разметка.

На базе таких принципов построены проекты Фонда Викимедиа [334]. Фонд Викимедиа – это некоммерческая благотворительная организация, которая поддерживает инфраструктуру для работы ряда мультязычных краудсорсинговых вики-проектов. Основным проектом фонда Викимедиа является Википедия, но в число проектов также входят: Викисловарь, Викицитатник, Викиучебник, Викиотека, Викигид, Викисклад, Викиданные, Викивиды, Викиновости, Викиверситет, Инкубатор Викимедиа и Мета-вики.

Обычно после того, как вики была создана, ей дается название, добавляются пользователи, описание и теги. На первой странице вики размещается основная информация о тематике данной вики и уведомляются добавленные пользователи. У каждого пользователя может быть свой уровень доступа, в зависимости от которого пользователь имеет право на изменение, создание и/или комментирование страницы сайта.

Например, рассмотрим процесс создания вики о машинах, целью которой является предоставление базы с описанием автомобилей всех стран и эпох. Создатель вики называет сайт «Машины

со всего света», на главной странице создает описание данной вики, приглашает пользователей и предоставляет им права на создание и/или изменение страниц. Главная страница будет содержать вводную информацию с объяснением темы и инструкцией по использованию (как создать, изменить страницу и т. д.). Такая страница будет основным помощником для единомышленников создателя вики.

Далее пользователь, который интересуется автомобилями Германии и владеет необходимой информацией, может создать раздел или страницу с заголовком «Автомобили Германии». А к данной странице создать уже дочерние страницы с описанием конкретных моделей автомобилей. Затем, другие пользователи по этому примеру могут создать страницу «Автомобили России», являющуюся по сути разделом ресурса, и наполнить эту страницу информацией, а сам раздел – ссылками на страницы с описаниями конкретных моделей автомобилей.

Помимо добавления новых страниц пользователи могут исправлять информацию на созданных ранее страницах. По дискуссионным вопросам пользователи могут организовать обсуждение, которое будет привязано к обсуждаемой странице, но не будет видно на ней непосредственно. По мере роста ценности и популярности ресурса количество пользователей может увеличиваться, приводя к росту активности по наполнению ресурса и повышению качества его контента. Как итог, вики может быстро разрастаться, наполняясь большим количеством сведений по своей тематике.

На разросшемся ресурсе поиск сведений о конкретной модели автомобиля может стать непростой задачей. Для ее решения вики-ресурс с помощью вики-движка может предоставлять много возможностей: пользователи могут выполнять поиск автомобилей по названию, искать названия в тегах или указателе страниц, искать среди нужных категорий или переходить с одной статьи на другую, используя ссылки между страницами вики (такие ссылки принято называть внутренними ссылками).

Сущность концепции вики заключается в том, что вики позволяет всем пользователям вики-ресурса создавать или редактировать страницу, используя обычный браузер. Вики поддерживает связи между страницами вики-ресурса за счет создания гипертекстовых ссылок на другие страницы вики-ресурса, т. е. внутренних ссылок. Ви-

ки-разметка позволяет создавать внутренние ссылки, основываясь на внутреннем имени страницы на вики-ресурсе, при этом допустимо создавать ссылки и на несуществующие страницы. При отображении ссылок на несуществующие страницы вики-движок визуально их выделяет, что указывает пользователям на целесообразность создания новой страницы с таким именем.

Важно понимать, что вики не является готовым сайтом, вики — это технология для организации непрерывного совместного самоорганизующегося процесса наполнения информационного ресурса, который реализуется за счет создания и редактирования страниц, категорий или других решений, который приводит к постоянному изменению (при правильной организации — улучшению) информационного ресурса.

Вики позволяет многократно редактировать текст страницы посредством только самой вики-среды, нет необходимости в использовании стороннего редактора. Для этого используется язык вики-разметки — это язык разметки, позволяющий легко и быстро размещать в тексте структурные элементы. Язык вики-разметки поддерживает механизм гиперссылок для создания связей между страницами вики-ресурса и является более наглядным и более безопасным, чем HTML. Большая безопасность вики-разметки обеспечивается тем, что использование JavaScript и каскадных таблиц стилей в ней ограничено.

В вики имеется возможность сравнения отредактированных страниц и восстановления их предыдущих версий, поскольку вики-движок ведет учет версий страниц. Изменения, внесенные на страницу, появляются на ней сразу после сохранения изменений. Вики поддерживает работу множества авторов, в некоторых вики изменения могут вносить все пользователи сайта.

Для того, чтобы вики-ресурс работал, необходима специальная информационная система, представляющая собой частный случай CMS (content management system — система управления контентом); такие системы называют вики-движками (wiki engine). Вики-движок — это программное обеспечение для организации вики, обычно является веб-приложением, выполняемым на одном или нескольких серверах. Обычно вики-движки относительно просто устроены по сравнению с другими CMS, так как в их устройстве и функциональности очень многие действия выполняются непосредственно пользователями.

Вики-разметка

Вики-разметка является облегченным языком разметки. Такие языки имеют минимальный набор функций, небольшой набор поддерживаемых тегов и простой синтаксис. Они разработаны для простого и быстрого создания текста с использованием любого универсального редактора, а также легко читаются пользователями в своей исходной форме. Их используют там, где может потребоваться прочитать необработанный документ и окончательный вывод. Например, пользователь, загружающий библиотеку программного обеспечения, может предпочитать читать документацию в текстовом редакторе, а не в веб-браузере. Другим видом использования таких языков является обеспечение ввода данных для веб-публикации, например, в блогах, на форумах, в социальных сетях и вики, где интерфейсом для ввода является простое текстовое поле. Облегченные языки разметки используются и в других веб-ресурсах, где человеку требуется подготовить текст в обычном редакторе или простом текстовом поле. Затем программное обеспечение сервера преобразует входную информацию в общий язык разметки документа, такой как HTML.

Рассматривая подходы к разметке текста, необходимо отметить, что различают два подхода к разметке текстов: логическую и визуальную разметки [335]. В первом случае речь идет только о том, какую роль играет данный участок документа в его общей структуре (напр., разметка указывает, что «данная строка является заголовком»). Во втором — определяется, как именно будет отображаться этот элемент (напр., разметка указывает, что «данную строку следует отображать жирным шрифтом»). Идея языков разметки состоит в том, что визуальное отображение документа должно автоматически получаться из логической разметки и не зависеть от его непосредственного содержания. Это упрощает автоматическую обработку документа и его отображение в различных условиях (напр., один и тот же файл может по-разному отображаться на экране компьютера, мобильного телефона и на печати, поскольку свойства этих устройств вывода существенно различаются). В этом смысле вики-разметка является логической разметкой. В редакторах, использующих принцип WYSIWYG (What You See Is What You Get — «что видишь, то и получишь») очень часто вместо логической разметки текста используется визуальная: например, создавая документ в редакторе MS Word, пользователь может выделять заголовки жирным шрифтом, но где не указывать, что эта строка является заголовком.

Вики-текст, также известный как вики-разметка, состоит из синтаксиса и ключевых слов, используемых программным обеспечением для форматирования страницы — вики-движком. С его помощью страницы, оформленные с применением вики-разметки, преобразуются в HTML для просмотра в веб-браузере. На данный момент не существует общепризнанного стандарта на вики-разметку, каждый вики-движок определяет собственный диалект вики-разметки, хотя обычно ключевые элементы вики-разметки схоже реализованы в разных вики-движках. Для определенности далее мы будем говорить о варианте вики-разметки вики-движка MediaWiki (Медиа-Вики) — инструмента, используемого Википедией и многими другими крупными публичными проектами.

Вики-разметка используется для создания и редактирования самых разнообразных элементов страницы вики. С ее помощью создают разные типы заголовков, определяют стиль и формат основного текста, создают списки. Также с помощью языка разметки можно создавать различные таблицы: синтаксис позволяет изменять их стиль, положение, выравнивать таблицу, выбирать ширину границы и работать с атрибутами конкретных ячеек. Более того, имеется возможность встраивать математические формулы и картинки в страницу вики. В частности, можно указывать их положение внутри текста, менять размер, добавлять или убирать рамку и так далее. Синтаксис вики-разметки позволяет работать с шаблонами — одним важнейших элементов вики, позволяющим поддерживать структуру как отдельных страниц, так и в каком-то смысле всего вики-ресурса. Шаблоны дают возможность использовать различные типовые элементы оформления вики страниц, в том числе, элементы, которые присущи только страницам, относящимся к каким-то специфическим группам (напр., к описанию организаций в Википедии). Пользователи могут создавать и редактировать необходимые им шаблоны аналогично тому, как они делают это со страницами.

Ключевым элементом страниц вики, определяющим структуру вики-ресурса, являются ссылки. Язык вики-разметки учитывает важность ссылок для вики-ресурсов и имеет большое количество инструментов, а именно специальных символов, для создания разного типа ссылок, используемых в вики [336].

В вики-разметке внутренние ссылки (т. е. ссылки на другие страницы того же вики-ресурса) создаются при помощи помещения двойных квадратных скобок вокруг текста, обозначающего название

страницы, на которую создается ссылка. Возьмем к примеру текст на странице Википедии: «В Лондоне хорошая [[погода]]». При преобразовании вики-движком текста внутри скобок в URL первая буква страницы назначения автоматически делается большой, внутренние пробелы автоматически представляются как подчеркивания (набор подчеркивания имеет тот же эффект, что и набор пробела, но не рекомендуется). Таким образом, приведенная ссылка ведет на <https://ru.wikipedia.org/wiki/Погода>, которая является статьей с названием «Погода».

Если падеж ссылающегося слова или словосочетания не совпадает с именительным, то следует в двойных квадратных скобках поставить вертикальную черту и слева от нее писать именительный падеж (форму, в которой определено имя страницы, на которую планируется сослаться), а справа — соответствующий грамматике предложения, в которое вставлена ссылка. Это свойство можно использовать, если вы хотите изменить название ссылки в тексте. Например: ссылка в виде «В Лондоне [[погода | солнечная погода]]» будет также вести на страницу <https://ru.wikipedia.org/wiki/Погода>, но в тексте будет отображаться так: «В Лондоне солнечная погода».

На страницах, сгенерированных Медиа-Вики, ссылки красного цвета говорят о том, что страница, на которую указывает ссылка, не существует. Эта страница может быть создана, если нажать на ссылку. Есть удобный синтаксис для создания ссылок, у которых написание в тексте отличается от целевого ресурса окончанием. Например, если в тексте вы пишете «гены», а необходимо сослаться на статью с названием «ген», то нужно указать в скобках только нужную часть: «[[ген]]ы». В таком случае выделится все слово как ссылка, но сослаться будет на статью «ген».

Вики-разметка позволяет сослаться не только на всю страницу, но и на определенные секции текущей страницы или секции другой страницы. Это очень удобно для структурирования и организации навигации по большим страницам. Для ссылки на секцию страницы используется знак «#» (например: [[Российская Федерация#История]]). Если секции, на которую идет ссылка, не существует, то ссылка приведет на начало страницы.

Вики-разметка позволяет легко создавать ссылки на страницы других вики-ресурсов, особенно если эти ресурсы используют тот же вики-движок. Например, в Википедии можно легко сослаться на статью на другом языке (технически проекты Википедии на раз-

ных языках — это отдельные вики-ресурсы), для этого в двойных квадратных скобках необходимо указать сначала код языка, например: «[[[:en:Lightweight markup language | Облегченные языки разметки]]». Такая ссылка указывает на страницу «Lightweight markup language» в англоязычной Википедии.

Существуют ссылки, которые называются интервики. Они ведут вас на другие сайты вики, для этого необходимо указать название такого сайта в начале квадратных скобок, например: [[Wiktionary:привет]]. Такая ссылка будет выглядеть в тексте как «Wiktionary:привет», но можно сократить ее до просто «привет», если написать так: «[[Wiktionary:привет]]». Такая ссылка ведет на страницу <https://ru.wiktionary.org/wiki/привет>. Если вам нужно указать ссылку на внешний ресурс, то для этого необходимо использовать одиночные квадратные скобки и начинаться она должна с «http://» или «https://». Например, ссылка «[http://www.wikipedia.org Wikipedia]» в тексте отобразится как «Wikipedia». В одиночных скобках указываются и ссылки на электронную почту, например: «[mailto: name@example.com name@example.com]».

Адрес, содержащий определенные символы, будет отображаться и ссылаться неправильно, если эти символы не закодированы в специальном формате [337]. Например, пробел должен быть заменен на %20 (примеры кодов для других распространенных символов можно увидеть в таблице 10).

Таблица 10

Кодировки некоторых спецсимволов для оформления ссылок в Медиа-Вики

<i>space</i>	"	'	,	;	<	>	?	[]
%20	%22	%27	%2c	%3b	%3c	%3e	%3f	%5b	%5d

Еще один тип ссылок — это ссылки на категорию. Категории — один из ключевых способов организации информации в Википедии, они предназначены для группировки страниц по подобным темам и помогают читателям находить предметную область и перемещаться по страницам, относящимся к ней. Чтобы добавить ссылку на категорию, нужно описать ссылку в виде: «[[Категория: ИмяКатегории]]»

на странице вики. Наличие такой ссылки приведет к тому, что страница добавится в категорию «ИмяКатегории», если такая есть. Если такой категории нет, то она будет автоматически создана. Такие ссылки, как и ссылки на статьи на другом языке, в Википедии рекомендуется указывать в конце статьи. Есть возможность создавать ссылки на страницу какой-либо категории, не добавляя свою статью в эту категорию, при помощи следующего синтаксиса: «[[Категория:ИмяКатегории]]». Такая ссылка перенаправит вас на страницу категории «ИмяКатегории», но сама статья к данной категории отнесена не будет.

Правила формирования вики-статей

Так как страницы в вики могут создаваться и редактироваться пользователями, то должны быть правила или указания, которые подскажут, как писать статью. У каждого вики-ресурса свои цели, своя организация и свои способы взаимодействия, и соответственно, каждый такой сайт имеет свои правила или указания по работе в нем. Соответственно, перед созданием страницы в вики-ресурсе необходимо ознакомиться с этими правилами. Эти правила определяются для того, чтобы поддерживать структуру вики в соответствии с целями и способами их достижения, определенными для этого ресурса. Такие правила являются неотъемлемой и такой же важной частью технологии создания вики-ресурсов, как вики-разметка и вики-движки.

Несмотря на индивидуальность конкретных правил каждого вики-ресурса, существуют общие правила и рекомендации, с большой вероятностью применимые ко всем типам вики-ресурсов. Их можно рассмотреть на примере правил самой известной вики — Википедии [338]:

- публикуемая информация должна соответствовать формату вики-ресурса;
- статья должна быть написана грамотным языком;
- перед созданием новой статьи нужно проверить, что подобную статью не написали раньше;
- статью нужно писать беспристрастно и объективно, придерживаясь нейтральной точки зрения;
- статья должна охватывать одну определенную тему и раскрывать ее максимально удобно для читателя;

- статья не должна состоять из двух слов или же, наоборот, быть сразу обо всем: необходимо найти золотую середину, в том числе учитывая, что при необходимости статью можно разбивать на несколько статей или новую статью можно не создавать, поместив информацию в раздел одной из существующих статей;
- хорошо, когда статья содержит иллюстрации или таблицы, а также использует готовые шаблоны, которые помогают структурировать информацию;
- необходимо поддерживать структуру вики-ресурса.

Последняя рекомендация требует подробного разъяснения. Вики-ресурсы хоть и не имеют четкой структуры, но все равно гораздо более структурированы, нежели Всемирная паутина. Структурированность вики-ресурсов достигается в основном за счет ссылок, категорий и использования шаблонов, применяемых для однотипного оформления аналогичных статей. Поэтому нужно ссылаться на свою статью со связанных по теме статей, а в самой статье давать ссылки на другие страницы, даже если таких страниц еще нет. Ссылаться можно на людей, места, организации, даты и события, на термины или другие ключевые слова. Также каждую статью нужно добавить в одну или несколько категорий. Категории позволяют логически сортировать статьи по схожести в содержании, что значительно упрощает поиск нужной информации. Желательно, чтобы у статьи не было категорий разного уровня, при этом статью можно добавить в разные ветки дерева категорий.

Данный свод рекомендаций является базовым, но нужно учитывать особенности каждого конкретного вики-ресурса. Каждый сайт может иметь свои требования, например по минимальному количеству знаков в статье, по стилю написания текста, по наименованию статей; некоторые ресурсы имеют специфические стилистические требования к оформлению текста. Но самое главное — нельзя оставлять статью одну не связанной ссылками и категориями со всем массивом статей вики-ресурса. Выполнение этого правила позволяет формировать и поддерживать структуру всей вики.

Вики-движки

Вики-движок — это программное обеспечение для организации вики, которое позволяет пользователям создавать и совместно редактировать страницы через веб-браузер. Вики-движок представляет собой

обычное веб-приложение, которое выполняется на одном или нескольких веб-серверах. Контент, включая все текущие и предыдущие версии страниц, обычно хранится в файловой системе или в базе данных. В настоящее время существуют десятки активно поддерживаемых вики-движков, реализованных на различных языках программирования, включая как приложения с открытым кодом, так и частные приложения. Они сильно различаются по поддержке различных платформ, поддержке языков, поддержке условий на естественном языке, а также по предположениям о техническом контроле за редактированием.

Так как существует три основных типа использования вики: публичные вики, корпоративные и личные, то и под каждый тип существуют свои вики-движки. Некоторые вики-движки предназначены для одного конкретного типа использования, в то время как другие могут использоваться для всех трех. Для поддержания разнообразных функциональных возможностей в вики-движках зачастую применяется механизм плагинов, позволяющий легко адаптировать вики-движок для конкретных задач вики-ресурса.

Публичные вики — это вики, которые могут быть прочитаны кем угодно, и, обычно, отредактированы кем угодно. Среди публичных вики доминирующим вики-движком является Медиа-Вики. Его популярность поддерживается тем, что он используется в самой популярной в мире вики — Википедии. Среди других вики-движков, активно используемых для публичных вики, стоит выделить: MoinMoin, PmWiki.

За последние 20 лет создания совместно пополняемых и управляемых баз знаний и энциклопедических сайтов, таких как Википедия, вики эволюционировали в социальный опыт обмена знаниями. Сейчас уже нет необходимости лично разворачивать вики, есть возможность создать свое вики-пространство в облачном сервисе. Например, Confluence wikis — коммерческое облачное решение или загружаемое программное обеспечение для установки на сервер заказчика. Рабочая среда Confluence позволяет создавать собственное пространство, использовать инструменты рабочего процесса, календари, «drag and drop» обмен файлами и социальные функции, такие как общий доступ, упоминания и потоки действий. Также можно создать библиотеки страниц вики в Microsoft Office 365. Microsoft Office 365 — это облачный продукт коммерческого класса для создания онлайн-библиотек, организации совместной работы коллег и под-

ключения корпоративных сообществ, общедоступных или частных. Библиотека вики легко настраивается в SharePoint Online в составе Office 365 и позволяет создавать вики-страницы для интрасети или внешней системы совместной работы и управления контентом.

Корпоративные вики используются для хранения документации и обмена знаниями между сотрудниками одной компании или организации (в том числе и неформальных организаций). В данном виде программного обеспечения, как правило, уделяется больше внимания таким функциям, как управление доступом, интеграция с другим программным обеспечением и управление документами. К корпоративным вики-движкам относят: XWiki, TWiki, Tiki Wiki CMS Groupware и другие. Они являются программным обеспечением с открытым исходным кодом и предназначены для корпоративного сегмента. Также для этих целей может использоваться и движок MediaWiki. Одним из самых популярных частных вики-движков корпоративного сегмента является Confluence. Многие компании и правительственные организации используют такие вики для организации своей работы, например, среди них такие известные компании, как Adobe Systems, Amazon, Intel, Microsoft, Kaspersky Lab и другие.

Программное обеспечение, специально разработанное для ведения личных вики-ресурсов, включает в себя NotePub, Pimki, WikidPad, TiddlyWiki и ConnectedText. Такие ресурсы часто используются для ведения блогов, для обмена информацией в кругу очень маленьких групп или, если для проведения исследования необходим удобный инструмент, для организации данных для исследования с возможностями вики. Движок MoinMoin является более универсальным, но также может использоваться для создания личных вики. Так как формированием контента личных вики-ресурсов пользуется только один пользователь, то движки, ориентированные только на это использование, пользуются меньшей популярностью в сравнении с остальными.

Рассмотрим наиболее популярные вики-движки на сегодняшний день и сравним их [339, 340, 341]. Для нашего анализа мы выбрали: MediaWiki, MoinMoin, PMWiki, XWiki, TWiki, Tiki Wiki CMS Groupware, Confluence, SharePoint Wiki Plus, DokuWiki, ConnectedText, TiddlyWiki.

В таблице 11 сравним вики-движки по общей информации. Сравнимые критерии:

Таблица 11

Сравнение вики-движков по общей информации

Вики-движок	(Бес)платный, цена лицензии	Язык программирования	Хранимые данные	Статус разработки	Целевая аудитория
MediaWiki	бесплатный, 0	PHP	база данных	готовый	конечные пользователи, публичные вики, образование
MoinMoin	бесплатный, 0	Python	файловая система	готовый	все, у кого меньше 100000 страниц
PMWiki	бесплатный, 0	PHP	файловая система	готовый	все
XWiki	бесплатный, 0	Java	база данных	готовый	предприятия, рабочие группы, разработчики
TWiki	бесплатный, 0	Perl, JS	файловая система	готовый	большие корпорации, маленький и средний бизнес
Tiki Wiki CMS Groupware	бесплатный, 0	PHP	база данных	готовый	для тех, кому нужна полностью функциональная CMS с баг-трекером, форумом и блогом
Confluence	платный, 10\$ за установку	Java	база данных	готовый	предприятия и технические команды
SharePoint Wiki Plus	платный, 3200\$ в месяц	.Net—C#	база данных	готовый	пользователи SharePoint, желающие использовать возможности корпоративной вики
DokuWiki	бесплатный, 0	PHP	файловая система	готовый	личные блоги, маленькие и средние компании
Connected-Text	платный, 40\$ в месяц	язык программирования не имеет значения (полностью автономный)	база данных	готовый	любой, кто проводит исследования, нуждается в инструменте с возможностями вики для организации и поиска данных
TiddlyWiki	бесплатный, 0	HTML, JS	файловая система	готовый	личные блоги, для маленьких групп

Глава 5

- бесплатный ли движок, если он платный, то какова цена лицензии;
- язык программирования, на котором разработан вики-движок;
- хранение данных – этот критерий определяет, какой метод используется для хранения страниц вики;
- статус разработки: разрабатывается ли движок до сих пор, насколько готовым является продукт;
- основная целевая аудитория.

В таблице 12 сравним движки по следующим критериям:

- поддерживаемые операционные системы;
- в каких базах данных может храниться информация;
- есть ли защита от спама;
- можно ли загружать файлы;
- можно ли редактировать изображения через интерфейс вики;
- являются ли форумы частью движка.

Если сравнить эти вики-движки по общим функциям, таким как:

- есть ли предпросмотр;
- помечаются ли незначительные изменения;
- можно ли добавить короткий комментарий к изменению страницы;
- можно ли просмотреть историю страницы;
- можно ли расширить функционал, используя плагины;
- индексированы ли страницы, –

то будет видно, что все отобранные движки выполняют эти функции, так как это базовый функционал.

В таблице 13 рассмотрим специальные возможности:

- поддержка кодировки Unicode;
- работает ли для RTL-языков;
- сколько языков поддерживает интерфейс;
- могут ли авторы или пользователи получать электронные письма об изменениях;
- можно ли добавить страницы в категории;
- можно ли создать автоматическое перенаправление с одной страницы на другую.

Если сравнить по поддерживаемым ссылкам, например, таким как:

Таблица 12
Сравнение вики-движков по ОС, БД и другим критериям

Вики-движок	Операционные системы	Базы данных	Защита от спама	Загрузка файлов	Редактирование изображений	Форумы
MediaWiki	*nix, Windows, Mac OS X	MySQL, PostgreSQL, Oracle, SQLite	да, капча, черный список ссылок, черновый список слов, блокирование IP-адреса	да	нет	плагин
MoinMoin	UNIX, Windows, MacOS X	текстовые файлы	да, фильтр плохого контента через регулярные выражения	да	да	нет
PMWiki	UNIX, Windows, MacOS X	текстовые файлы, MySQL (плагин), SQLite (плагин), RCS (плагин)	да	да	плагин	плагин
XWiki	любая платформа, поддерживающая JDK 1.8 и выше	MySQL, PostgreSQL, Oracle	да, капча	да	плагин	плагин
TWiki	Linux, Windows, OS-X	текстовые файлы, RCS	да	да	плагин	плагин
Tiki Wiki CMS Groupware	Linux, FreeBSD, Windows, MacOS X	MySQL	да, капча	да	да	да
Confluence	Windows, Linux, Unix	MySQL, PostgreSQL, Oracle	да, капча	да	плагин	плагин
SharePoint Wiki Plus	Windows Server 2003/2008	MS SQL Server	да	да	да	да
DokuWiki	Linux, UNIX, Windows, MacOS X	текстовые файлы	да, черный список	да	нет	нет
ConnectedText	Windows	текстовые файлы, Firebird	нет	да	нет	нет
TiddlyWiki	любая	текстовые файлы, самостоятельно содержится в html-файлах	нет	плагин	да	плагин

Таблица 13

Сравнение вики-движков
по специальным возможностям

Вики-движок	Поддержка Unicode	RTL-языки	Количество языков	Извещение об изменениях	Категории страниц	Автоматическое перенаправление
MediaWiki	да	да	140	да	да	да
MoinMoin	да	да	—	да	да	да
PMWiki	да	да	38	да	да	да
XWiki	да	нет	38	да	да	да
TWiki	да	нет	19	да	да	да
Tiki Wiki CMS Groupware	да	да	34	да	да	да
Confluence	да	нет	15	да	да	плагин
SharePoint Wiki Plus	да	да	—	да	да	да
DokuWiki	да	да	55	да	плагин	плагин
ConnectedText	да	нет	5	нет	да	да
TiddlyWiki	да	да	4	—	да	плагин

- можно ли создать ссылки, которые содержат любые символы;
- можно ли отобразить все страницы, которые ссылаются на данную интервики;
- можно ли создать ссылки между несколькими вики сайтами;
- можно ли использовать картинки в качестве ссылок, –

то будет ясно, что все отобранные движки их поддерживают, т. е. этот функционал можно назвать базовым для развитых вики-движков.

Таким образом, мы сравнили самые популярные движки, используемые для разных задач по основным функциям. Конечно, у каждого движка существует еще десятки разных дополнительных функций, которые могут быть присуще только ему, и нужно выбирать движок, который будет удобен для конкретной задачи. Самая популярная в мире вики, Википедия, остановила свой выбор на вики-движке Медиа-Вики, который будет рассмотрен более подробно ниже.

Вики-технологии для создания корпоративных баз знаний

Корпоративные вики используются для хранения документации или создания баз знаний. База знаний — это особого рода база данных, разработанная для управления знаниями (метаданными), т. е. сбором, хранением, поиском и выдачей знаний [342].

В зависимости от уровня сложности систем, в которых применяются базы знаний, различают:

- БЗ всемирного масштаба — например, Интернет или Википедия (все языковые разделы);
- БЗ национальные — например, Википедия (один языковой раздел);
- БЗ отраслевые — например, Автомобильная энциклопедия;
- БЗ организаций;
- БЗ экспертных систем;
- БЗ специалистов.

Простые базы знаний могут использоваться для создания экспертных систем и хранения данных об организации: документации, руководств, статей технического обеспечения. Главная цель создания таких баз — помочь менее опытным людям найти существующее описание способа решения какой-либо проблемы из определенной предметной области.

База знаний является важным компонентом интеллектуальной системы. Наиболее известный класс таких программ — это экспертные системы, предназначенные для построения способа решения специализированных проблем, основываясь на записях базы знаний и на пользовательском описании ситуации.

Для организации, в частности банка, корпоративная база знаний — это некая среда, область, в которой хранится вся информация о работе компании или определенном аспекте работы компании. В эту базу можно записать накопленные знания и опыт, результаты встреч и конференций, основные тезисы и задачи, выявленные там, полезные статьи, архив реализованных проектов и т. д. Главное при создании и ведении такой системы — это вести ее с самого начала аккуратно, чтобы любой пользователь мог отыскать нужную информацию, а также грамотно, чтобы, прочитав, каждый мог ее понять и применить. Без выполнения этих требования вики, задуманная как база знаний, превратится в огромный поток неструктурированной информации.

К особенностям вики-систем, полезных для корпоративного управления, можно отнести:

- Возможность вводить информацию с помощью быстрых и простых алгоритмов создания страниц, содержащих ссылки на другие корпоративные информационные ресурсы, ускоряя тем самым формирование базы знаний.
- Возможность избежать перегрузки электронной почты. Вики позволяют распределить всю необходимую информацию среди людей, занятых в данном проекте. В отличие от списков рассылки, которые могут содержать большое количество информации, избыточной для большинства подписчиков (т. е. по сути порождавая внутрикорпоративный спам), при использовании корпоративной вики только люди, заинтересованные конкретным явлением (например, проектом), будут просматривать связанные с ним страницы корпоративной вики.
- Структурирование информации. Вики позволяют пользователям структурировать как новую, так и имеющуюся информацию.
- Вики позволяет структурировать выражение мнений разных сотрудников по конкретному вопросу на одной странице. Эта функция очень полезна при написании документов, подготовке презентаций и других информационных объектов, когда мнения участников работы расходятся.
- Разграничение прав и ролей. Пользователям может быть отказано в доступе для просмотра и/или редактирования данной страницы в зависимости от их роли в организации и данном проекте.
- Возможность всеобъемлющего поиска – вики-движки позволяют эффективно искать информацию в рамках вики-ресурса, при этом корпоративные вики-движки зачастую поддерживают дополнительные опции поиска, особенно полезные для поиска по корпоративным базам знаний, сформированным в форме вики.

Часто корпоративная база знаний создается практически всеми подразделениями компании: отделом продаж, разработчиками ИТ-систем, генеральным директором. Каждый отдел компании имеет свою специфическую и важную информацию, которую необходимо зафиксировать для более удобного обращения к ней, а также чтобы не потерять накопленные знания и опыт в случае ухода сотрудника. Это и является ценностью. Знания не завязаны на сотрудниках, а принадлежат компании, накапливаются и передаются.

При разработке базы знаний надо учитывать ее размер, степень сложности, стабильность и форму знаний. Эти вопросы определяют выбор методологии, а также программного и аппаратного обеспечения, которое планируется использовать при создании базы знаний.

Как пример хотелось бы привести систему Atlassian Confluence [343]. Это одна из самых известных в мире вики-систем для создания баз знаний компаний и рабочих команд. Она дает возможность собрать всю необходимую информацию в одном месте, создавать документы, структурировать их и публиковать с различными правами доступа. Она имеет огромное количество известных клиентов, среди которых Лаборатория Касперского, которая использует ее для создания баз знаний в рабочих командах. Помимо Atlassian Confluence лидерами среди сервисов для ведения корпоративной вики являются: MediaWiki, Redmine, SharePoint Server.

Корпоративные вики полезны и используются компании любой сферы от ИТ-гигантов до финансовых компаний, консалтинга, банков. К услугам корпоративных вики прибегают компании из всех сфер бизнеса и самого разного размера в случае, если им нужно хранить накопленный в компании опыт. Можно найти множество примеров внедрения и использования баз знаний и в российских компаниях и банках. В приведенных ниже примерах мы рассмотрим опыт некоторых российских банков по использованию корпоративных вики-ресурсов.

В Почта Банке внутренняя информация о тарифах, регламентах, формах заявлений и многом другом, т. е. почти вся ключевая информация, необходимая для сотрудников фронт-линии, клиентской службы и головного офиса, доступна в едином универсальном хранилище, которое получило название «База знаний». Благодаря этой базе знаний все основные изменения, которые сотрудникам «Почта Банка» необходимо учитывать в процессе обслуживания клиентов, собраны на одном ресурсе. Для дополнительного удобства пользователей «Базы знаний» им на электронную почту могут приходить оповещения о ее обновлениях. База знаний интегрирована во внутренний корпоративный портал Почта Банка и доступна для сотрудников по прямой ссылке как внутри сети Почта Банка, так и за ее пределами. Такой инструмент очень удобен для сотрудников, так как обладает современным интуитивным интерфейсом и удобным поиском по разделам и продуктовым категориям [344].

Другим примером является Альфа-Банк. До появления в банке структурированной базы знаний во внутренних информационных документах сотрудников розничного блока Альфа-Банка существовала серьезная проблема: для обслуживания клиентов сотрудникам приходилось искать инструкции, регламенты, информационные письма в огромном количестве разрозненных источников. Опытный сотрудник наизусть знал «секретные» папки, где можно найти нужную информацию, но и при наличии большого опыта все запомнить было невозможно. А вот новому сотруднику было очень сложно приобрести ценный опыт по поиску информации и фактически воссоздать фрагмент неявной корпоративной базы знаний в своей голове. Более того, опыт сотрудников быстро устаревает, так как бизнес-процессы и регуляторная среда в банковском секторе очень быстро меняются, внутренняя документация и ее структура постоянно обновляются, и об этом необходимо практически постоянно оповещать огромное количество сотрудников. При этом даже единичная ошибка сотрудника может стоить банку серьезных убытков. Чтобы решить проблему, сразу два бизнес-направления – розничный бизнес и операционный блок – параллельно запустили аналогичные проекты по созданию корпоративных баз знаний.

В рамках проекта, выполняемого компанией Microsoft, было предложено экономичное и гибкое решение, которое позволило объединить проекты розничного и операционного блоков и построить единую базу знаний для всего Альфа-банка. Для решения была выбрана платформа управления знаниями на основе SharePoint, которая внедрялась в два этапа. На первом этапе все существовавшие на тот момент информационные ресурсы банка были проиндексированы корпоративным поиском, а на втором была создана специализированная база знаний и доработаны шаблоны страниц и компоненты системы. После выполнения этих работ документы были представлены в виде вики-страниц. Для отдельных направлений в базе знаний были разработаны специальные шаблоны, которые помимо описания самого продукта содержат информацию о часто задаваемых вопросах, ошибках, сопутствующих продуктах. Это помогает сотрудникам оперативно реагировать на сложные вопросы клиентов, что способствует повышению лояльности клиентов и повышению уровня их удовлетворенности качеством услуг банка.

Более того, внедренное решение является гибким и универсальным: база знаний может бесконечно трансформироваться под новые

бизнес-задачи. Гибкость решения проявила себя в том, что сотрудникам операционного блока удалось без привлечения службы ИТ разработать новый шаблон формы для открытия счета корпоративным клиентам. Подобная автоматизация процесса позволит сэкономить банку до 5,5 млн руб. только в 2016 г. [345].

5.2. Википедия

Цели Википедии

Википедия — это бесплатная свободная энциклопедия, поддерживаемая фондом Викимедиа и основанная на модели открытого для редактирования контента. На сегодняшний день Википедия имеет около 5,5 млн статей в англоязычной Википедии и около 46,8 млн статей во всех языковых разделах в сумме. В англоязычной Википедии за всю ее историю было произведено более 925 млн изменений и зарегистрировано свыше 32 млн пользователей, включая 1241 администраторов и свыше 130000 активных пользователей. Русскоязычный раздел Википедии содержит около 1,4 млн статей и 5,5 млн страниц. В нем было произведено свыше 100 млн правок, зарегистрировано свыше 2,2 млн пользователей, включая 83 администратора [346].

Википедия совместно написана добровольцами, которые, в основном, занимаются данной работой без оплаты. Любой пользователь, имеющий доступ к Интернету, может вносить изменения в статьи Википедии. Основной целью Википедии является передача накопленных знаний и постоянное увеличение объема и качества этих знаний, создание полноценной, точной, свободной энциклопедии.

Основополагающими принципами Википедии являются следующие [347]:

- Википедия — это энциклопедия. Она сочетает в себе многие особенности общих и специализированных энциклопедий, альманахов и справочников.
- Википедия пишется с нейтральной точки зрения. Википедия стремится к статьям, которые документируют и объясняют основные точки зрения беспристрастным тоном, избегает пропаганды. Все статьи должны стремиться к достоверности, ссылаясь на надежные, авторитетные источники.
- Википедия — это свободный контент, который каждый может использовать, редактировать и распространять. Поскольку все ре-

дакторы свободно предоставляют свою работу публике, ни один редактор не владеет статьей, и любые вклады могут и будут беспощадно редактироваться и перераспределяться. Заимствование несвободных носителей иногда допускается как справедливое использование, но сначала стараются найти бесплатные альтернативы.

- Редакторы Википедии должны относиться друг к другу с уважением и вежливостью. Должны действовать добросовестно и быть открытыми и гостеприимными с новичками, возникшие конфликты должны спокойно обсуждаться на соответствующих страницах обсуждения.
- У Википедии нет твердых правил. В Википедии есть политики, рекомендации, их содержание и интерпретация могут развиваться со временем. Каждая предыдущая версия страницы сохраняется, поэтому ошибки можно легко исправить.

Википедия — это живое сотрудничество, отличное от бумажных справочных источников. В отличие от печатных энциклопедий, Википедия постоянно создается и обновляется, причем статьи об исторических событиях появляются в ней в течение нескольких минут, а не месяцев или лет. Поскольку каждый может помочь улучшить контент, Википедия стала более всеобъемлющей, чем любая другая энциклопедия.

Помимо количества контента авторы Википедии также работают над улучшением его качества. Википедия — это незавершенное производство, со статьями на разных этапах реализации. По мере развития статей они, как правило, становятся более полными и сбалансированными. Качество также улучшается с течением времени, поскольку дезинформация и другие ошибки удаляются или исправляются. Однако поскольку каждый может щелкнуть «редактировать» в любое время и добавить материал, любая статья может содержать необнаруженные факты дезинформации, ошибки или вандализм. Осознание этого помогает читателю получать достоверную информацию.

Ежедневно на всех языках просматривается около 1 млрд страниц Википедии, сайт Википедии во всех странах посещают около 320 млн пользователей в день [348, 349]. Наибольшее количество пользователей приходится на английскую Википедию, около 180 млн пользователей в день, в то время как среднее количество посещений русской Википедии, около 25 млн пользователей в день. Русскоязыч-

ная Википедия почти ежегодно получает премию Рунета в номинации «Наука и образование», а также множество других наград [350].

Сообщество Википедии в значительной степени самоорганизуется, так что любой пользователь может создать репутацию компетентного редактора и участвовать в любой роли, которую он может выбрать при условии одобрения со стороны коллег. Частные лица часто принимают участие в специализированных задачах, таких как просмотр статей по просьбе других, просмотр текущих изменений для целей идентификации вандализма, просмотр вновь созданных статей для контроля качества.

Редакторы, которые считают, что они могут лучше обслуживать сообщество, взяв на себя дополнительную административную ответственность, просят своих коллег о получении дополнительных прав. В настоящее время для того, чтобы получить такие полномочия, требуется как минимум высокий уровень поддержки других членов сообщества (кандидатура должна получить одобрение от 75–80% участников процедуры). Администраторы, одобренные сообществом, могут удалять статьи, блокировать учетные записи или IP-адреса и редактировать полностью защищенные статьи. Бюрократы могут переименовывать учетные записи участников, присваивать и снимать статус администратора.

Существует Арбитражный комитет Википедии. Они рассматривают споры, которые остаются нерешенными после того, как другие попытки разрешения споров не увенчались успехом. Члены комитета избираются сообществом, и, как правило, выбираются из числа опытных администраторов. Стюарды обладают высшим уровнем разрешений сообщества. Стюарды имеют право выполнять некоторые технические операции, и никто почти никогда не слышит об их действиях, так как они обычно действуют, только когда местный администратор или бюрократ не доступен, а значит – почти никогда. Стюардов очень мало.

Для участников сообщества существуют и реальные встречи, например, вики-встречи – локальные встречи участников Википедии одного или нескольких городов. Обычно они проходят на территории организаций, музеев и других достопримечательных мест с целью популяризации труднодоступных энциклопедических знаний. Наряду с ними существуют вики-конференции – более глобальные встречи на уровне страны. В России такие конференции проходят ежегодно, начиная с 2008 г. Самым главным событием является Ви-

кимания [351] – ежегодная международная конференция Фонда Викимедиа. Являясь одновременно научной конференцией и встречей единомышленников, Викимания собирает вместе участников различных проектов Фонда Викимедиа для заслушивания докладов и презентаций о текущих и завершенных проектах, обмена идеями и поиска новых контактов. Викимания проводится с 2005 г.

Еще одним способом координации и связи участников является мета-вики (Meta-Wiki) [352], сайт глобального сообщества для разных проектов Фонда Викимедиа и движения Викимедиа в целом. Обсуждение мета-вики варьируется от координации и документации до планирования и анализа будущих мероприятий Викимедиа.

Описание вики-движка Медиа-Вики

Медиа-Вики (MediaWiki) [353] – это бесплатное серверное программное обеспечение, предназначенное для работы на большой серверной ферме для веб-сайта, который получает миллионы обращений в день. Этот движок использует Википедия и другие проекты Фонда Викимедиа. Медиа-Вики чрезвычайно мощный, масштабируемый и многофункциональный вики-движок, который использует PHP для обработки и отображения данных, хранящихся в реляционных базах данных, таких как MySQL, PostgreSQL, Oracle, SQLite. Медиа-Вики работает на операционных системах семейства Unix и Linux, Windows, Mac OS X и на любых веб-серверах с поддержкой PHP. Никаких других системных требований нет.

Страницы в Медиа-Вики используют вики-разметку, так что пользователи без знания HTML или CSS могут легко их редактировать. Когда пользователь редактирует страницу, Медиа-Вики записывает ее в базу данных, но не удаляет предыдущие версии страницы, что позволяет легко возвращаться к предыдущим версиям в случае вандализма или спама. Медиа-Вики позволяет редактировать конкретную секцию страницы вместо всей страницы, что удобно для длинных текстов, а также позволяет создавать шаблоны страниц для упрощения создания новых страниц. Синтаксис вики-разметки Медиа-Вики поддерживает математические формулы, пользовательские стили, выделение синтаксических ошибок, сноски, комментарии, добавленные автором и видимые только на исходной странице, включение других страниц в новую. Медиа-Вики поддерживает свободные ссылки, содержащие любые знаки, обратные ссылки, ко-

торые ссылаются на данную страницу, интервики-ссылки (ссылки между разными вики-ресурсами), использование картинок в качестве ссылок.

В Медиа-Вики для комментирования страниц можно создавать специальные страницы обсуждений; кроме того, все страницы можно категорировать, движок поддерживает кодировку Unicode и RTL-язык, поддерживает множество языков интерфейса и может сообщать пользователю об изменении страницы по электронной почте. Медиа-Вики формирует список недавних изменений, список несуществующих страниц, на которые есть ссылки, список страниц, на которые никто не ссылается, а также списки наиболее и наименее популярных страниц. Медиа-Вики поддерживает HTML-, XML-экспорт страниц или же экспорт в исходном синтаксисе вики-разметки.

Медиа-Вики также может управлять изображениями и мультимедиа, которые хранятся в файловой системе. С помощью этого движка могут создаваться галереи изображений, календари и форумы, для двух последних необходим плагин. Flash-файлы и видео могут быть напрямую встроены в страницу. Для больших вики с большим количеством пользователей Медиа-Вики поддерживает кеширование и может быть легко сопряжен с программным обеспечением прокси-сервера Squid.

Но есть задачи, с которыми движок Медиа-Вики справляется слабо. Поскольку MediaWiki был разработан для открытого контента, он часто не подходит для ситуаций, когда нужно ограничить доступ к части вики. Этот движок был разработан для обслуживания сайтов с высоким трафиком, таких как Википедия. Он был оптимизирован для этого использования и может не подходить для небольших сайтов, где дисковое пространство или память являются большими ограничениями, чем пропускная способность. Он обычно не является хорошей заменой для специального программного обеспечения под форумы и блоги. Для всех этих случаев существуют более подходящие вики-движки.

Организация и структура Википедии

В данном разделе мы рассмотрим, как формируется структура Википедии. Структура любого вики-ресурса создается за счет использования различных технологий и правил формирования связей страниц друг с другом. В частности, в Википедии для этого используются

правила и указания, по которым нужно писать статьи, а также такие вики-технологии, как пространства имен статей, категории, списки, а также различные виды шаблонов, включая навигационные. Все это позволяет организовывать информацию в Википедии в гораздо более удобном и структурированном виде, чем во Всемирной паутине.

Пространства имен

Пространства имен Википедии оформляются с помощью использования двоеточия для разделения пространства имен и имени статьи, например, в имени страницы «Википедия:Помощь_начинающим» слово «Википедия» — это пространство имен (оно содержит различные статьи, связанные собственно с функционированием Википедии, например, страницы помощи, правила и руководства, дискуссии), а «Помощь_начинающим» — имя статьи. Пространства имен позволяют организовать и отличать страницы контента от служебных страниц.

Пространства имен разделяют страницы на несколько больших групп, страницы в одних группах предназначены для публичного просмотра, а в других предназначены для людей редактирующих Википедию. Служебные страницы, как и обычные пользовательские, могут использовать шаблоны и категории. Все пространства имен используют префикс, например, страницы с описанием разных шаблонов имеют префикс «Шаблон», пример страницы в этом пространстве имен: «Шаблон:Карточка компании». Таких пространств имен в Википедии существуют десятки, например, к ним относятся пространства имен: «Википедия», «Справка», «Категория». По умолчанию поисковая система Википедии ограничена пространством имен статей. Однако ввод имени префикса пространства имен, за которым следует двоеточие, ограничивает результаты поиска конкретным пространством имен.

Статьи

Статья или запись в Википедии — это страница с энциклопедической информацией. Хорошо написанная статья в энциклопедии идентифицирует заметную энциклопедическую тему, всесторонне рассматривает эту тему, содержит ссылки на достоверные источники и ссылки на другие связанные темы. Большинство статей состоят

из абзацев и изображений, но они также могут быть отформатированы как автономные списки или таблицы. Эти списки или таблицы также считаются статьями. Движок Медиа-Вики считает статьями все страницы основного пространства имен, не являющиеся страницами перенаправления и содержащие хотя бы одну внутреннюю ссылку. При этом движок не различает, куда ведут эти ссылки, не умеет отличать ссылки на статьи и ссылки на обсуждения и правила Википедии и не делает различий между ссылками на существующие и несуществующие страницы. Такой механизм работает при подсчете общего количества статей, а также при генерации различных служебных отчетов.

Все статьи Википедии должны удовлетворять каждому из трех критериев [354]:

- не менее 150–300 знаков видимого текста с пробелами;
- наличие определения термина статьи;
- наличие хотя бы одного нетривиального факта, имеющего прямое отношение к предмету статьи помимо определения.

Также существуют и дополнительные частные требования по оформлению к некоторым классам статей, например, существуют свои требования для статей о литературных произведениях или музыкальных релизах. Эти требования касаются названия статьи и списка обязательных сведений об описываемом объекте.

Помимо требований существуют и рекомендации к оформлению. Сюда относится множество рекомендаций по стилю, написанию скобок, знаков препинания и других элементов оформления. Например, статьи русскоязычной Википедии должны писаться на литературном русском языке. Но все же главным требованием остается соблюдение структуры Википедии, которой стоит уделить особое внимание.

Каждая статья разбита на разделы – фрагменты, освещающие различные вопросы. Разделы позволяют структурировать статью. Это не только упрощает чтение статьи и получение необходимой информации, но также позволяет создавать ссылки на конкретные разделы, а не на начало всей статьи. Для этого в конце ссылки на статью нужно использовать символ «#», а после него указать имя раздела. Например, ссылка «Россия#Происхождение_названия» указывает на раздел «Происхождение_названия» в статье «Россия».

Для структуры Википедии особенно важна практика использования внутри статей ссылок и других объектов, формирующих связи

между статьями. В частности, к статье могут прилагаться внутренние ссылки, которые должны располагаться в специальном разделе «См. также», который, впрочем, не является обязательным. В этот раздел могут быть включены ссылки на статьи Википедии, тематически связанные с данной статьей, на которые еще нет ссылок в основном тексте статьи.

В основном тексте статьи могут присутствовать сноски. Текст сносок должен отображаться в разделе «Примечания». Сноски могут содержать как комментарии, так и ссылки на внешние источники, на которых основаны утверждения статьи. Сноски позволяют показать связь между текстом в статье и источниками, находящимися за пределами Википедии.

При наличии большого количества разнородных источников, рекомендуется создавать в статье раздел «Литература», куда будут включены ключевые печатные издания и публикации. Ссылки на источники, перечисленные в разделе «Литература», в примечаниях могут приводиться в сокращенном формате. При необходимости в конце статьи в разделе «Ссылки» даются ссылки на другие электронные и печатные информационные ресурсы.

Категории, в которые включается статья, должны располагаться в конце текста. Если категорий несколько, то ссылка на каждую из категорий приводится на отдельной строке. Порядок следования ссылок на категории должен быть единым у всех статей с одинаковой принадлежностью к категориям.

Для создания статей рекомендуется использовать шаблоны. Википедия содержит огромное количество шаблонов, которые упрощают создание страниц и позволяют спроектировать информацию. Например, в самом начале статьи при необходимости могут располагаться стандартные шаблонные пометки о выставлении статьи на удаление, переименование, объединение, о возможном нарушении нейтральности, о продолжающейся в данный момент работе над статьей и т. д. Подробнее о шаблонах будет рассказано далее в специальном подразделе.

Помимо обычных страниц со статьями в Википедии существуют и различные технические страницы. Например, они могут рассказать о самой Википедии, о том, как создавать категории или страницы, дать информацию о правилах и рекомендациях, предоставить статистику. Кроме того, в Википедии имеется большое количество страниц обсуждений. Для каждой статьи автоматически создается своя

страница обсуждения, на которую можно перейти по вкладке «Обсуждение» сверху страницы. На таких страницах обычно обсуждают возможные пути улучшения содержания, решают споры касательно изменения текста статьи, стремятся повысить качество страницы.

Категории

Каждая страница в пространстве имен статей должна относиться по крайней мере к одной категории. Категории статей позволяют классифицировать их по различным признакам. Обычно статья не должна быть как в категории, так и в ее подкатегории. Например, статья «Microsoft Office» находится в категории «Программное обеспечение Microsoft», поэтому ее не надо помещать в категорию «Программное обеспечение». Это правило может нарушаться, когда статья определяет категорию, а также находится в более высокой категории. Например, страница «Огайо» находится в обеих категориях: «Штаты Соединенных Штатов» и «Категория:Огайо».

Каждая категория должна быть подкатегорией какой-либо другой категории. Вы можете добавить новую категорию в более общую категорию верхнего уровня. Если же есть понимание, что более хороший родитель, вероятно, существует, но его не удастся найти, то можно добавить к категории тег `{{Uncategorized}}`. Тогда данная категория будет отображаться на странице некатегоризованных категорий.

Структуру категорий можно изобразить в виде ориентированного графа, в котором вершины соответствуют категоризируемым страницам или категориям, а ребра соответствуют включениям страниц в категорию. Ребра характеризуются критерием включения: каждому ребру можно поставить в соответствие тот или иной вид семантической связи, заданный в критерии родительской категории. Все новые категории добавляются к уже существующему графу категорий. При этом, нужно помнить, что категории, как и статьи, могут иметь сразу несколько родительских категорий. Начальной точкой, вершиной графа категорий, к которой мы будем приходить из любой категории, переходя все выше к родительским категориям, является категория «Все». Существует страница в Википедии «Дерево категорий», с ее помощью для любой категории можно построить дерево всех категорий, включенных в данную категорию.

Подход, когда категории устроены иерархически, но при этом у категорий (и у статей) может быть несколько «родительских» катего-

рий, позволяет использовать преимущества как классических таксономий, так и классификации на основе облака тегов (флаксономии). Как и в классических таксономиях, иерархические категории позволяют строить глубокие иерархии понятий, что не доступно для облака тегов. Однако, в отличие от классической таксономии каждая статья или категория может быть отнесена ко множеству категорий, а не только к одной родительской категории. В этом иерархические теги близки к концепции классификации на основе облака тегов.

Еще одним важным преимуществом категорий является автосвязывание. Достаточно создать ссылку на категорию на странице статьи, и соответствующая ссылка на эту статью будет видна на странице категории. Таким образом, отнесение к категории аналогично построению двунаправленной ссылки между страницей и ее категорией. Во многих случаях такое двунаправленное связывание значительно удобнее классических однонаправленных гипертекстовых ссылок.

Но у механизма категорий, реализованного в Википедии с помощью Викимедиа, есть и недостатки. Категории нельзя редактировать непосредственно. Для редактирования состава категории нужно переходить на статьи, входящие в категорию. Например, для того чтобы исключить статью из категории, нужно перейти на нее из категории и удалить в самой статье привязку к категории. Из-за этого категорию с сотнями статей нельзя переместить, кроме как отредактировав сотни статей. Отслеживание изменений в категории затруднено, потому что история редактирования категорий не формируется. Поэтому нет простого способа узнать, когда статья была удалена из категории — она просто исчезает из категории без указания того, что она там когда-либо была. Кроме того, категории не отображаются в мобильном режиме просмотра Википедии. И в обычном браузере отображение элементов в категории ограничено двумястами ссылками на странице. Поэтому, чтобы увидеть полное содержимое категории с большим количеством объектов, нужно просматривать несколько страниц.

Однако категории являются очень хорошим инструментом для поиска информации по конкретной теме и позволяют быстро собрать большой объем информации. Например, пройдя всего несколько шагов от статьи о какой-либо организации в Википедии вверх по дереву категорий, можно прийти к категории «Организации по алфавиту» и из этой категории получить ссылки на страницы всех организа-

ций, представленных в Википедии. С помощью данной категории можно получить список всех страниц, касающихся экономических субъектов: компаний, предприятий, заводов и пр. В данной категории русскоязычной Википедии находится около 20000 страниц, которые, если знать про такой инструмент, как категории, находятся очень быстро. Найти же эти же 20000 страниц вручную среди 5,5 млн страниц русскоязычной Википедии было бы очень сложно.

Для отнесения страниц к категориям используется не только указание категорий, к которым относится страница, но и специальные шаблоны. Так, например, в приведенном выше примере большинство компаний на своих страницах используют уже готовый шаблон «Карточка компании», который позволяет структурировать информацию о компании. В случае, если создатель забудет указать нужную категорию, но использует шаблон «Карточка компании», страница все равно попадет в категорию «Организации по алфавиту».

Шаблоны

Шаблон – это вики-страница, которая является заготовкой оформления текста (текста, рамки, изображения, изменяемого текста) для вставки в создаваемую вики-страницу. Шаблоны являются мощным средством, помогающим улучшить организацию Википедии. Существует несколько основных классов шаблонов [355], к ним относят:

- 1) общие;
- 2) служебные;
- 3) тематические;
- 4) юзербоксы;
- 5) навигационные;
- 6) карточки.

К общим шаблонам (а) относят технические пометки, категории, ссылки на другие проекты, предупреждения, шаблоны для пометок о незавершенных статьях и другие.

К служебным (б) – шаблоны, применяемые при обсуждениях и голосованиях, для наполнения информационных блоков на главных страницах и другие.

Тематические шаблоны (в) – это шаблоны, специально подготовленные для отображения данных по конкретным темам, например, география, спорт и другие.

Юзербоксы (г) — это специальный шаблон, размещающийся на странице конкретного участника ресурса. Такие шаблоны могут использоваться для указания определенных технических навыков, интересов, статуса участника или для поднятия настроения тем, кто посетил его страницу.

Навигационные шаблоны (д) представляют собой группировку ссылок, используемых в нескольких связанных статьях, для облегчения навигации между этими статьями. Существуют два основных варианта шаблона навигации: навигационные блоки, предназначенные для размещения в самом низу статей, и боковые панели, предназначенные для размещения сбоку от статьи. Эти шаблоны являются взаимодополняющими. Навигационные блоки лучше подходят для более длинных списков ссылок в небольшом количестве подкатегорий. Навигационные блоки должны создаваться только тогда, когда будут действительно полезны в качестве навигационных инструментов. Боковые панели располагаются преимущественно вертикально и размещены относительно заметно в теле статей наряду с текстом. Это делает их полезными для относительно небольшого количества релевантных ссылок.

Карточка (е) представляет собой инфобокс — панель, расположенную обычно в правом верхнем углу статьи, рядом с ведущим разделом, в которой суммируются ключевые факты статьи в разрезе вопросов, характерных для страниц данной тематики. Инфобокс может также включать изображение и карту. Его цель — суммировать ключевые факты, которые фигурируют в статье. Чем меньше содержащейся в нем информации, тем эффективнее она служит этой цели, что позволяет читателям сразу выявлять ключевые факты.

Шаблоны очень полезны, когда нужно получить краткую и важную информацию о статье в достаточно хорошо структурированном виде. Шаблоны в Википедии во многих случаях могут являться своего рода тегам, указывающими на присутствие у статьи каких-то характеристик. А такие шаблоны, как карточки, позволяют извлекать ключевую информацию из статей с использованием самых простых методов анализа. Несмотря на внушительное количество шаблонов в Википедии, при работе со статьями определенной тематики обычно удается выявить относительно небольшое количество наиболее ценных шаблонов, использование которых позволит весьма легко повысить эффективность извлечения структурированной информации из интересующих статей.

Это очень ценно при сборе информации из Википедии, особенно при использовании автоматизированных технологий сбора информации.

5.3. Википедия как источник, описывающий окружающую действительность

Википедия – это ресурс, который используется везде и всеми. За счет огромного количества ссылок он является наиболее связным веб-сайтом на планете. Используя результаты работы ученых, преподавателей, журналистов, отраслевых специалистов, студентов и других людей, обладающих знаниями в каких-либо отраслях, она отражает коллективное знание обширной сети экспертов о самых различных вопросах. Всего за почти 17 лет Википедия стала одним из крупнейших результатов совместных усилий в истории человечества. Согласно статистике с сайта <https://www.alexa.com>, который занимается оценкой популярности сайтов в глобальном масштабе, Википедия занимает 5 место в мире по популярности среди веб-ресурсов [356]. По популярности Википедию опережают только: google.com, youtube.com, facebook.com и baidu.com.

В Википедии на сегодняшний день можно найти почти любую информацию в достаточном удобном виде, и при этом точность такой информации будет весьма высока. За счет того, что Википедия достаточно хорошо структурирована, из нее относительно просто извлекать информацию об окружающем мире, причем в том числе и с использованием автоматизированных технологий.

Википедия гораздо более структурирована, чем Всемирная паутина (WWW) в целом, но далека от идеала. На самом деле, идеал в данной области, как и во многих других областях, по-видимому не достижим. Чем больше различной информации предоставляет ресурс, тем сложнее его структурировать, если это вообще возможно, т. е. при попытках создания существенно более структурированного ресурса об окружающем мире, сложности по структурированию информации непременно скажутся на объеме содержащейся в нем информации, и по объему знаний о мире такой ресурс будет уступать не только всемирной паутине в целом, но и Википедии.

В Википедии предпринято очень много организационных и технологических усилий и вложено много живого труда людей для того, чтобы работа с информацией в ней была удобной, и она была хоро-

Глава 5

шо организованным и структурированным источником знаний. Это все помогает держать Википедию между двумя крайностями: огромным потоком очень слабо структурированной информации, какой представляет из себя Всемирная паутина, и ресурсами максимально структурированными, но содержащими очень незначительный объем информации об окружающем мире.

Глава 6

ТЕХНОЛОГИИ СЕМАНТИЧЕСКИХ СЕТЕЙ И ИХ ПРИМЕНЕНИЕ

6.1. Введение в семантику и Semantic Web

Введение в семантику

Начиная с формальных определений, нужно сказать, что термин «семантика» весьма многозначен, при этом все большее количество значений этого термина формируется за счет развития информационных технологий и конкретно семантических технологий. Семантика — это и раздел языкознания, изучающий смысловое значение единиц языка, и дисциплина, изучающая формализацию значений конструкций языков программирования. Формальная семантика занимается изучением семантики, или интерпретации, формальных и естественных языков путем их формального описания в математических терминах. В то время как общая семантика рассматривает проблематику взаимодействия людей с миром через призму «абстракций» (напр., языковых конструкций), которые могут ограничивать и искажать наше описание реальности, наши гипотезы и теории о реальности по сравнению с объективно существующей реальностью. Несмотря на такое многообразие разделов научного знания, использующих в своем названии термин «семантика», все они в своей сути действительно опираются на значение древнегреческого слова *σημαντικός* — «обозначающий». Все эти направления исследуют значения, смысл, формальное описание языковых единиц, относящихся как к естественным, так и к искусственным язы-

кам, и проблематику соотнесения этих смыслов друг с другом и с объективной реальностью.

Возвращаясь от значений термина «семантика» к семантическим технологиям, мы можем наблюдать очевидную родовую взаимосвязь между разделами знания, использующими общий корень в названии. Действительно, так как семантические технологии ориентированы на создание возможности оперировать значениями (смыслами) отдельно от сырых данных и программного кода, то по сути они дают возможность компьютерным системам перейти от оперирования данными и программным кодом к оперированию смыслами.

Таким образом можно заключить, что семантические науки формируют возможность проникнуть в смысл языковых конструкций, формализуют процесс, который дан человеку естественным путем, в то время как с помощью семантических технологий компьютеры получают возможность приблизиться к человеческим возможностям понимания смысла информации, построения логических заключений и обмена смысловыми, а не информационными сообщениями. Семантические технологии предоставляют информационным системам более высокий уровень абстракции, который позволяет им гибко интерпретировать смысловое содержание информации, связывать данные и процессы, адаптируясь к различным источникам информации, различным субъектам обмена информации и изменениям в форме информационных сообщений.

Далее в этом подразделе мы рассмотрим некоторые базовые термины и понятия, используемые как в семантических науках, так и в семантических технологиях. Начнем это рассмотрение с акта коммуникации. Цель любой коммуникации – передать информацию об объекте или ситуации в реальном мире. Рассмотрим типичную ситуацию: преподаватель на лекции объясняет студентам новый материал, который в общем виде можно представить как некую смысловую конструкцию. Так как на данный момент передать смысловые конструкции напрямую из мозга одного человека в мозг другого невозможно, преподавателю приходится формулировать эту конструкцию в виде последовательности высказываний на некотором языке (это может быть как естественный язык, так и искусственный формальный язык, например, язык математики). Правила формирования высказываний этого языка определены в виде некой последовательности знаков. Последовательность знаков уже легко можно передать от одного человека другому с помощью письменнос-

ти или речи. После приема сформулированного высказывания получатель сообщения восстанавливает из него смысловую информацию с помощью семантики. Таким образом, семантика языка – это способ сопоставления некоторого знака с объектом реального мира или ситуацией из некоторой предметной области (домена). В свою очередь, денотат – это объект или ситуация, который соотносится со знаком (рисунок 55) [357].



Рис. 55. Диаграмма, показывающая взаимоотношения знака, домена и денотата

Но для определения денотата среди всего множества объектов или ситуаций необходимо понимать смысл знака. Концепт (понятие, смысл, десигнат, интенционал) – это некоторый набор условий, которым должен удовлетворять объект или ситуация, чтобы соотноситься со знаком. Таким образом, когда объект или ситуация удовлетворяет условиям концепта, то сопоставляется со знаком, т. е. становится денотатом (рисунок 56).



Рис. 56. Диаграмма, показывающая взаимоотношения знака, концепта и денотата

Однако, не во всех языках концепты явно и формально определены. Во многих случаях возникают ситуации, когда концепт невозможно определить. Например, молодой человек пригласил девушку на танец. В данном случае во время танца информация передается при помощи последовательности движений, которые в данной ситуации являются знаками. Если попытаться подобрать для этих знаков доступные формальные концепты, то скорее всего нам удастся определить вид танца, который исполняет пара. Таким образом, денотатом окажется вид танца, а использованными формальными концептами — описания видов танца.

Однако несмотря на то, что вид танца определен верно, сомнительно, что во время танца именно эту смысловую информацию молодой человек хотел передать девушке. Скорее всего, он хотел передать ей денотат, относящийся к домену «эмоции» (рисунок 57). Однако к этому денотату не подходит концепт, использованный для определения вида танца. Более того, очень сложно (или невозможно) формально определить концепт, который подходит к данному смыслу передаваемой информации. Таким образом, в данном случае мы видим пример языка с неявной семантикой. Формально говоря, языки с неявной семантикой — это языки, концепты для которых не определены [358], т. е. в случае этих языков человек интуитивно понимает их концепты, однако не может сформулировать их для объяснения другим людям и тем более — машинам.

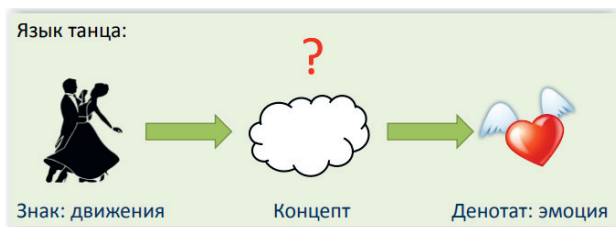


Рис. 57. Концепт с неявной семантикой

Помимо языков с неявной семантикой существуют языки с явной неформальной семантикой: концепты знаков этих языков явно определены, однако это определение дано не с помощью формального языка. Данные языки машины могут понимать, однако каждый концепт необходимо программировать вручную. Например, для открытия страницы сайта браузеру необходимо понимать язык разметки

HTML. Для языка HTML существует стандарт, однако он сформулирован на естественном языке, и машина не способна самостоятельно прочесть определения из этого стандарта. Для этого человеку необходимо запрограммировать браузер, таким образом переведя неформальное определение в формальный программный код, который уже позволяет машине работать с языком HTML (рисунок 58). Однако, если в HTML будут добавлены новые теги и эти теги будут использоваться на странице сайта, то браузер не сможет корректно работать с новыми тегами, пока его не перепрограммируют.

Среди языков с явной семантикой есть и языки с явной формаль-

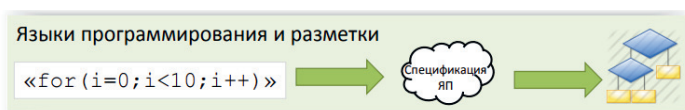


Рис. 58. Концепт с явной неформальной семантикой

ной семантикой. Концепты знаков этих языков явно определены на метаязыке с явной семантикой. Для данных языков машины могут самостоятельно прочесть и понять определение концептов. Это значит, что машину можно «научить» работать с концептами этого языка напрямую. Следствием такой работы будет то, что при появлении в языке новых концептов не будет требоваться вмешательство человека, машина сама сможет с помощью нового концепта сопоставить знак и денотат. Примером языка, с помощью которого можно формально описывать концепты, является язык логики и математики (рисунок 59). Компьютер можно запрограммировать ав-



Рис. 59. Концепт с явной формальной семантикой

томатически интерпретировать выражения на этих языках и таким образом самостоятельно осваивать концепты языка с явной формальной семантикой.

Семантика во Всемирной паутине и в Семантической паутине

Семантическая сеть — это модель предметной области в виде ориентированного графа, где вершины — объекты предметной области, а ребра — отношения между ними.

Semantic Web, или по-русски Семантическая паутина, впервые была описана Тимом Бернесом-Ли, изобретателем Всемирной паутины (World Wide Web — WWW), и была предложена им как «следующий шаг в развитии Всемирной паутины» [359]. Семантическая паутина — это общедоступная глобальная семантическая сеть, которая формируется на базе Всемирной паутины с помощью стандартизации представления информации в виде, пригодном для машинной обработки [360].

Если говорить про Всемирную паутину, то она основана на html-страницах, где информация приспособлена для чтения и понимания человеком. Несмотря на то, что информация, находящаяся во всемирной паутине, очень ценна для разработчиков и пользователей информационных систем, заставить программы «понимать» содержимое html-страниц очень непросто. Так как данные во Всемирной паутине обычно представлены в виде текста на естественном языке, машина может понять их смысл, только используя один из методов обработки естественного языка. Несмотря на прогресс в методах автоматической обработки текстов на естественном языке, данные технологии все еще очень сложны, малоэффективны и ненадежны.

Одна из ключевых проблем «понимания» текста на естественном языке связана с тем, что люди в большинстве случаев интуитивно могут понять концепт, однако машина сделать это не способна. Если на одной из веб-страниц разместить иносказательное описание белого зайца, то человек поймет, что речь идет о белом зайце, а машина, без специальной подсказки, оформленной, например, в виде специальных тегов, хранящих метаинформацию (информацию об информации), сделать этого не сможет. Поэтому до сих пор существенного прогресса в прямом использовании информационными системами произвольной информации из Всемирной паутины добились лишь некоторые крупные компании, лидирующие в области высокотех-

нологичной обработки сверхбольших объемов информации. И даже эти компании используют свои технологии только для весьма узкого круга задач, например, для создания поисковых сервисов для людей, ищущих информацию в Интернете.

Идея Семантической паутины заключается в том, чтобы вместо того, чтобы пытаться учить машину читать Всемирную паутину, информация в которой сформулирована в виде, удобном только для чтения людьми, начать формировать и хранить информацию во Всемирной паутине так, чтобы читать ее стало удобно не только человеку, но и машине. Технологии Семантической паутины предназначены для того, чтобы в дополнение ко Всемирной паутине, можно сказать «поверх» нее, построить Семантическую паутину, которая состоит из машинно-читаемых элементов, образующих семантическую сеть.

Формально семантическая сеть — это модель предметной области в виде ориентированного графа, где вершины — это объекты предметной области, а ребра — отношения между ними. В Семантической паутине вершинами семантического графа являются URI (унифицированные идентификаторы ресурса или адреса), которые однозначно обозначают, именуют некоторые объекты. Хорошо нам известным частным случаем URI является URL, которые мы используем в качестве адресов веб-ресурсов. В отличие от URL, URI создают не только для страниц, но и для объектов реального мира (людей, городов, художественных произведений и так далее) и даже для абстрактных понятий (например, таких как «имя», «должность», «цвет»). Благодаря уникальности URI одни и те же предметы можно называть одинаково в самых разных местах Семантической паутины.

С помощью Семантической паутины у программ появится возможность считывать из содержимого Всемирной паутины утверждения в виде «предмет—вид взаимосвязи—другой предмет» и выводить из этого логические заключения. С помощью семантических технологий, которые позволяют обрабатывать и интерпретировать (по сути, понимать) большие массивы таких утверждений, у информационных систем, работающих в Семантической паутине, появится ряд принципиально новых возможностей, таких как:

- Семантический поиск — поиск по смыслу, а не по ключевым словам. В качестве запросов к таким системам можно будет давать сложные условия, например: «Ноутбук с объемом памяти больше 8 ГБ и не дороже 60000 рублей».

- Объединение данных – поиск ответов на запросы, которые отсутствуют во всех источниках по отдельности, но присутствуют в их совокупности. Например, в качестве ответа на запрос: «побочные эффекты лекарства от туберкулеза», – будет агрегирована информация из таких источников, как: DrugBank, Diseasesome, Spider.
- Логический вывод – вычисление новых знаний на базе уже имеющихся. Например, если в исходной информационной базе есть следующая информация: Виктория – мать Эдуарда VII, Эдуард VII – отец Георга V, Георг V – отец Георга VI, Георг VI – отец Елизаветы II, то система сможет найти праправнуков королевы Виктории (Елизавета II – праправнучка Виктории).
- Интеллектуальный агент – программа, которая способна автономно выполнять указанное задание по поиску и обработке информации. Например, «Siri, закажи мне столик в лучшем японском ресторане в Москве». На данный момент такие сервисы (Apple Siri, Google Assistant, Amazon Alexa, Microsoft Cortana, Яндекс.Алиса) уже разрабатываются и тестируются компаниями-лидерами в области интеллектуальной обработки сверхбольших объемов данных.

Несмотря на то, что вполне эффективные технологии семантической разметки для Всемирной паутины появились более десяти лет назад, скорость реального развития Семантической паутины все еще невысока. Во многом это связано с тем, что превращение веб-ресурса из ресурса, предназначенного только для людей, в ресурс, которым могут эффективно пользоваться и люди, и машины, достаточно сложно: по сути одну и ту же информацию, представленную на веб-ресурсе, приходится дублировать, представляя ее и в формате, удобном для людей, и в машиночитаемом формате. Затраты на поддержание такой более сложной структуры веб-ресурса достаточно высоки, при этом выгода от того, что веб-ресурс станет машиночитаемым, совсем не очевидна.

Областью, в которой вложения авторов контента Всемирной паутины в обеспечение машиночитаемости стали окупаться, оказалась поисковая оптимизация (или SEO, от английского Search Engine Optimization – оптимизация (веб-ресурса) для поисковых машин). Большинство авторов контента очень заинтересованы в том, чтобы их веб-ресурсы хорошо находились в поисковых сервисах (т. е. ока-

зывались в начале поисковых выдач на запросы пользователей, релевантные теме веб-ресурса). При этом поисковые сервисы заинтересованы в том, чтобы по поисковым запросам пользователям предоставлялась наиболее релевантная информация. Для решения этой задачи разработчикам поисковых сервисов приходится учить свои машины читать страницы, написанные для людей, чтобы понять их смысл и в соответствии с этим пониманием предоставлять ответы на поисковые запросы. И здесь идея Семантической паутины оказывается как нельзя кстати — и авторы страниц, и разработчики поисковых сервисов готовы вкладывать дополнительные усилия для обеспечения более релевантных результатов поиска.

В 2011 г. разработчиками крупнейших поисковых сервисов (в том числе Google, Yahoo!, Microsoft Bing, Yandex) была запущена прикладная семантическая технология Schema.org. Эта технология позволяет разработчикам веб-сайтов с помощью HTML5 просто и эффективно выполнять семантическую разметку своих страниц. Основной целью Schema.org является помощь разработчикам контента Всемирной паутины в создании качественных метаданных, что, в свою очередь, позволит улучшить качество поиска. В качестве основного формата разметки веб-страницы метаданными разработчики schema.org предлагают microdata (микроданные) — теги и атрибуты для разметки структурированной информации на веб-страницах, появившиеся в стандарте HTML5. Пример использования таких метаданных предложен на рисунке 60, где жирным шрифтом отмечены теги,

```
<div itemscope itemtype
="http://schema.org/Movie">
  <h1 itemprop="name">Avatar</h1>
  <span>Director:           <span
itemprop="director">James   Cameron</span> (born
August 16, 1954)</span>
  <span itemprop="genre">Science fiction</span>
  <a href=" ../movies/avatar-theatrical-
trailer.html" itemprop="trailer">Trailer</a>
</div>
```

Рис. 60. Пример семантической разметки контента html-страницы с помощью словаря schema.org

связанные с разметкой содержимого html-страницы метаданными. Метаданные на сайтах, использующих схемы, описанные на Schema.org, могут быть напрямую проанализированы поисковыми роботами, помогая последним лучше «понимать» содержимое веб-ресурсов. По сути использование метаданных Schema.org является простым и наглядным примером того, как Всемирная паутина может становиться более проста для понимания машинами и таким образом приближаться к концепции Семантической паутины.

6.2. Стек технологий Семантической паутины

Прежде чем мы перейдем к подразделам, описывающим конкретные семантические технологии, обзорно рассмотрим эти технологии в рамках всего современного технологического стека Семантической паутины. Вид технологического стека представлен на рисунке 61.

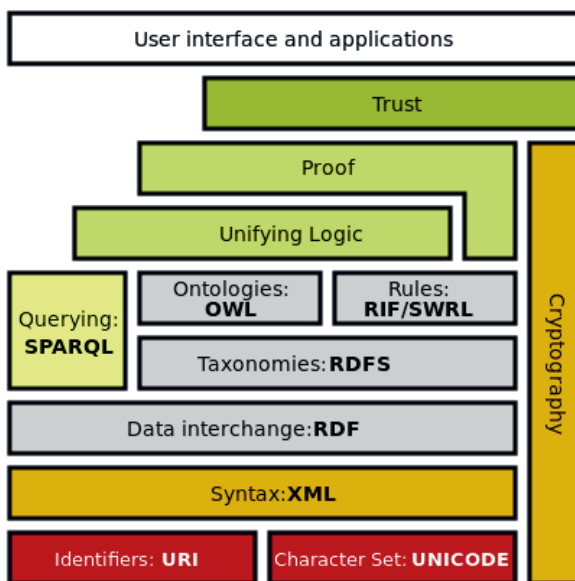


Рис. 61. Стек технологий Семантической паутины

Как видно из рисунка 61, в базе технологического стека Семантической паутины находятся такие универсальные технологии (стандарты), как Unicode, URI, XML. Стандарт Unicode обеспечивает воз-

возможность использования символов практически всех языков мира при записи текста на цифровых носителях и на данный момент доминирует в Интернете. Для Семантической паутины Unicode обеспечивает возможность формировать записи, содержащие информацию практически на любом языке мира.

Как говорилось выше, в Семантической паутине URI (унифицированные идентификаторы ресурса или адреса) используются для обозначения вершин семантического графа. URI однозначно обозначают, именуют, объекты или явления, при этом частным случаем URI являются URL, которые мы используем в качестве адресов веб-ресурсов. URI создают не только для страниц, но и для объектов реального мира (людей, городов, художественных произведений), и для абстрактных понятий (например, «имя», «должность», «цвет»). Благодаря уникальности URI одни и те же предметы можно называть одинаково в самых разных местах семантической паутины.

В более поздних версиях стандартов Семантической паутины вместо URI принято использовать IRI (Internationalized Resource Identifier). IRI представляют собой обобщение URI, которое допускает использование более широкого диапазона символов Unicode, что упрощает именование объектов на различных языках. Для удобства и единообразия далее мы будем говорить об URI даже в тех случаях, когда в более современных стандартах вместо него может использоваться IRI.

XML (eXtensible Markup Language – расширяемый язык разметки) – язык, определяющий базовый синтаксис для формирования структуры различных типов документов, используемых в Семантической паутине. XML создан как язык с простым синтаксисом, удобный для создания и обработки документов и программами, и человеком с нацеленностью на использование в Интернете. XML называется расширяемым языком, потому что на его базе легко создать собственную разметку документа для формирования типа документа, относящегося к хранению и передаче информации в конкретной предметной области. Нужно отметить, что многие технологии Семантической паутины наряду с XML-синтаксисом поддерживают альтернативные виды записи, более удобные для некоторых практических применений. При этом XML-запись документа и альтернативные форматы обычно являются полностью взаимозаменяемыми, и существует простой способ перекодирования одного формата файла в другой.

Выше в технологическом стеке представлены технологии, непосредственно относящиеся к Семантической паутине. Собственно, технологии Семантической паутины представлены в виде уровней.

Самый базовый уровень Семантической паутины представлен стандартом RDF (Resource Description Framework – среда описания ресурса), описывающим способ формирования модели для представления данных (в особенности – метаданных) в виде, пригодном для машинной обработки. RDF обеспечивает запись утверждений в виде триплетов: <субъект> <предикат> <объект>. Например, утверждение «небо голубого цвета» в RDF-терминологии можно представить, как: субъект – «небо», предикат – «имеет цвет», объект – «голубой». Для обозначения субъектов, отношений и объектов в RDF используются URI. Для записи и передачи RDF используется как XML-формат, так и JSON, RDFa (RDF in attributes – запись внутри атрибутов HTML-документа), компактные формы записи, удобные для обработки человеком, такие как N3 и Turtle.

Чтобы перейти к описанию более высокоуровневых технологий Семантической паутины, нужно понимать, что RDF только предоставляет способ построения информационных моделей в виде триплетов, но не касается семантики того, что описывается в документе. RDF-документ по сути описывает ориентированный граф, в вершинах которого находятся URI, а ориентированным связям приписаны предикаты. Таким образом, без дополнительного контента RDF-документ – это всего лишь граф. Толкование смысла RDF основывается на способности пользователей RDF (в том числе и программных комплексов) интерпретировать отдельные URI, строковые литералы и структуру графа, и по ним интерпретировать остальные URI и семантику данных. Для получения семантики RDF-документа требуется, чтобы он был связан со специальными словарями, таксономиями и онтологиями. Рассмотрим суть упомянутых сущностей:

- В контексте Семантической паутины словарь представляет собой собрание терминов, имеющих во всех контекстах использования этого словаря одинаковый смысл.
- В контексте Семантической паутины таксономия – это иерархически организованный словарь.
- Онтология в Семантической паутине использует предопределенный словарь терминов для определения концепций и отношений между ними в рамках конкретной предметной области.

Онтологии можно использовать для выражения семантики терминов словаря, их взаимоотношений и контекстов использования.

Теперь, имея представление о том, какими семантическими конструкциями нужно окружить RDF-документ, проще понять назначение технологий Семантической паутины, расположенных в технологическом стеке над RDF.

RDFS (RDF Schema) представляет собой словарь для RDF, который используется в основном при описании других словарей. RDFS тесно связан с RDF, в частности, описание дополнительной семантической нагрузки RDFS расположено в основных стандартах RDF. Словарь RDFS декларирует:

- существование ресурсов, литералы разных типов данных, в том числе строк различного вида;
- существование классов и подклассов ресурсов;
- свойства, которые относятся к ресурсам определенного класса;
- такие часто используемые свойства, как «наименование», «описание», «ресурс-синоним» и некоторые другие;
- классы и свойства, позволяющие организовать в модели RDF базовые виды списков;
- классы и свойства, необходимые для обеспечения возможности записи триплетов, объектом и субъектом которых являются другие триплеты.

Для записи более сложных логических отношений описываемых ресурсов различных предметных областей в онтологиях служит язык описания онтологий OWL (Web Ontology Language). OWL в целом похож на RDFS, но преследует иные цели: если RDFS используется для того, чтобы описать структуру данных, то OWL описывает семантические (смысловые) взаимосвязи между ресурсами. OWL представляет принципиально больший словарь для описания отношений между ресурсами. В частности, OWL позволяет описывать данные с помощью операций теории множеств (см. примеры на рисунке 62). OWL существенно более строгий язык описания взаимосвязей, который декларирует не только то, как можно связать ресурсы, но и как это делать нельзя. Например, нельзя определить ресурс одновременно как класс и как конкретный объект. В OWL есть выразительные средства для создания аннотаций и работы с мета-ме-

Рисунок 1.1 Example:Mother (Example:Parent, Example:Woman)	owl:unionOf
Рисунок 1.2 AcmeCompany:JohnSmith PersonalDatabase:JohnQSmith	owl:sameAs
Рисунок 1.3 Example:MyState (State:NewYork, State:California)	owl:allValuesFrom

Рис. 62. Записанные в нотации Turtle примеры триплетов, использующих OWL

тадными. Например, средствами OWL можно легко объединить несколько онтологий, описать устаревшую взаимосвязь или взаимосвязь, введенную для обратной совместимости. Таким образом, OWL является самым высокоуровневым языком, описывающим взаимосвязь между ресурсами уже семантическом уровне.

Так как технологический стек Семантической паутины фактически определяет парадигму представления данных, и в этой парадигме предполагается хранить большое (а в случае Семантической паутины в целом — крайне большое) количество данных, то естественно должен существовать и язык для формулировки запросов к этим данным. Таким языком является SPARQL (рекурсивный акроним от SPARQL Protocol and RDF Query Language).

6.3. RDF

RDF (Resource Description Framework) является частью концепции Семантической паутины [361]. Это модель для представления данных в виде утверждений о ресурсах, пригодных для машинной обработки. Данная модель разработана World Wide Web Consortium (W3C). RDF используется для:

- Добавления машиночитаемой информации на веб-страницы с использованием, например, популярного словаря Schema.org, позволяющего поисковым системам и другим приложениям извлекать и обрабатывать информацию из веб-страниц в более структурированном формате.
- Обогащения наборов данных путем их привязки к сторонним наборам данных. Например, набор данных о картинах можно обогатить, связав картины с создавшими их художниками

в Wikidata, тем самым предоставив доступ к широкому спектру информации об авторах картин и связанных с ними информационных ресурсах.

- Использования открытых взаимосвязанных наборов данных, использующих технологии, базирующиеся на RDF (такие данные принято называть связанными данными (linked data)), например, построение агрегированных данных по конкретным темам.
- Создания распределенных социальных сетей путем связывания RDF-описаний людей на нескольких веб-сайтах.
- Предоставления стандартного способа обмена данными между базами данных.
- Предоставления взаимной связи различных наборов данных внутри организации, что позволяет выполнять запросы для межсайтовых наборов данных с использованием SPARQL.

Базовая структура данных RDF – это набор триплетов, каждый из которых состоит из субъекта, предиката и объекта. Множество триплетов называют графом RDF. RDF имеет абстрактный синтаксис, отражая модель данных в виде графа, а формальная семантика позволяет совершать логические выводы на основе RDF-данных. В целом RDF основывается на следующих компонентах:

- Простая модель данных на основе графов.
- Формальная семантика, поддерживающая логические выводы, с детально описанными отношениями следствия – позволяет делать логические рассуждения на основе RDF-данных.
- Расширяемые словари на основе URI.
- Синтаксис на основе XML (опционально: JSON, RDFa, N3, Turtle).
- Использование типов данных XML-схемы.

Основой RDF является возможность делать утверждения о ресурсах. Формат этих утверждений прост. Оператор всегда имеет следующую структуру триплета: <субъект> <предикат> <объект>. Триплет выражает направленную связь между двумя ресурсами: субъектом (источником) и объектом (приемником) [362]. Предикат представляет характер их отношений, при этом отношение направлено от источника к приемнику. Эта взаимосвязь (триплет) называется RDF-свойством. RDF-триплет состоит из трех компонентов [363]:

- Субъект – URI (унифицированный идентификатор ресурса, представленный строкой) или пустой узел;

- Предикат – URI;
- Объект – URI, литерал (определение значений чисел, дат, строк посредством лексического представления) или пустой узел.

Примеры RDF-свойств приведены на рисунке 63. Набор RDF-свойств образует RDF-граф. RDF-граф – это набор RDF-триплетов, узлами RDF-графа являются объекты и субъекты триплета. На рисунке 64 приведена визуализация RDF-графа для рисунка 63. Один и тот же ресурс может употребляться в разных триплетах. В рассмотренном примере <Bob> 4 раза упоминается в качестве субъекта, а <the Mona Lisa> один раз в качестве субъекта и два раза в качестве объекта. Способность одного и того же ресурса быть как субъектом, так и объектом позволяет связывать триплеты в один граф.

```

<Bob> <is a> <person>.
<Bob> <is a friend of> <Alice>.
<Bob> <is born on> <the 4th of July 1990>.
<Bob> <is interested in> <the Mona Lisa>.
<the Mona Lisa> <was created by> <Leonardo da Vinci>.
<the video 'La Joconde à Washington'> <is about> <the Mona Lisa>

```

Рис. 63. Примеры триплетов

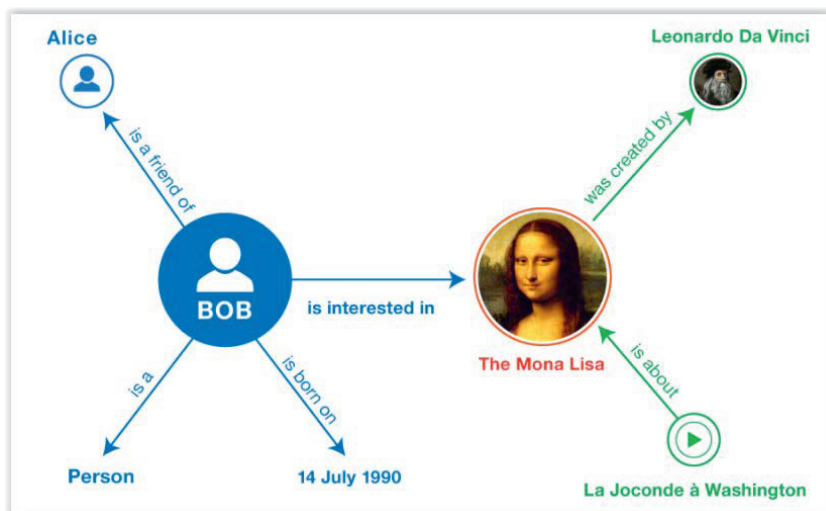


Рис. 64. Неформальный граф RDF

Существует три типа узлов: URI, литералы и пустые узлы. Литералы и URI обозначают что-то в этом мире. Узлы этих типов называются ресурсами. Все может быть ресурсом, включая физические вещи, документы, абстрактные понятия, числа и строки. Ресурс, обозначенный URI, называется его референтом, а ресурс, обозначенный литералом, называется его буквальным значением. Литералы имеют типы данных, которые определяют диапазон их возможных значений; например, могут использоваться такие типы литералов, как строки, числа и даты. В отличие от URI и литералов, пустые узлы не идентифицируют конкретные ресурсы. Состояния, связанные с пустыми узлами, говорят о существовании некоей сущности с данными отношениями, без явного именованя этой сущности.

В RDF URI (унифицированные идентификаторы ресурса или адреса) является строкой символов, которая используется для обозначения вершины семантического графа. URI однозначно обозначают, именуют объекты или явления. Частным случаем URI является URL, которые используются в качестве адресов веб-ресурсов.

В более поздних стандартах RDF предлагается вместо URI использовать IRI (Internationalized Resource Identifier). IRI в RDF-графе является строкой Unicode, которая записана согласно стандарту RFC3987. IRI представляют собой обобщение URI, которое допускает использование более широкого диапазона символов Unicode. Таким образом, в отличие от URI идентификатор IRI позволяет удобно записывать имена ресурсов на всех языках, поддерживаемых Unicode. Каждый абсолютный URI и URL является IRI, но не каждый IRI является URI. При использовании IRI в операциях, где необходимо использовать только URI, есть возможность преобразовать IRI в URI с использованием процедуры, описанной в RFC3987.

URI используются для идентификации ресурсов, таких как документы, люди, физические объекты и абстрактные понятия. Например, URI для Леонардо да Винчи в DBpedia: http://dbpedia.org/resource/Leonardo_da_Vinci. URI для видео о Моне Лизе под названием «La Joconde à Washington» в Europeana: <http://data.europeana.eu/item/04802/243FA8618938F4117025F17A8B813C5F9AA4D619>.

URI создают не только для страниц, но и для объектов реального мира (людей, городов, художественных произведений), для абстрактных понятий (например, «имя», «должность», «цвет»). Благодаря уникальности URI одни и те же предметы можно называть одинаково в самых разных местах семантической паутины.

В RDF литералы используются для таких значений, как строки, числа, даты. Фактически литерал — это строка с необязательной меткой языка. Литерал состоит из двух или трех (если есть метка языка) элементов:

- все литералы имеют лексическую форму в виде строки символов Unicode;
- простые литералы состоят из лексической формы и необязательной ссылки на язык (например: ru, en);
- типизированные литералы состоят из лексической формы и URI-ссылки на тип данных, задаваемой в формате RDF URI.

URI (IRI) и литералы вместе составляют основной материал для записи RDF-свойств. Однако, иногда удобно говорить о ресурсах, не пытаясь использовать какой-либо глобальный идентификатор. Например, мы могли бы заявить, что картина Мона Лиза (см. рисунок 65) имеет на заднем плане неопознанное дерево, которое в литературе принято обозначать, как кипарисовое дерево. Ресурс без глобального идентификатора, такой как кипарисовое дерево на картине, может быть представлен в RDF пустым узлом. Пустые узлы похожи на переменные в алгебре; они представляют собой некую сущность с не-

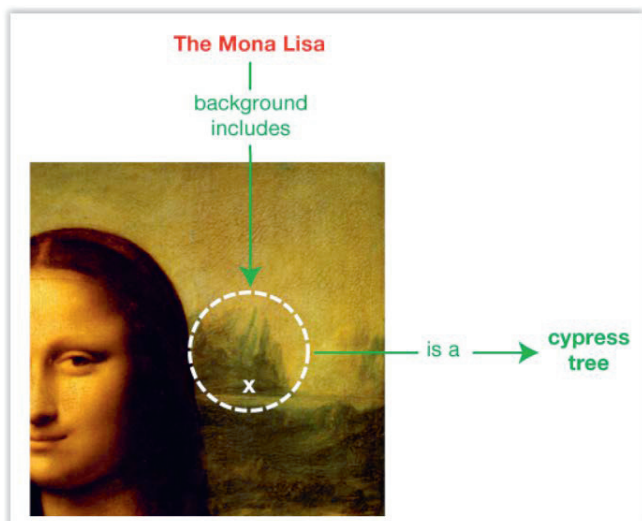


Рис. 65. Пример пустого узла

определенным значением. Пустые узлы в RDF-свойствах могут появляться как в позиции субъекта, так и в позиции объекта.

На базе RDF можно построить набор данных RDF, который представляет собой несколько RDF-графов и включает в себя:

- ровно один RDF-граф по умолчанию, этот граф не имеет имени и может быть пустым;
- ноль или более именованных графов, каждый именованный граф представляет собой пары: «URI или пустой узел (имя графа) – RDF-граф», причем имена графов должны быть уникальны в рамках набора RDF.

Несмотря на использование слова «имя» в «именованном графе», имя графа не требуется для обозначения графа. Оно просто синтаксически сопряжено с графом. RDF не устанавливает каких-либо формальных ограничений на то, какой ресурс может обозначать имя графа, а также на связь между этим ресурсом и графом.

6.4. OWL

OWL (Web Ontology Language) – язык описания онтологий, – это элемент Семантической паутины, который используется для описания классов и отношений между ними [364]. Язык OWL используется в следующих целях [365]:

- формализации предметной области (домена) с помощью определения классов и свойств этих классов;
- определения конкретных представителей описанных классов и их свойств.

По сути OWL – это расширение словарей RDF и RDFS. Для написания OWL-онтологий используется синтаксис и формальная семантика OWL. Поскольку Семантическая паутина распределена, OWL позволяет собирать информацию из распределенных источников. Имеется возможность объединять онтологии, например, с помощью импорта данных одной онтологии в другую онтологию. OWL предполагает открытость: описания ресурсов не ограничены количеством файлов или темой. В частности, класс, определенный в одной онтологии, может быть расширен в других онтологиях. При этом факты одной онтологии не могут опровергать другую онтологию, хотя онтологии могут быть противоречивыми.

Виды OWL

Существует несколько видов языков описания онтологий, входящих в общее определение OWL [366]:

- OWL Lite используется, когда в основном необходима классификационная иерархия и простые ограничения. Как и другие виды OWL-онтологий, OWL Lite поддерживает ограничения кардинальности. Как видно из названия, OWL Lite является облегченной версией OWL – OWL Lite легче поддерживать в различных приложениях;
- OWL DL поддерживает дескриптивную логику. OWL DL используется, когда необходима большая выразительность без потери вычислений и разрешаемости систем, строящих логические заключения;
- OWL Full поддерживает максимальную выразительность с синтаксической свободой RDF, однако не дает никаких гарантий на полноту вычислений. Данный вариант OWL позволяет создавать онтологии, которые расширяют состав предопределенного словаря (OWL или RDF).

В таблице 14 приведено сравнение различных языков описания онтологий семейства OWL.

Структура OWL

В данном подразделе будут представлены основные компоненты документов OWL.

Namespaces

Использование пространств имен (namespaces) позволяет однозначно интерпретировать идентификаторы в OWL-документах. Для этого используется синтаксис, принятый в XML-документах. Для объявления пространства имен используется компонент онтологии, который является набором объявлений XML namespace внутри открывающего тега `rdf:RDF`.

Объявления, приведенные в примере на рисунке 66, определяют следующие действия:

- говорит, что имена тегов относятся к текущей онтологии;
- объявляет пространство имен онтологии с приставкой `vin`;

Таблица 14

Сравнение различных языков описания онтологий семейства OWL

	OWL Lite	OWL DL	OWL Full
Совместимость с RDF	Теоретически, ни один документ RDF не может считаться совместимым с OWL Lite	Теоретически, ни один документ RDF не может считаться совместимым с OWL DL	Все RDF документы полностью заполнены в OWL
Ограничения на определение классов	Требуется разделение классов, экземпляров, свойств и значений данных	Требуется разделение классов, экземпляров, свойств и значений данных	Классы могут быть экземплярами и свойствами одновременно. Напр., в OWL Full вполне законно иметь идентификатор «Python», который действует как имя класса (обозначая список преподавателей, обучающихся этому предмету) и как индивидуальное имя (напр., экземпляр класса «Языки программирования»)
RDF-смешение	Ограничивает смешение конструкций RDF и OWL	Ограничивает смешение конструкций RDF и OWL	Свободно позволяет смешивать конструкции RDF и OWL
Метамоделирование	Не допускает метамоделирования	Не допускает метамоделирования	Допускает метамоделирование. Таким образом, конструкции RDF и OWL могут быть дополнены или переопределены
Классы	OWL класс является подклассом RDFs class	OWL класс является подклассом RDFs class	RDFS- и OWL-классы эквивалентны

```

<rdf:RDF
  xmlns      = "http://www.w3.org/TR/2004/REC-owl-guide-20040210/wine#"
  xmlns:vin  = "http://www.w3.org/TR/2004/REC-owl-guide-20040210/wine#"
  xml:base   = "http://www.w3.org/TR/2004/REC-owl-guide-20040210/wine#"
  xmlns:food = "http://www.w3.org/TR/2004/REC-owl-guide-20040210/food#"
  xmlns:owl  = "http://www.w3.org/2002/07/owl#"
  xmlns:rdf  = "http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:rdfs = "http://www.w3.org/2000/01/rdf-schema#"
  xmlns:xsd  = "http://www.w3.org/2001/XMLSchema#">
    
```

Рис. 66. OWL Namespaces

- определяет базовый URI (или IRI) документа;
- определяет вспомогательную онтологию с префиксом `food`;
- декларирует OWL, которая используется для ссылки на OWL, т. е. элементы документа, которые начинаются с «`owl:`», обращаются к понятиям в данном пространстве имен;
- определяет ссылку на конструкцию RDF;
- определяет ссылку на конструкцию RDFS;
- определяет ссылку на конструкцию XML Schema datatypes.

Headers

После создания пространств имен в OWL обычно включается набор утверждений об онтологии, сгруппированных по тегу `owl:Ontology` (см. пример на рисунке 67). Элементы внутри тега `owl:Ontology` используются для решения таких задач, как комментирование, контроль версий и включение других онтологий.

```
<owl:Ontology rdf:about="">
  <rdfs:comment>An example OWL ontology</rdfs:comment>
  <owl:priorVersion rdf:resource="http://www.w3.org/TR/2003/PR-owl-guide-20031215/wine"/>
  <owl:imports rdf:resource="http://www.w3.org/TR/2004/REC-owl-guide-20040210/food"/>
  <rdfs:label>Wine Ontology</rdfs:label>
  ...
```

Рис. 67. OWL Headers

Элемент «`owl:Ontology`» — это место, где собирается большая часть метаданных OWL. Это не гарантирует, что документ описывает онтологию в традиционном смысле. В некоторых случаях онтологии — это не конкретные сущности, а только классы и свойства, которые определяют предметную область. При использовании OWL для описания коллекции данных может понадобиться тег `owl:Ontology` для записи информации о версии документа и импорта определений, от которых зависит документ.

Атрибут `rdf:about` предоставляет имя или ссылку для онтологии. Если значение атрибута является пустым, то имя онтологии является базовым URI элемента `owl:Ontology`. Как правило, это URI документа, содержащего онтологию. Тег `rdfs:comment` — это тег, обеспечивающий возможность аннотировать онтологию.

Тег `owl:priorVersion` — это стандартный тег для обеспечения контроля версий. Так как онтологии изменяются со временем, в преде-

лах элемента owl:Ontology может быть указана ссылка на предыдущую версию онтологии. Для обеспечения связи существует свойство owl:priorVersion (см. пример на рисунке 68). Причем версии онтологий могут быть несовместимыми друг с другом.

```
<owl:Ontology rdf:about="">
...
<owl:priorVersion rdf:resource="http://www.w3.org/TR/2003/CR-owl-guide-20030818/wine"/>
...
</owl:Ontology>
```

Рис. 68. Версии онтологий

Основные элементы документа OWL

В данном подразделе будут рассмотрены основные элементы документа, описывающего онтологию. Начнем наше знакомство с этими элементами с простых именованных классов. Главным классом в OWL является owl:Thing, т. е. каждый конкретный экземпляр является членом этого класса. Соответственно, каждый класс, созданный пользователем, является подклассом owl:Thing. Класс объявляется при помощи синтаксической конструкции, пример которой приведен на рисунке 69.

```
<owl:Class rdf:ID="Регион"/>
```

Рис. 69. Создание класса в OWL

Определение класса содержит две части: название или ссылка и список ограничений. Каждое выражение, которое содержится в определении класса, ограничивает свойства. Конструктор rdfs:subClassOf связывает дочерний класс с базовым классом. Если X является дочерним классом класса Y, то представитель класса X является представителем класса Y, и так далее вверх по иерархии. Пример использования конструкции rdfs: subClassOf приведен на рисунке 70.

```
<owl:Class rdf:ID="Регион">
  <rdfs:subClassOf rdf:resource="#Город" />
  ...
</owl:Class>
```

Рис. 70. Создание дочернего класса в OWL

Представители или экземпляры класса — это описанные в OWL члены класса. Представителя класса можно создать несколькими способами, основной и альтернативный способ представлены на рисунках 71 и 72, соответственно. В альтернативном способе элемент `rdf:type` позволяет связать экземпляр с классом, членом которого он является.

```
<Регион rdf:ID="ЦентральноеПобережье" />
```

Рис. 71. Создание представителя класса в OWL

```
<owl:Thing rdf:about="#ЦентральноеПобережье">  
  <rdf:type rdf:resource="#Регион"/>  
</owl:Thing>
```

Рис. 72. Альтернативный способ создания представителя класса в OWL

Следующим рассматриваемым элементом являются свойства. В OWL свойства — это бинарные отношения. В OWL существует два типа свойств: свойства-значения — отношения между представителями классов и RDF-литералами; и свойства-объекты — отношения между двумя классами.

Свойства, как и классы, образуют иерархии при помощи элемента `rdfs:subPropertyOf`. Например, объявление, приведенное на рисунке 73, означает, что свойство «имеетЦвет» — частный случай свойства «имеетОписание». Кроме того, диапазон свойства «имеетЦвет» дополнительно ограничен с помощью класса «ЦветВина». Отношение `rdfs:subPropertyOf` в этом случае означает, что все, что имеет свойство «имеетЦвет» со значением X, также имеет свойство «имеетОписание» со значением X.

Картирование онтологий

Для достижения максимального эффекта от использования онтологий они должны быть широко распространены и пригодны для многократного использования [367]. Хорошим примером такого использования является случай, когда новая онтология строится на базе нескольких уже существующих онтологий. Например, можно воспользоваться онтологией даты из одного источника и онтологией

```
<owl:Class rdf:ID="ОписаниеВина" />

<owl:Class rdf:ID="ЦветВина">
  <rdfs:subClassOf rdf:resource="#ОписаниеВина" />
  ...
</owl:Class>

<owl:ObjectProperty rdf:ID="имеетОписание">
  <rdfs:domain rdf:resource="#Вино" />
  <rdfs:range rdf:resource="#ОписаниеВина" />
</owl:ObjectProperty>

<owl:ObjectProperty rdf:ID="имеетЦвет">
  <rdfs:subPropertyOf rdf:resource="#имеетОписание" />
  <rdfs:range rdf:resource="#ЦветВина" />
  ...
</owl:ObjectProperty>
```

Рис. 73. Создание свойств в OWL

физического расположения из другого источника и получить местоположение с учетом периода времени, в течение которого оно сохранялось. Для поддержки таких возможностей в OWL имеются специальные возможности.

При совмещении ряда онтологий иногда приходится указывать, что какой-либо класс или свойство в одной онтологии эквивалентны классу или свойству в другой, если онтологии не противоречат друг другу. Свойство `owl:equivalentClass` используется, чтобы сказать, что два класса имеют одинаковых представителей. Пример на рисунке 74 показывает, что класс `foo` эквивалентен классу `bar`.

```
<owl:Class rdf:ID="foo">
  <owl:equivalentClass rdf:resource="#&vin;bar"/>
</owl:Class>
```

Рис. 74. Объявление эквивалентности между классами в OWL

Механизм указания идентичности между экземплярами похож на объявление эквивалентности классов. Элемент `sameAs` используется, когда необходимо объединить индивидов, определяемых в разных документах, с целью унификации двух онтологий. Пример использования этого элемента приведен на рисунке 75.

```
<Вино rdf:ID="КабернеСовиньон">  
  <owl:sameAs rdf:resource="#любимоеВиноВасилия" />  
</Вино >
```

Рис. 75. Объявление идентичности двух экземпляров в OWL

Надо помнить, что в OWL нет допущения об уникальности имени, поэтому отличие двух имен еще не значит, что это разные индивиды. В примере на рисунке 76 демонстрируется, что не всегда необходимо указывать идентичность индивидов через sameAs (подойдет простое перечисление, если не будет противоречия с другой информацией в онтологии).

```
<owl:Thing rdf:about="#BancroftChardonnay">  
  <изготовитель rdf:resource="#Bancroft" />  
  <изготовитель rdf:resource="#Beringer" />  
</owl:Thing>
```

Рис. 76. Объявление идентичности двух экземпляров в OWL (альтернативный вариант)

В OWL поддерживается механизм, противоположный по смыслу элементу sameAs. Использование элемента differentFrom — это один из способов указать, что три значения взаимно отличны. Иногда этот факт следует специально определять. Пример использования элемента представлен на рисунке 77.

```
<СладостьВина rdf:ID="Сухое">  
  <owl:differentFrom rdf:resource="#Полусухое"/>  
  <owl:differentFrom rdf:resource="#Сладкое"/>  
</СладостьВина>
```

Рис. 77. Объявление различности нескольких экземпляров в OWL

6.5. SPARQL

SPARQL — это набор спецификаций, которые предоставляют протоколы для запроса и управления содержимым графа RDF [368]. SPARQL позволяет пользователям запрашивать информацию из баз

данных или любого источника данных, который может быть сопоставлен с RDF. Стандарт SPARQL разработан и одобрен W3C и помогает пользователям и разработчикам сосредоточиться на том, что они хотели бы знать, а не на том, как организована база данных.

SPARQL и SQL

Так же, как SQL позволяет пользователю извлекать и изменять данные в реляционной базе данных, SPARQL предоставляет ту же функциональность для графовых баз данных NoSQL, таких как GraphDB.

Кроме того, запрос SPARQL может выполняться в любой базе данных, которая может рассматриваться как RDF через промежуточное программное обеспечение. Например, запрос к реляционной базе данных может быть построен с помощью SPARQL с использованием программного обеспечения для сопоставления реляционной базы данных с RDF. В отличие от SQL, запросы SPARQL не ограничены работой в одной базе данных: с помощью федеративных запросов можно обращаться с одним запросом сразу к нескольким хранилищам данных. Таким образом SPARQL преодолевает ограничения, связанные с локальным поиском.

SPARQL рассматривает данные в виде ориентированного графа с метками в узлах и на связях, который имеет внутреннее представление в виде триплетов, состоящих из объекта, предиката и объекта. Соответственно, запрос SPARQL состоит из набора шаблонов триплетов, в которых каждый элемент (субъект, предикат и объект) может быть переменной. Значения переменных находятся за счет сопоставления шаблонов в запросе с тройками в наборе данных.

SPARQL поддерживает четыре типа запросов:

- ASK – есть ли хотя бы одно совпадение шаблона запроса в данных RDF;
- SELECT – выбрать все или некоторые совпадения в табличной форме;
- CONSTRUCT – построить RDF-граф, заменяя переменные в совпадениях на значения из набора шаблонов триплетов;
- DESCRIBE – описание совпадений, найденных путем построения соответствующего RDF-графа.

Пример использование SPARQL

Сила SPARQL заключается в поиске отношений в данных RDF-графа путем сопоставления графов, где простые шаблоны могут быть объединены в более сложные, которые исследуют более сложные отношения в данных. Такие отношения могут быть изучены с помощью базовых шаблонов, объединений шаблонов, их пересечений, путем добавления необязательных шаблонов, которые могут расширять информацию о найденных решениях, и т. д. Кроме того, пути свойств допускают последовательный состав, параллельную композицию, итерации, инверсии и т. д.

Основной шаблон диаграммы состоит из триплетов, в которых каждый элемент может быть переменной. Например, шаблон «Джон» (субъект) → «имеет сына» (предикат) → X (объект подстановочного знака) будет иметь в качестве решения каждый триплет в графе RDF, который соответствует субъекту, соответствует предикату и имеет любой объект (графическая визуализация этого шаблона представлена на рисунке 78). Итак, если у Джона два сына – Боб и Майкл, триплеты «Джон» → «есть сын» → «Боб» и «Джон» → «есть сын» → «Майкл» будут решениями данного запроса SPARQL.

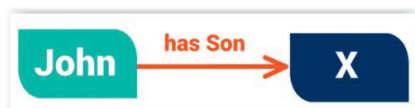


Рис. 78. Графическая визуализация шаблона запроса SPARQL

Запрос SPARQL может также выражать объединение альтернативных шаблонов графов. Любое решение хотя бы одного из шаблонов является решением объединения. Например, у объединения шаблонов «Джон» → «есть сын» → X и «Джон» → «имеет дочь» → X будет иметь в качестве решений всех сыновей и всех дочерей Джона (графическая визуализация этого шаблона представлена на рисунке 79).



Рис. 79. Графическая визуализация объединения шаблонов запросов SPARQL

Групповой шаблон представляет собой соединение двух (или более) базовых шаблонов. В отличие от объединения графов, данная операция требует, чтобы оба (или все) шаблоны были найдены в триплетах одновременно. Итак, соединение (группировка) шаблонов «Джон» → «имеет сына» → Y и «Джон» → «имеет сына» → Z будет иметь в качестве совпадающих решений сыновей сыновей (внуков по мужской линии) Джона (графическая визуализация этого шаблона представлена на рисунке 80).

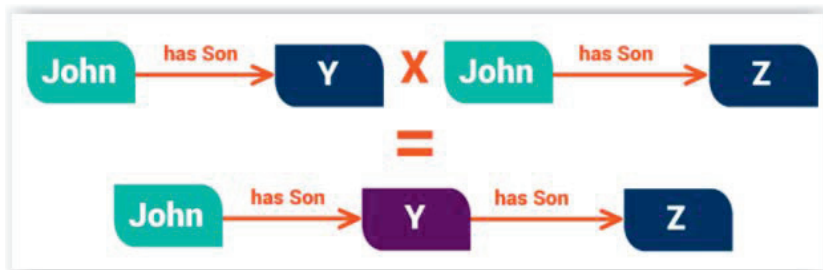


Рис. 80. Графическая визуализация пересечения шаблонов запросов SPARQL

Сыновья дочерей Джона не будут возвращены описанным выше шаблоном, потому что первый базовый шаблон в запросе, а именно «Джон» → «имеет сына» → Y, не будет соответствовать триплетам в данных, таким как «Джон» → «имеет дочь» → «Анна».

Поэтому даже если имеется триплет «Анна» → «имеет сына» → «Тимми», Тимми не появится в качестве решения вышеупомянутого соединения. К счастью, запрос с альтернативными шаблонами и группировка шаблонов могут быть легко объединены. Например, объединение «Джон» → «есть сын» → Y и «Джон» → «есть дочь» → Y, сгруппированное с шаблоном Y → «имеет сына» → Z, найдет всех внуков Джона (графическая визуализация этого шаблона представлена на рисунке 81).

6.6. Примеры онтологий и семантических сетей

Dublin core

Дублинское ядро (Dublin core) – словарь основных понятий английского языка, которые могут использоваться для описания веб-ресур-

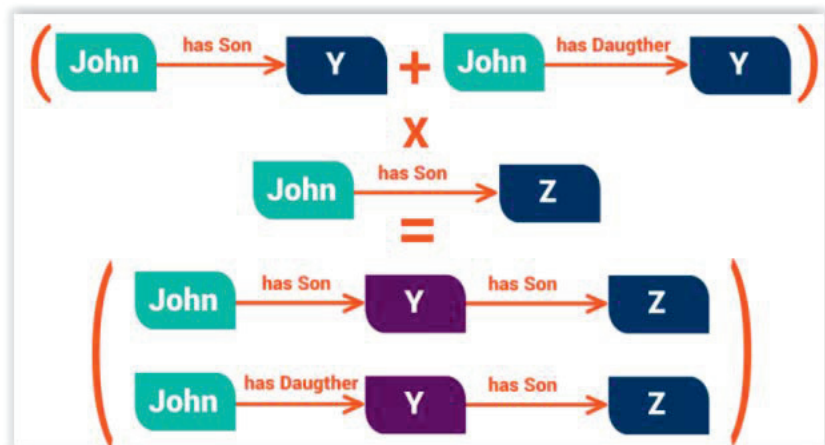


Рис. 81. Графическая визуализация сложного шаблона запроса SPARQL

сов [369]. Этот словарь предназначен для унификации метаданных в рамках описания веб-ресурсов. Семантическая сеть Дублинского ядра представлена в формате RDF [370]. Полный набор терминов метаданных Дублинского ядра предоставлен Инициативой метаданных Дублинского ядра [371]. Основная инициатива метаданных Dublin (DCMI) – предоставление открытого форума для разработки совместимых онлайн-стандартов метаданных для широкого круга задач и бизнес-моделей. Две ключевые цели Dublin Core:

- описание ресурсов;
- объединение словарей метаданных с различными стандартами метаданных для обеспечения совместимости и реализации Семантической паутины.

Набор данных Дублинского ядра состоит из 15 элементов метаданных, известных как набор элементов метаданных Дублинского ядра (DCMES), и включает следующие элементы:

- Title;
- Creator;
- Subject;
- Description;
- Publisher;

- Contributor;
- Date;
- Type;
- Format;
- Identifier;
- Source;
- Language;
- Relation;
- Coverage;
- Rights.

Каждый элемент необязателен и может повторяться. DCMI установил стандартные способы уточнения элементов. В Дублинском ядре нет предписанного порядка для представления или использования элементов. Дублинское ядро активно используется в бизнес-приложениях. В частности, оно применяется в следующих инструментах [372, 373, 374, 375]:

- Rarian – системе каталогизации документов, используемой компьютерами GNOME;
- Браузерах справки KDE;
- Серверах документации ScrollServer;
- PBCore – словаре метаданных публичного вещания;
- Формате электронных книг EPUB;
- Платформе eXo.

Friend of a Friend (FOAF)

FOAF (Friend of a Friend) – это машиночитаемая онтология, описывающая людей, их деятельность и их отношения к другим людям и объектам. Любой может использовать FOAF для описания себя. FOAF позволяет группам людей описывать социальные сети без необходимости создания централизованной базы данных.

FOAF – это описательный словарь, выраженный с использованием RDF и OWL [376, 377]. Компьютеры могут использовать профили FOAF, чтобы найти, например, всех людей, живущих в Европе, или перечислить всех людей, которых вы знаете. Эта функциональность реализуется за счет определения отношений между людьми. Каждый профиль в FOAF имеет уникальный идентификатор (напр., адрес электронной почты человека, международный номер телефона,

имя учетной записи Facebook, идентификатор Jabber или URI персональной веб-страницы человека), который используется при определении этих отношений.

Онтология FOAF активно применяется в таких ресурсах, как:

- LiveJournal;
- DeadJournal;
- MyOpera.

Поддержка FOAF также присутствует в службах Identi.ca, FriendFeed, WordPress и TypePad [378]. Поисковая платформа блога Яндекса поддерживает поиск информации по профилю FOAF [379]. Существуют также модули или плагины для поддержки профилей FOAF или авторизации FOAF+SSL для различных языков программирования, а также для различных систем управления контентом [380].

DBPedia

DBPedia — это краудсорсинговый проект, направленный на извлечение структурированной информации из данных, созданных в рамках проекта Википедия, и публикации ее в виде наборов данных, доступных под свободной лицензией. Проект был отмечен Тимом Бернерсом-Ли как один из наиболее известных примеров реализации концепции связанных данных [381].

По состоянию на сентябрь 2014 г. базы данных DBpedia описывают более 4,58 млн понятий, из которых 4,22 млн классифицированы в соответствии с онтологией, в том числе 1,4 млн персоналий, 735 тыс. географических объектов, 123 тыс. музыкальных альбомов, 87 тыс. фильмов, 19 тыс. видеоигр, 241 тыс. организаций, 251 тыс. таксонов и 6 тыс. заболеваний. DBpedia содержит 38 млн меток и аннотаций на 125 языках; 25,2 млн ссылок на изображения и 29,8 млн ссылок на внешние веб-страницы; 50 млн внешних ссылок на другие базы данных RDF-формата; 80,9 млн связей с категориями Википедии [382].

Проект использует RDF для представления извлеченной информации. На сентябрь 2014 г. база состояла из более чем 3 млрд RDF-триплетов, из которых 580 млн были взяты из английского раздела Википедии и 2,46 млрд извлечены из разделов на других языках. Архитектура предоставления данных DBpedia представлена на рисунке 82.

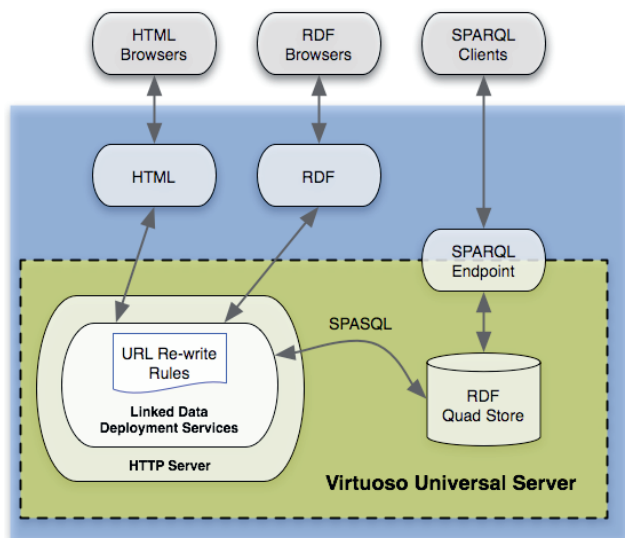


Рис. 82. Текущая архитектура предоставления данных DBpedia

Ниже приведено несколько примеров использования DBpedia:

- AlchemyAPI – облачная текстовая интеллектуальная платформа, которая обеспечивает семантическую маркировку посредством обработки естественного языка, включая извлечение именованных сущностей, анализ эмоциональной окраски, тегирование, извлечение информации об авторских правах, извлечение связей, очистку веб-страниц, определение языка, извлечение ключевых слов, извлечение цитат и тематическую категоризацию. С 2015 г. эта система является компонентом IBM Watson [383, 384].
- Yahoo! Content Analysis API – система, обнаруживающая сущности, понятия, категории и отношения в неструктурированном контенте. Данная система оценивает обнаруженные сущности по их общей значимости, если возможно, связывает их со страницами Википедии и аннотирует теги соответствующими метаданными [385].
- Query – фреймворк на языке Python для преобразования вопросов на естественном языке к формализованным запросам к базе данных. Фреймворк может быть легко настроен на различные типы вопросов и запросов [386].

6.7. Применение семантических сетей

На данный момент основными направлениями использования семантических сетей являются поисковые сервисы и обработка текстов на естественном языке. К первой подгруппе можно отнести алгоритмы Google и алгоритм «Королёв» от Яндекса [387], которые частично построены на семантических технологиях. Ко вторым относятся системы, разрабатываемые компаниями Lokalize [388] и АBBYY. Все эти примеры объединяет одно свойство: основное ядро их алгоритмов строится на нейронных сетях, которые расширяются за счет семантических сетей. Причиной такого ограниченного распространения семантических технологий является высокая сложность использования этих технологий и необходимость сопряжения семантических технологий с другими новаторскими и сложными технологиями, такими как технологии искусственных нейронных сетей.

Однако в 2016 г. начал использоваться такой термин, как «Семантический банк», демонстрирующий проникновение семантических технологий в банковскую сферу [389]. Данная прикладная семантическая технология берет свое начало от FIBO (Financial Industry Business Ontology). FIBO – это открытый стандарт «общего языка» для определения условий, фактов и отношений, связанных с финансовыми контрактами [390]. FIBO охватывает финансовые инструменты (справочные данные по продуктам), рыночные цены и финансовые процессы. FIBO был создан и предложен советом EDM – некоммерческой торговой ассоциацией, основанной финансовой отраслью для повышения эффективности управления данными в качестве бизнес- и операционного приоритета. Совет является ведущим сторонником разработки и внедрения стандартов данных и публикации лучших практик по управлению данными.

Пример Deutsche Bank должен помочь прояснить суть концепции. Банк использовал FIBO в качестве основного языка правил и классификаций для стандартизации контента, и семантики данных внутри банка. Использование FIBO позволило банку сокращать и в конечном итоге устранять необходимость согласования транзакций за счет жестких, прозрачных, неизменных правил [391]. Кроме того, банк устранил необходимость в картографировании, переводе и копировании данных, необходимых для их эффективного использования. Эти результаты позволили быстро выводить на рынок новые продукты и услуги, устраняя необходимость в создании и развертывании дополнительной ИТ-инфраструктуры [392].

Первым шагом банка по внедрению FIBO было его использование для реализации своих целей в сегменте «закупаемых данных», на приобретение которых банк ежегодно расходует десятки миллионов евро. Банку было необходимо перевести данные, которые он покупает у внешних поставщиков, на общий «язык», преобразовать их в формат хранилища RDF и дать возможность бизнес-пользователям осуществлять поиск по всему массиву данных независимо от их происхождения.

Раньше пользователи должны были использовать специальный портал для поиска по массивам покупаемых данных. Массивы данных на этом портале были каталогизированы по источникам данных, использованным для их заполнения. При этом пользователям было необходимо выбрать, по данным из какого источника они хотят осуществлять поиск, т. е. пользователи перед началом поиска должны были знать, в данных из какого источника они смогут найти необходимый ответ.

Поскольку FIBO является онтологией, он дал банку возможность перейти от стиля ETL (Extract–Transform–Load) к связанным данным (linked data), семантическим триплетам и RDF. По результатам проекта банк отметил, что создание подобной системы может дать следующие преимущества:

- практическое представление общего языка для элементов данных;
- модель данных (включая сущности, атрибуты и отношения);
- бизнес-правила;
- таксономия (классификации и иерархии);
- списки кодов и их перечисления (например, валюты и коды стран).

В целом ценность бизнеса, которую FIBO предлагает как банкам, так и регулирующим органам, включает в себя стандартизованный информационный каркас, позволяющий отрасли ориентироваться на общие концепции финансовой предметной области, выступая в качестве выразительной и доказуемой модели для обеспечения соответствия нормативным требованиям, поддержки финансовой и системной информации о рисках, а также для проверки, увязки и согласования данных. Кроме того, использование данного формата позволяет эффективнее использовать данные, оценивая и анализируя гипотезы и выявляя новые возможности.

Глава 7

ТЕХНОЛОГИИ ВИКИ-ГРАФА И ИХ ПРИМЕНЕНИЕ

7.1. Вики-граф

Понятие веб-графа

В данном разделе мы будем активно использовать понятие Вики-графа, однако прежде чем вводить данное понятие рассмотрим более общий случай. Дело в том, что всю сеть Интернет можно представить в виде так называемого веб-графа [393]. Вершинами веб-графа являются веб-страницы, и вершины соединяются ребрами лишь в том случае, если на одной странице имеется гипертекстовая ссылка на другую. Из однонаправленности гипертекстовых ссылок естественным образом следует, что веб-граф является ориентированным, т. е. ребра имеют направления. Кроме того, имеет смысл проводить столько ребер между вершинами, сколько есть ссылок между двумя соответствующими сайтами. Таким образом, веб-граф может иметь кратные ребра или петли

Корпуса гипертекстовых документов уже традиционно являются одним из объектов активного изучения. Данные исследования не только имеют теоретическую ценность, но и позволяют находить новые подходы к поиску, индексации, категоризации и организации для больших массивов документов, вплоть до анализа содержимого всей Всемирной паутины. При этом во многих случаях достаточно продуктивным подходом оказывается отказ от глубокого анализа текста каждого гипертекстового документа в пользу анализа только

гиперссылок между документами, т. е. веб-графа. Например, алгоритм PageRank [394], легший в основу поискового алгоритма Google, отличался от предшествовавших ему поисковых алгоритмов конкурентов именно тем, что вместо глубокого анализа текста страниц анализировал именно структуру ссылок между ними.

Технологии, разработанные в рамках исследований веб-графа или, говоря более широко, графа гипертекстовых документов, могут иметь большое практическое значение для крупного бизнеса, оперирующего большим количеством документов. Например, крупный банк, ежедневно обрабатывающий десятки тысяч документов, уже за один год своей работы порождает огромный массив информации, который обычно даже не пытаются рассматривать как единый информационный артефакт. Тем не менее каждый документ в банке (а в качестве документа в банковской деятельности можно привести практически любую транзакцию, от бухгалтерской проводки до заказа банковской карты) может быть представлен как гипертекстовый документ, ссылающийся на другие документы и бизнес-объекты. Таким образом, по сути массив информации, порождаемый банком, может быть представлен как крупный корпус гипертекстовых документов. Создание эффективных поисковых механизмов для подобных корпоративных гипертекстовых графов может существенно повысить эффективность многих бизнес-процессов банка, а для некоторых из них — способствовать их полной автоматизации.

Активным изучением веб-графа исследователи начали заниматься уже в конце XX в. Рассматривались вопросы о том, каким законам подчиняется развитие Интернета, собиралась информация, разрабатывались математические модели. В своих работах [395, 396, 397] А.-Л. Барабаш и Р. Альберт отметили следующие характеристики веб-графа. Например, было показано, что веб-граф является разреженным, т. е. число ребер в нем приблизительно линейно пропорционально количеству вершин. Кроме того, они обнаружили, что, несмотря на довольно малое (в сравнении с полным графом) количество ребер в веб-графе его диаметр (т. е. наиболее длинный кратчайший путь между вершинами) весьма мал. Так, в 1999 г. диаметр веб-графа оценивался в 19 переходов по ссылкам. Более того, Барабаш и Альберт обнаружили, что структура веб-графа такова, что его диаметр будет слабо увеличиваться при увеличении количества вершин. Например, при увеличении количества узлов веб-графа в 10 раз его диаметр увеличится незначительно — с 19 до примерно 21 пере-

хода. Стоит также отметить, что веб-граф является сильно связным, т. е. между любыми двумя его вершинами можно проложить путь в обе стороны. Конечно, данное утверждение является справедливым, если не рассматривать страницы, не связанные с внешним миром.

Специфика Вики-графа

Обозначим описанный выше веб-граф как $G = (V, E)$, где множество V вершин графа G состоит из всех веб-страниц сети Интернет. Выделим из множества V подмножество $V' \subset V$, представляющее все страницы Википедии (или страницы одного из языковых разделов Википедии), которые являются словарными статьями или категориями, а из множества E ребер графа G выделим подмножество $E' \subset E$, в котором конечные и начальные вершины ребер принадлежат множеству V' . Таким образом, графом Википедии, или Вики-графом, мы будем называть граф $G' = (V', E')$, являющийся подграфом веб-графа G .

Заметим, что в Вики-графе ребра могут быть либо однонаправленными (ребра, соединяющее две статьи), либо двунаправленными (ребра, соединяющие статьи и категории, и ребра, соединяющие статьи между собой). О специфике функционирования ссылок на категории мы подробно говорили в разделе, посвященном анализу Википедии.

Среди исследований массивов гипертекстовых документов, в том числе по анализу гипертекстовых графов, исследование Википедии давно считается привлекательным и популярным. Для этого существует несколько причин:

- весь контент Википедии распространяется свободно, поэтому сбор информации не затруднен технически и абсолютно легален и этичен;
- объем Википедии достаточно велик для получения очень большого графа: в крупнейших языковых разделах Википедии по несколько миллионов статей, а количество связей между ними имеет порядок сотен миллионов;
- объем вики-графа не является чрезвычайно большим и, в отличие от веб-графа, не требует сильно ресурсоемких ИТ-решений; например, множество содержательных задач анализа вики-графа можно выполнять на одной мощной современной рабочей станции;

- так как статьи Википедии создаются членами сообщества по достаточно строгим единым правилам, информация на них (в том числе и ссылки) является частично структурированной, что существенно облегчает ее анализ и использование;
- так как Википедия является гигантской универсальной энциклопедией, то содержащаяся в ней информация достаточно полно (для информационной базы такого объема) описывает окружающую реальность.

Конечно, изучение структуры Википедии имеет большую практическую ценность, так как это крупнейший информационный артефакт современной эпохи, обладающий уникальным сочетанием полноты информации о мире и структурированности. Однако нужно понимать, что Википедия является очень специфичным корпусом гипертекстовых документов; в частности, факты, обнаруженные для Вики-графа, в общем случае нельзя переносить на весь веб-граф или, например, на гипертекстовый граф для корпоративного хранилища документов.

Как было отмечено, несмотря на большой размер Вики-графа, сейчас существуют технологии, позволяющие оперировать с этим графом с помощью достаточно бюджетных решений. Далее в этом разделе будет рассмотрена проблематика построения Вики-графа. Сначала будут рассмотрены технологии, которые позволяют оперировать веб-графом, а затем техника построения такого графа на примере веб-графа русскоязычной Википедии, построенного авторами данного исследования. Так как ключевой технологией для решения задач работы с Вики-графом являются графовые базы данных, следующий пункт данного раздела будет посвящен знакомству с технологиями графовых баз данных.

7.2. Графовые базы данных

Основные черты реляционных и графовых баз данных

При попытке практической работы с Вики-графом одним из первых возникает вопрос: в каком виде хранить сам Вики-граф. Для работы с графами существует множество специализированных программных библиотек, таких, как NetworkX [398] или iGraph [399], однако подобные библиотеки обычно не предназначены для работы с гра-

фами очень большого размера (порядка 1×10^6 вершин и 1×10^8 связей) и не поддерживают специальной инфраструктуры для надежного хранения рабочей версии графа и построения запросов к нему по аналогии с тем, как решают системы управления базами данных. Другим вариантом для работы с Вики-графом является традиционная реляционная система управления базами данных с доступом к данным посредством языка SQL. Но за последнее десятилетие для таких решений появилась интересная альтернатива в виде баз данных, использующих подход NoSQL, на базе различных технологий. Среди NoSQL-технологий для работы с Вики-графом наиболее подходящим решением являются так называемые графовые базы данных. Ниже мы рассмотрим и сравним эти альтернативы.

Реляционные базы данных

Реляционная модель баз данных появилась не сразу. Можно сказать, что реляционная модель данных отражала попытку упростить и отладить более старые модели, такие как иерархическая и сетевая модель данных. В 1970 г. Эдгар Кодд разработал четкое математическое определение реляционной базы данных и задал основу для операций над ней. Говоря неформально, можно дать такое определение: «Реляционная база данных — это такая база данных, где все данные, которые видит пользователь, организованы строго в виде таблиц и где все операции базы данных работают с этими таблицами» [400, с. 51].

Позже, в 1985 г., Кодд написал статью, в которой указал 12 правил, которым должны подчиняться реляционные базы данных. Перечислим некоторые из них:

- вся информация должна быть представлена единственным способом — в виде таблиц;
- каждое атомарное значение данных должно быть доступно логически с помощью комбинаций имени таблицы, названия столбца и первичного ключа;
- должны поддерживаться неизвестные (или отсутствующие) значения NULL.
- В то же время система управления такой базой данных должна ориентироваться на требования ACID:
- атомарность: либо все операции транзакции будут выполнены, либо не будет выполнено ни одной;

- согласованность: база данных должна оставаться согласованной, все правила и ограничения на данные должны соблюдаться, чтобы транзакция успешно завершилась;
- изолированность: каждая транзакция должна выполняться сама по себе, не взаимодействуя с другими транзакциями;
- устойчивость: если транзакция завершилась, то все изменения должны быть сохранены (даже в случае сбоя в приложении или устройстве).

Графовые базы данных

Прежде чем переходить к рассмотрению графовых баз данных, введем определение графа свойств (property graph), который описывает базовую структуру данных, лежащую в основе большинства современных графовых баз данных. Граф свойств – это ориентированный мультиграф (т. е. граф, в котором разрешается присутствие нескольких ребер, соединяющих одни и те же вершины) с помеченными вершинами, имеющими собственный идентификатор, помеченными ребрами и, возможно, петлями [401]. Кроме того, каждой вершине и ребру графа свойств могут быть приписаны свойства, являющиеся наборами пар «ключ–значение». Схема графа свойств наглядно отражена на рисунке 83 [402], а пример такого графа – на рисунке 84.

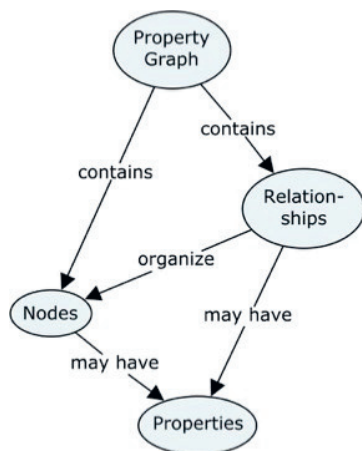


Рис. 83. Модель графа свойств

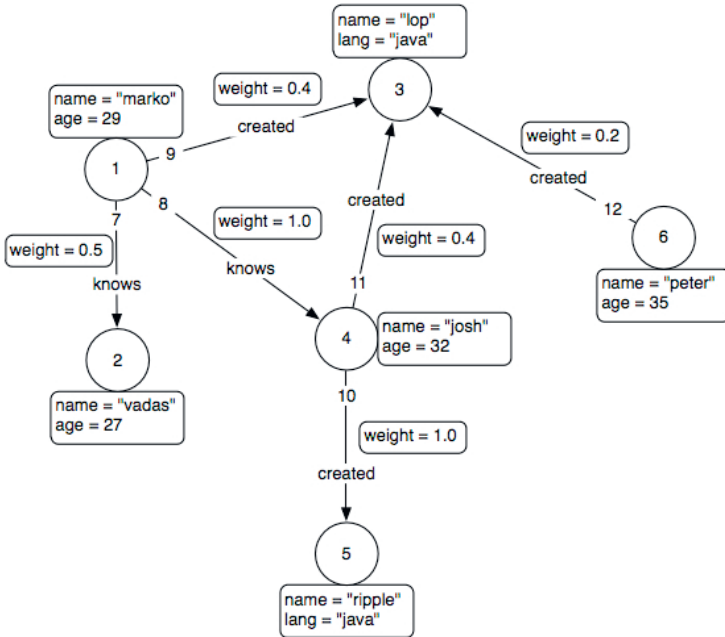


Рис. 84. Пример графа свойств

В отличие от реляционных баз данных, опирающихся на реляционную алгебру, графовые базы данных базируются на фундаменте теории графов и оперируют понятиями узла (синоним вершины графа), связи (синоним ребра графа) и свойства (характерного для графа свойств, описанного выше). В графовой базе данных узел представляет сущность и по своей сути является эквивалентом записи в реляционной базе данных. Ребра отражают связи между узлами, поэтому их также называют связями. Связи являются ключевой особенностью графовых баз данных, предоставляя то, что напрямую не реализовано в других системах. Свойства хранят информацию, связанную с узлами и связями. Таким образом, в графовых базах данных модель представления данных является графом свойств. Работая с такими простыми понятиями, как узлы и связи, графовая база данных позволяет строить модели предметной области любой сложности. При этом вся сложная схема взаимосвязей между данными оказывается представлена в более естественной форме.

Существуют различные подходы к хранению и управлению графами. Некоторые базы данных хранят данные непосредственно в виде графа, в то время как другие каким-либо образом сериализуют его и хранят в таком виде (например, используя для этого реляционные базы данных).

Сравнение реляционных и графовых баз данных

Реляционные базы данных были «рабочей лошадкой» для разработки приложений с 1980-х гг. и остаются таковой до сих пор. Однако в некоторых ситуациях их использование может оказаться неэффективным.

В реляционных базах данных ссылки на другие таблицы отражаются ссылками на их первичный ключ с помощью внешнего ключа. Таким образом, каждый раз, когда требуется воспользоваться связью между двумя сущностями, выполняется операция соединения (JOIN), которая означает поиск ключа в другой таблице. Даже если ключи проиндексированы (что, впрочем, замедляет работу других операций, таких как вставка или обновление данных), то в лучшем случае речь идет о логарифмической (в зависимости от количества записей в таблицах) сложности выполнения этой операции. При большом объеме данных выполнение перехода по связи с такими затратами может оказаться весьма ресурсоемкой операцией.

В графовых базах данных обычно структура хранения данных устроена так, что переход по связям между узлами имеет фиксированную вычислительную сложность и не зависит от количества узлов в графе. Таким образом, если переход по связям является ключевой операцией для хранимых данных, например, если они имеют графовую природу и если объем хранимых данных значителен, то в таких ситуациях графовые базы данных покажут явно лучшую производительность из-за того, что используемые ими структуры данных лучше соответствуют решаемой проблеме.

Нужно отметить, что кроме преимуществ в производительности при работе с графовыми данными графовые базы имеют существенно более подходящий инструментарий для решения задач, которые обычно рассматриваются при работе с графами. Это проявляется в самых разных аспектах, от специфики языка запросов до особенностей пользовательского интерфейса.

Кроме того, необходимо отметить, что графовые базы данных в целом существенно более гибкие в смысле структуры организа-

ции данных. Это позволяет разрабатывать структуру данных параллельно с погружением в проблемную область, а не задавать ее заранее, когда о предметной области еще известно мало. Графовые базы данных позволяют добавлять новые узлы, свойства, подграфы в имеющуюся структуру, не меняя уже существующую функциональность приложения. Конечно, во многих случаях большая гибкость структуры данных несет в себе много рисков, так как в случае графовых баз данных пользователь не может быть полностью уверен в структуре запрашиваемых данных до момента получения результата запроса.

Neo4J

В настоящее время на рынке представлены десятки различных графовых баз данных, из которых самыми известными являются базы данных: Neo4J, OrientDB, Titan, ArangoDB. У каждой из них есть свои преимущества и недостатки. В данной работе мы ограничимся знакомством с базой данных Neo4J, как с имеющей самую долгую историю и, судя по многим оценкам, наиболее популярной базой данных рассматриваемого типа. Перечислим некоторые особенности Neo4J:

- поддерживает навигацию от одного узла к другому за фиксированное время вне зависимости от размера графа;
- рассчитана на работу с запросами к особо важной информации в режиме реального времени;
- гарантирует безопасность данных с помощью ACID-транзакций;
- разработана для работы с высоконагруженными приложениями и большими данным (десятки миллионов узлов);
- данные хранятся в собственном формате, специализированном для представления информации в виде графа, вместо моделирования графа посредством реляционной СУБД.

Языки запросов к графовым базам данных

В отличие от реляционных баз данных, которые обычно поддерживают (правда с большим количеством оговорок) стандартный язык запросов — SQL, для графовых баз данных существует множество разнообразных языков запросов, перечислим лишь наиболее извест-

ные из них: Cypher, SPARQL, GraphQL, Gremlin. Остановимся подробнее на языке Cypher, который используется в базе данных Neo4J.

Особенность Cypher в том, что его базовые конструкции оформлены так, что они в целом легко воспринимаются не только продвинутыми разработчиками, имеющими опыт работы с SQL, но и людьми, не имеющими опыта написания запросов к базам данных. Это достигается за счет того, что синтаксис языка очень нагляден и хорошо согласуется с тем, как принято изображать графы. Cypher позволяет пользователю запросить данные, подходящие под некий шаблон. Можно сказать, что Cypher фокусируется на выражении того, что требуется получить из графа, а не на том, как это сделать. По сути, запрос имеет вид «найди мне что-то вроде этого», где «это» изображается с помощью ASCII-арта, понятного на интуитивном уровне. Так, пара скобок «()» означает круг, которым обычно изображается вершина графа; символы «-->» образуют стрелку, как и принято обозначать ребро в ориентированном графе.

Например, пусть данные о трех людях представлены в виде графа на рисунке 85 [403].

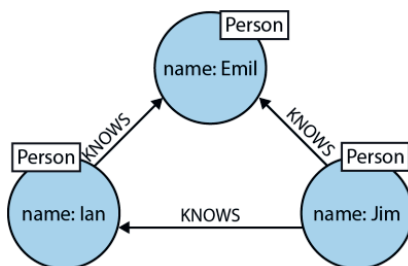


Рис. 85. Пример графа

На языке Cypher с помощью ASCII-арта его можно описать следующим образом (см. рисунок 86).

```
(emil:Person {name:'Emil'})  
  <-[:KNOWS]-(jim:Person {name:'Jim'})  
  -[:KNOWS]->(ian:Person {name:'Ian'})  
  -[:KNOWS]->(emil)
```

Рис. 86. Описание графа на языке Cypher

Аналогичным образом создаются и запросы. Запрос, представленный на рисунке 87, позволяет найти общих друзей пользователя Jim:

```
MATCH (a:Person {name:'Jim'})-[:KNOWS]->(b)-[:KNOWS]->(c),  
      (a)-[:KNOWS]->(c)  
RETURN b, c
```

Рис. 87. Пример запроса на языке Cypher

Разумеется, язык Cypher предоставляет намного более широкие возможности по работе с данными, чем описанные выше, однако их рассмотрение выходит за рамки данной работы.

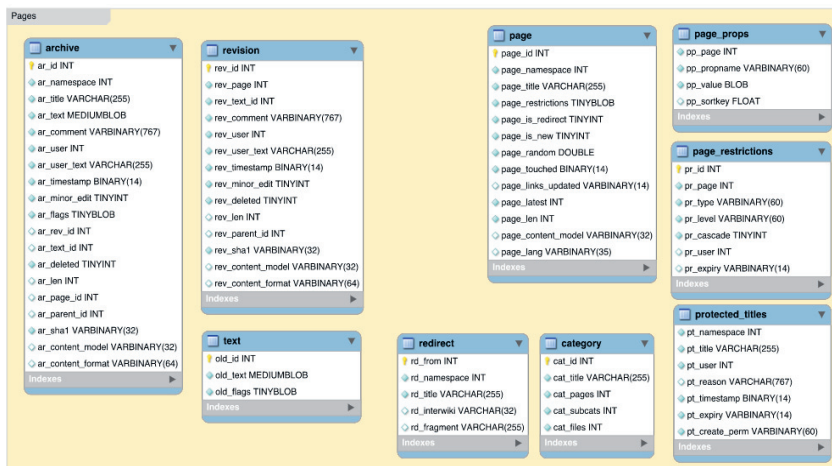
7.3. Построение Вики-графа

В данном разделе описывается практически реализованная авторами технология построения Вики-графа. Данная технология была разработана в рамках построения Вики-графа для русскоязычного сегмента Википедии. Результаты исследования данного Вики-графа приведены в разделе «Анализ Вики-графа».

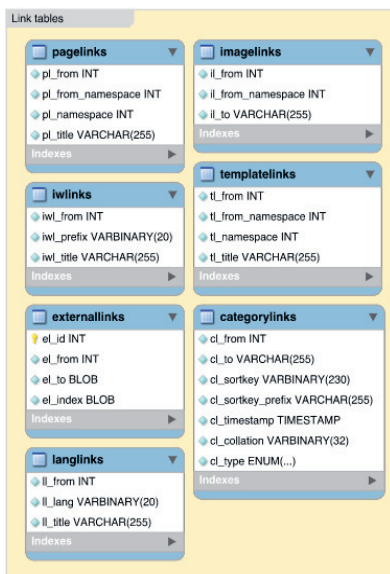
Получение и анализ данных

В рамках проведенного авторами исследования была поставлена цель построить Вики-граф русскоязычного сегмента Википедии и сохранить его в виде графа в графовой базе данных Neo4J.

Одним из преимуществ политики Википедии по предоставлению свободного доступа к данным является то, что для сбора данных о Вики-графе не нужно писать алгоритм-краулер, который будет автоматически обходить все страницы Википедии и собирать на них данные о ссылках. Фонд Википедия выкладывает в свободный доступ исходные данные (так называемые дампы) базы данных, в которой содержатся все данные Википедии, т. е. нет необходимости создавать базу данных с нуля, нужные данные можно получить из дампов базы данных Википедии. Однако, Википедия работает с реляционной базой данных, поэтому все данные в ней представлены в виде таблиц. На рисунке 88 [404] представлены схемы тех частей базы данных Википедии, которые связаны со страницами (а) и ссылками между ними (б).



a)



б)

Рис. 88. Таблицы БД Википедии, содержащие данные о ссылках

Разработка модели графа

Прежде чем переходить непосредственно к работе с Neo4J нужно детально продумать модель будущего графа. В [405] предложены основные принципы, которых стоит придерживаться при создании такой модели. Вот некоторые из них:

- название таблицы отражается меткой на узлах графа;
- каждая строка таблицы — это узел графа;
- столбцы таблицы — это свойства узла графа;
- внешние ключи заменяются связями между узлами;
- данные со значениями по умолчанию можно удалить, так как нет нужды их хранить.

Получение дампов базы данных Википедии

Наконец, последним шагом является получение дампов БД Википедии. Ресурс [406] предоставляет доступ ко всем вики фонда Викимедия, в том числе к русскоязычному разделу Википедии. Так, дампы `ruwiki` содержит все необходимые данные о русской Википедии. При этом, нужно отметить, что объем данных дампа весьма большой. Например, дампы в виде скриптов на языке SQL, заполняющих таблицы реляционной базы данных, отвечающие за хранение информации о страницах и категориях Википедии, имеют суммарный размер более 11 ГБ. Еще раз уточним, что этот объем не содержит текста самих страниц Википедии, изображений и мультимедиа материалов — только общую информацию о страницах и категориях, описанную в схемах базы данных на рисунке 86.

Экспорт данных из реляционной БД в CSV

Наиболее простым способом импортировать большой объем данных из реляционной базы данных в Neo4J является создание CSV-дампа содержимого таблиц реляционной базы данных и импорт этих данных в базу Neo4J при помощи специальной утилиты. Следовательно, в первую очередь скачанные дампы Википедии нужно развернуть, используя реляционную СУБД. Так как проект Википедия использует для работы вики-движка Медиа-Вики базу данных MySQL, то проще всего для разворачивания локальной копии базы данных использовать СУБД MySQL.

В результате выполнения скриптов дампов была получена база данных в СУБД MySQL, хранящая в реляционном виде информацию о страницах и категориях русскоязычной Википедии и связи между ними. На жестком диске такая база занимает порядка 50 ГБ, время импорта данных при создании такой базы на рабочей станции средней производительности составляет порядка 20 часов.

Далее есть возможность воспользоваться тем, что СУБД MySQL предоставляет легкий способ экспортировать результаты SQL-запросов в CSV-файл. Простейший запрос, позволяющий выполнить такую операцию, представлен на рисунке 89.

```
SELECT
  *
FROM
  someTable
INTO OUTFILE 'filename'
FIELDS ENCLOSED BY '"'
TERMINATED BY ','
ESCAPED BY ''
LINES TERMINATED BY '\r\n';
```

Рис. 89. Создание CSV-файла в MySQL

В результате выполнения такого запроса на сервере будет создан файл с именем filename, содержащий результат SQL-запроса в виде CSV-файла. В случае, если нет возможности получить доступ к файлу на сервере, можно использовать MySQL Workbench для экспорта CSV-файла на локальной машине.

Импорт данных из CSV в Neo4J

Основным инструментом для загрузки CSV-файлов в графовую базу данных Neo4J является использование в языке запросов Cypher команды LOAD CSV. На самом деле возможности этого механизма несколько больше, чем просто считывание данных из файла: он позволяет считывать данные по URI-адресу, преобразовывать данные и, что самое важное, создавать структуры и связи.

Для импорта средних по объему CSV-файлов достаточно команды LOAD CSV. Приведем пример. Пусть имеется CSV-файл, содержащий данные о страницах Википедии с заголовками ID, Title и Content. Тогда создать узлы, соответствующие каждой странице, можно с помощью запроса, представленного на рисунке 90.

```
USING PERIODIC COMMIT
LOAD CSV WITH HEADERS FROM "file:pages.csv" AS page
CREATE (:Page {id:page.ID, title: page.Title, content: page.Content})
```

Рис. 90. Пример создания узлов из CSV файла

Пусть также имеется CSV-файл, содержащий данные о связях между страницами Википедии с заголовками From, To, Type. Тогда создать связи между двумя узлами можно с помощью запроса, представленного на рисунке 91.

```
USING PERIODIC COMMIT
LOAD CSV WITH HEADERS FROM "file:links.csv" AS link
MATCH (page1:Page {id: link.From})
MATCH (page2:Page {id: link.To})
MERGE (page1)-[lnk:LINK]->(page2)
ON CREATE SET lnk.linkType = link.Type;
```

Рис. 91. Пример создания связей из CSV-файла

LOAD CSV хорошо подходит для импорта малого или среднего количества данных. Однако для больших объемов данных (когда количество загружаемых объектов превышает сотни тысяч) использовать его не рекомендуется. Для решения задач импорта данных больших объемов в комплекте с Neo4J идет специальная утилита, предназначенная для массового импорта данных из CSV файлов в базу данных: neo4j-import. В качестве параметров для neo4j-import выступают название базы данных, список CSV-файлов, описывающих вершины графа, и список файлов, описывающих ребра графа.

В рамках проведенного авторами исследовательского проекта загрузка Вики-графа русскоязычной Википедии была проведена именно с помощью утилиты neo4j-import. Время импорта базы, содержащей 3,3 млн вершин и порядка 100 млн связей, составило порядка трех часов. Суммарный размер графовой базы данных на жестком диске составил около 5 Гб. Практически десятикратное уменьшение объема базы данных по сравнению с реляционным аналогом объясняется избавлением от некоторых технических столбцов исходных таблиц и использованием при хранении графа более адекватной структуры данных.

7.4. Анализ Вики-графа

Исследование Вики-графа русскоязычного сегмента Википедии

В данном разделе приводятся результаты анализа Вики-графа, построенного авторами по данным русскоязычного сегмента Википедии. В целом русскоязычная Википедия насчитывает более 5,3 млн документов, большая часть которых имеет технический характер, и более 150 млн ссылок между этими документами. Для дальнейшего анализа были отобраны только документы из основного пространства имен – типа страниц Википедии, к которому относятся словарные статьи (далее – статьи Википедии) и категории – специализированные документы Википедии, используемые для тематического структурирования информации в Википедии. Как видно из таблицы 15, категорий и ссылок на них в Википедии примерно на порядок меньше, чем статей и прямых ссылок между статьями.

В среднем на одну статью Википедии (здесь и далее в разделе под Википедией подразумевается ее русскоязычный сегмент) приходится 27,9 ссылок на другие статьи и 2,8 ссылок на категории (с помощью таких ссылок статья относится к категории). В отличие от ссылок между статьями ссылки со статьи на категорию фактически являются двунаправленными: на всех страницах категорий автоматически формируются ссылки на все статьи, ссылающиеся на эту категорию. В среднем к каждой категории относится 22,7 статей. Наряду с обычными статьями категории сами могут быть отнесены к одной или нескольким категориям более высокого уровня, таким образом в Википедии обеспечивается тематическое структурирование не только статей, но и самих категорий. В среднем каждая категория отнесена к 1,9 категорий более высокого уровня. Необходимо отме-

Таблица 15

Интегральные показатели построенной сети Википедии

Тип страницы	Количество страниц	Ссылок	
		на статьи	на категории
Статья	3305221	92167866	9187747
Категория	405363	9187747	770864
Всего	3710584	101355613	9958611

туть, что при загрузке Вики-графа авторами был исключен ряд категорий, имеющих технический характер (около 18 тыс. категорий).

Результат анализа распределения степеней (количества связей) узлов сети, относящихся к статьям Википедии, представлен на рисунке 92. На графике распределения входящих степеней отражена зависимость количества статей Википедии от количества ссылок на данные статьи. Видно, что в двойном логарифмическом масштабе эта зависимость очень близка к прямой на интервале входящих степеней от 0 до 90, далее наклон прямой незначительно меняется, а зависимость зашумляется и демонстрирует свою дискретную природу для высоких степеней узлов. Вид графика показывает, что входящая степень узлов может хорошо описываться степенным законом распределения $P(k) \sim k^{-\gamma}$ (здесь k — степень узла, а $P(k)$ — доля узлов сети, имеющих степень, γ — показатель степенного закона распределения, являющийся параметром распределения).

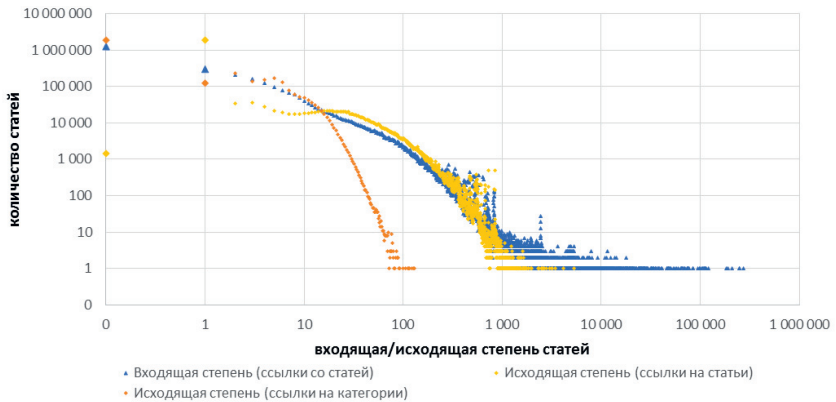


Рис. 92. Распределение степеней (количества связей) статей русскоязычной Википедии в двойном логарифмическом масштабе

Графы со степенным распределением вершин принято называть безмасштабными, их формирование хорошо описывается моделью Барабаша–Альберта. Согласно этой модели, граф возникает в результате пошагового роста графа, при котором действует принцип «предпочтительного присоединения», т. е. новые связи с большей вероятностью образуются у вершин, уже имеющих большое количество связей. Это хорошо согласуется с логикой развития Вики-графа.

В результате роста графа по модели Барабаши–Альберта образуется небольшое количество ключевых вершин графа — «хабов», на распределении степеней узлов это проявляется как «тяжелый хвост» распределения. На рисунке 90 видно, что хвост распределения для входящих степеней даже более тяжелый, чем предполагает степенной закон распределения. Аналогичная картина наблюдается для распределения входящих степеней узлов для категорий (рисунки 93).

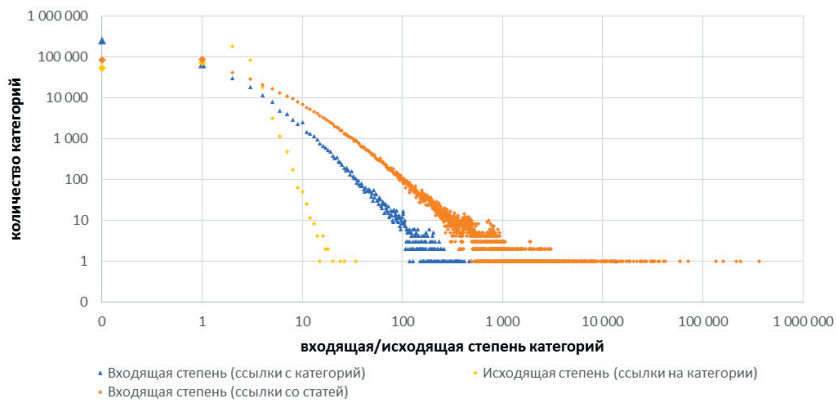


Рис. 93. Распределение степеней (количества связей) категорий русскоязычной Википедии в двойном логарифмическом масштабе

Очень существенно от степенного закона распределения отличается распределение степеней узлов для исходящих степеней статей как в голове и середине распределения, так и в хвосте распределения — он не такой протяженный. Это вызвано существенно иной природой возникновения исходящих ссылок в статьях: они формируются в процессе правки статьи и зависят от ее проработанности и целесообразности добавления новых ссылок, а не от результата большого количества правок на миллионах других статей, поэтому для исходящих ссылок требуется адаптация модели предпочтительного присоединения.

Кроме распределения степеней узлов был рассмотрен вопрос средней длины пути в сети Википедии. Для сетей небольшого размера обычно проводится расчет длины кратчайшего пути между всеми парами узлов сети. Ввиду высокой вычислительной сложности зада-

чи для сети Википедии средняя длина пути была получена в результате расчета длины кратчайшего пути между парами статей из случайной выборки из 50 статей Википедии. Путь строился по прямым ссылкам между статьями с учетом их направления без использования категорий, для каждой пары статей строилось два пути: по одному в каждом направлении. Средняя длина составила 4,4 перехода, распределение длин приведено на рисунке 94. Полученные результаты говорят о большой компактности сети и целесообразности анализа его принадлежности к сетям тесного мира [407].

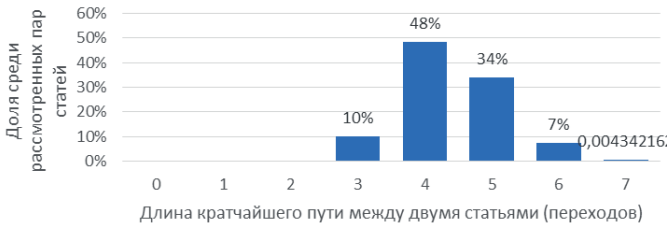


Рис. 94. Гистограмма распределения длин кратчайших путей (измеряемых в количестве переходов) между статьями русскоязычной Википедии, попавшими в случайную выборку

Проведенный в исследовании первичный анализ сети русскоязычного сегмента Википедии показал, что для построения модели формирования входящих ссылок может быть применена модель Барабаши—Альберта, однако для моделирования исходящих ссылок как для статей, так и для категорий потребуются разработка специализированной модели. Сеть Википедии демонстрирует маленькую среднюю длину пути между узлами, после дополнительного анализа может рассматриваться вопрос ее принадлежности к сетям тесного мира.

Результаты исследований Вики-графа другими авторами

Помимо описанного выше авторского исследования Вики-графа в литературе известно множество результатов аналогичных исследований Вики-графа англоязычной Википедии. Так, в [408] показано, что распределение степеней вершин англоязычной Википедии тоже имеет степенной закон со степенным показателем $\gamma \approx 2,2$, что неплот-

хо согласуется со значением этой величины для веб-графа, для которого в исследовании [394] этот параметр был оценен как $\gamma \approx 2,1$.

Перечислим еще несколько примечательных свойств Вики-графа. Во-первых, вершины с большим количеством входящих вершин с большой вероятностью также имеют большое количество исходящих вершин. Во-вторых, страницы Википедии ссылаются не на случайные страницы, а на аналогичные по содержанию, поэтому Вики-граф имеет более кластеризованную структуру, чем произвольный безмасштабный граф.

Назовем коэффициентом кластеризации условную вероятность того, что две вершины имеют общее ребро, если существует другая вершина, имеющая общие ребра с названными вершинами, т. е. $P(\{\exists(i, j)\} | \{\exists k: \exists(k, i) \text{ и } (k, j)\})$. Оказывается, что среднее значение коэффициента кластеризации Вики-графа довольно высоко по сравнению с другими безмасштабными графами (приблизительно 7×10^{-3}). Таким образом, можно утверждать, что статьи, похожие на некоторую конкретную статью, находятся в ее ближайшем окружении. Этот факт эффективно используется на практике, например, при расчете семантической близости концептов, что будет описано ниже.

7.5. Практическое применение Вики-графа

Расчет семантической близости концептов

Банковский бизнес подразумевает обработку большого количества текстовых документов, при этом далеко не всегда эти документы имеют жестко регламентированный вид, что означает, что выполнение операций с этими документами требует либо применения дорогостоящего ручного труда, либо создания информационных систем, способных определить смысл полученных документов и за счет этого либо автоматически обработать документы, либо существенно снизить трудозатраты на их ручную обработку.

Одной из ключевых задач выявления смысла документов является правильное понимание смысла, стоящего за встречающимися в тексте словами или понятиями. Если говорить более формально, то для находящихся в тексте символов (слов или словосочетаний, определяющих некоторое понятие) нужно определить их концепты (т. е. смысл). Кроме того, нужно построить корректные соотношения между различными концептами.

Человек может интуитивно оценить тесноту связи между двумя концептами, поскольку, сравнивая их, он имеет нечто большее, чем просто определение двух понятий. Имея в своем арсенале знания о предметной области, он может выстроить некоторую ассоциативную цепочку понятий, чтобы сделать итоговый вывод о тесноте связи каких-либо двух из них.

Вики-граф является по сути базой знаний о концептах и может быть использован для связывания концептов с помощью алгоритмов, воспроизводя таким образом в машине способность, аналогичную имеющейся у человека. Реализация этой способности в машине позволяет автоматически получать численную меру семантической близости между двумя концептами.

Рассмотрим основные методы расчета семантической близости двух концептов, которые можно разделить на контентные и ссылочные. Контентные меры семантической близости подразумевают использование текстового содержимого статей Википедии. В этом случае тексты статей представляются в виде векторов слов над пространством терминов. Недостаток данного подхода состоит в том, что нужно хранить (либо получать) тексты статей для изучаемых концептов, что является трудоемкой задачей.

Ссылочные меры близости концептов основаны на представлении концептов вершинами в Вики-графе. Многие из них основываются на модели случайного блуждания и так или иначе используют алгоритм PageRank. Одним из примеров является алгоритм WikiWalk [409]. Кратко его можно описать следующим образом: рассматривается указатель на вершину графа, который перемещается по вершинам графа, случайно переходя по ребрам из текущей вершины. С небольшой вероятностью этот указатель может «телепортироваться» в любую другую вершину графа с заданной вероятностью перехода. Процедура «телепортации» позволяет преодолеть тупиковые ситуации, которые могут возникнуть при определенных конфигурациях Вики-графа. В оригинальном PageRank точка назначения «телепортации» задается равномерным распределением, здесь же она задается неравномерным распределением, с учетом «вектора телепортации». Такой алгоритм запускается на каждом из сравниваемых концептов, и затем вычисляется близость полученных результатов.

Меры на основе случайного блуждания обеспечивают высокое качество определения семантической близости концептов, однако это достигается за счет большой временной сложности, кото-

рая, в общем случае, не меньше, чем $O(n)$, где n – количество вершин графа.

Так как вследствие большого размера Вики-графа использование такого алгоритма не является эффективным, стоит рассмотреть другой подход к вычислению меры семантической близости. Выше уже было сказано, что наиболее похожие по смыслу концепты расположены вблизи изучаемого концепта. Поэтому представляется целесообразным использовать в качестве меры семантической близости кратчайшее расстояние между вершинами. Однако, и такой подход оказывается неэффективным с точки зрения вычислений: предварительный расчет и хранение результатов оказывается дорогим в плане памяти, а обход графа при выполнении программы является дорогим в плане времени. В [410] было предложено использовать для решения этой задачи специальные структуры данных – индексы, которые с одной стороны относительно компактны, а с другой – позволяют снизить расходы на вычисление кратчайших путей между вершинами графа.

В целом, исследования показали, что для ссылок на статьи и категории прослеживается тенденция к убыванию семантической близости концептов при увеличении средней длины пути между ними.

Разрешение многозначных фразеологизмов

Еще одна прикладная задача по анализу текстов на естественном языке, которую можно решить с помощью Вики-графа, – это разрешение многозначных фразеологизмов. Википедия содержит большое количество многозначных терминов; так, среднее количество значений многозначного термина англоязычной Википедии равно 17. Возникает задача: выбрать среди возможных значений терминов наиболее подходящее для данного контекста. Это можно сделать, например, следующим образом.

Пусть исследуется многозначный термин t , пусть у него возможные значения t_1, \dots, t_k . Выделим в окружении термина t некоторое количество однозначных терминов c_1, \dots, c_n . Тогда из значений термина t можно выбрать такое, которое максимизирует сумму значений близости к терминам контекста:

$$t_{result} = \max_{t_i} \sum_{j=1}^n sim(t_i, c_j),$$

где $sim(t_i, c_j)$ – некая мера семантической близости, например, описанная в предыдущем разделе.

Проверка данного алгоритма на реальных данных показала, что в 72% случаев многозначный термин определяется правильно. Такой процент является достаточным для дальнейшей работы с термином, например, при решении задач классификации текстов.

Рассмотренные решения не являются чисто теоретическими моделями. На данный момент прототипы подобных решений уже имеют реализацию. Например, в ИСП РАН разработана технология TEXTERRA и программный модуль Visontia, который позволяет получить численное значение семантической близости между несколькими концептами. Кроме того, для каждого многозначного термина он показывает возможные значения, упорядоченные по близости к контексту.

Рубрикация текстов

Рассмотрим еще одну задачу обработки документов на естественном языке, актуальную для практической деятельности банка или крупной компании — задачу автоматической рубрикации текстов. Например, автоматическая рубрикация может использоваться при автоматической фильтрации входящих документов для направления их исполнителям, работающим с конкретными случаями. Наличие подобных фильтров может позволить углубить разделение труда, снизить трудозатраты на обработку одного документа и повысить скорость реакции на входящие документы.

Обычно задача автоматической рубрикации решается методами машинного обучения. На вход классифицирующему алгоритму предоставляется обучающая выборка с заранее классифицированными текстами, на которой алгоритм проходит обучение. Однако подготовка хорошей обучающей выборки может быть весьма трудозатратной операцией. Более того, при изменении специфики входящих документов может потребоваться новый раунд обучения и соответствующая повторная подготовка обучающей выборки.

Сложностей с формированием и поддержкой обучающей выборки можно избежать, если использовать рубрикатор текста на основе семантической близости, который не требует привлечения обучающей выборки. Это возможно потому, что для этой задачи можно использовать алгоритм, работающий по точно такой же логике, как и алгоритм разрешения многозначности в тексте.

Технологии Вики-графа и их применение

Пусть имеется Вики-граф, и пусть заданы рубрики c_1, \dots, c_n , которые являются статьями Википедии. Пусть также текстовый документ содержит некоторые термины t_1, \dots, t_k . Тогда текстовый документ классифицируется в рубрику, которая максимизирует сумму близости терминов к рубрике:

$$c = \max_{c_i} \sum_{j=1}^n \text{sim}(c_i, t_j)$$

Результаты тестирования данного алгоритма показали, что в 80% случаев результат классификации новостной статьи с Google News совпадает с рубрикой, в которой и была размещена статья.

Как и в предыдущих задачах, решение этой задачи было реализовано в рамках технологии TEXTERRA. Она позволяет проводить классификацию текстов, где каждая рубрика задается названием статьи Википедии.

Глава 8

ТЕХНОЛОГИИ КОНЦЕПТУАЛЬНЫХ КАРТ И ИХ ПРИМЕНЕНИЕ

8.1. Суть и специфика концептуальных карт

Концептуальные карты и технология их построения

В последние годы все больше литературы и научных исследований так или иначе касаются тем, связанных с концептуальными картами и возможными областями их применения. Однако, несмотря на то, что концептуальные карты являются мощным инструментом для сферы управления знаниями, обучения и построения онтологий, активность практического использования концептуальных карт в практических приложениях все еще достаточно низка.

Концептуальные карты, или концептуальные диаграммы, представляют собой графический инструмент организации и визуализации знаний, диаграмму, визуально описывающую взаимосвязи между концептами (идеями, понятиями). Любая концептуальная карта в своей основе опирается на понятия концепта и связующего слова. Джозеф Д. Новак, один из авторов теории концептуальных карт, определяет понятие концепта как «воспринимаемые закономерности или шаблоны в событиях или объектах, или записи событий или объектов, обозначенных меткой».

Связь между концептами в свою очередь описывается с помощью множества связующих слов. В качестве таких слов как правило используются краткие глаголы, описывающие непосредственно саму логику взаимосвязи. Совокупность двух и более понятий и связую-

щих слов между ними могут образовывать семантическую единицу – предложение. Возможность интерпретации группы концептов и отношений между ними в виде осмысленных предложений является одной из ключевых особенностей концептуальных карт, позволяющих конструировать новые знания в разрезе определенной предметной области.

Сам процесс построения стандартной концептуальной карты может быть представлен в виде следующего алгоритма:

- Шаг 1: выделение фокус-вопроса (вопрос или тезис определяющий суть проблемы/задачи для решения которой требуется построение соответствующей концептуальной карты).
- Шаг 2: составление списка ключевых понятий, относящихся к выделенному фокус-вопросу.
- Шаг 3: построение концептуальной карты с использованием ключевых понятий и связей между ними.
- Шаг 4: дополнение карты понятиями, не вошедшими в первоначальный список и связывание этих понятий с понятиями, уже включенными в карту.
- Шаг 5: оценка полноты карты, при необходимости, возврат к шагу 5.

Таким образом, можно заметить, что концептуальная карта подразумевает ее построение в итеративном ключе. По различным оценкам для формирования хорошей концептуальной карты необходимо совершить около трех последовательных итераций.

Диаграмма на рисунке 95 представляет собой пример карты, описывающей само понятие концептуальных карт. Возможность создания такого самоописания и легкость его чтения показывает, сколь гибкими и естественными для восприятия являются концептуальные карты в задачах формирования системы понятий, связанных определенными отношениями.

Сравнение концептуальных карт с другими формами визуализации знаний

Для лучшего понимания специфики концептуальных карт имеет смысл сравнить ее со схожими формами визуализациями. В качестве аналогов для сравнения мы рассмотрим интеллект-карты (mind map) и диаграммы классов на языке UML (unified modelling language).

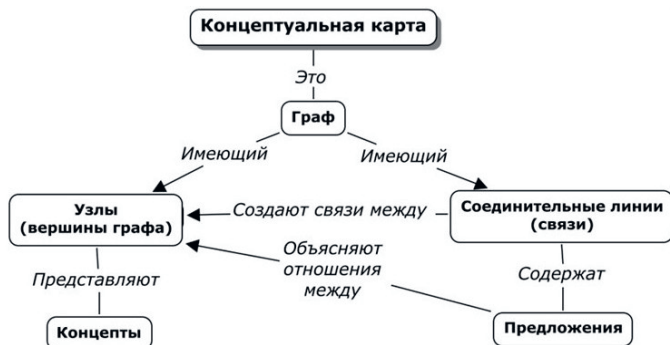


Рис. 95. Концептуальная карта, описывающая понятие концептуальных карт

Интеллект-карты в своей основе подразумевают древовидную структуру и фокусирование на определенной теме, в то время как концептуальные карты реализуют универсальную структуру, что позволяет концептуальным картам делать акцент на определенной абстрактной системе или группе концептов, а не единичной теме, как это обычно бывает у интеллект-карт. За счет использования связующих слов, отношения между концептами в концептуальной карте типизированы и имеют определенные свойства, а связи интеллект-карты, напротив, не типизированы и указывают лишь на подчиненность родительскому узлу. Возможность типизации связей повышает гибкость и емкость диаграмм концептуальных карт, но при этом требует проведения более сложного и трудоемкого анализа.

На рисунке 96 представлен типовой пример интеллект-карты. Аналогично концептуальной карте, рассмотренной выше, дан-



Рис. 96. Пример интеллект-карты

ная карта также представляет из себя некое подобие самоописания. В данном примере сразу прослеживаются ключевые отличия интеллект-карт от концептуальных, включая древовидную структуру, построенную вокруг единой темы, а также набор нетипизированных связей.

Проводя сравнение специфики концептуальной карты с диаграммой классов UML стоит отметить, что сам факт использования языка UML априори предполагает специфичную цель, для которой они строятся (хотя встречаются примеры, когда UML-диаграммы создаются не для целей анализа и проектирования при разработке информационных систем) и наличие определенных стандартов построения диаграмм. Подобная «стандартизация» UML предполагает существование ограниченного набора типовых элементов, связей и диаграмм. Это существенно отличает UML от концептуальных карт, чья универсальная структура исключает наличие неких четких правил составления. За счет такой универсальной структуры практически для любой диаграммы, составленной с использованием языка UML, может быть построена эквивалентная концептуальная карта. Однако обратное не всегда верно. Преобразование некоторых концептуальных карт в UML-диаграммы возможно, но это зачастую сопровождается потерей определенной смысловой информации.

В качестве наглядного примера на рисунке 97 изображена концептуальная карта и диаграмма классов UML, использованная как основа для построения соответствующей карты.

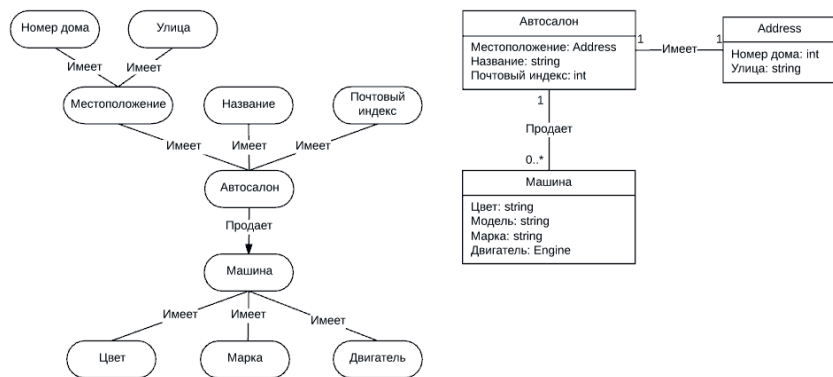


Рис. 97. Концептуальная карта (слева), построенная на основе диаграммы классов UML (справа)

Теоретическая основа концептуальных карт

Теоретическая составляющая концептуальных карт базируется на теории обучения Дж. Д. Новака, которая в свою очередь расширяет и дополняет основные постулаты когнитивной психологии Д. Осубела, рассмотренные в контексте школьного обучения.

Основная идея в психологии Дэвида Осубела, так называемой ассимиляционной теории, заключается в утверждении, что процесс обучения происходит за счет ассимиляции новых концептов и препозиций в существующие концепты и пропозициональные структуры, которыми обладает обучающийся [411]. Подобная структура знаний, которой обладает обучающийся, также называется когнитивной структурой сознания человека.

Таким образом, интеграция новых концептов в когнитивную структуру человека в процессе осмысленного обучения происходит за счет сопоставления новых знаний с уже имеющимися концептами. Концептуальная карта в данном случае является визуализацией подобных связей между концептами в когнитивной структуре нашего сознания. Это и объясняет эффективность применения концептуальных карт для обучения и управления знаниями: по сути построение подобных карт представляет из себя процесс конструирования концептов на основе уже существующих.

8.2. Применение концептуальных карт

Концептуальные карты как инструмент управления знаниями

Управление знаниями является на данный момент одним из ключевых направлений менеджмента ресурсов организации. Вопросы эффективного управления знаниями особенно остро стоят для крупных организаций, ориентированных на нематериальное производство, таких как банки. Для этих организаций знания являются одним из основных активов и при этом их объем слишком велик, чтобы эффективно их использовать (собирать, хранить, распространять) без использования специализированных технологий. Концептуальные карты являются эффективной технологией для решения задач управления знаниями, что делает их применение в банках очень перспективным.

Под управлением знаниями в организации понимаются процессы по контролю и обеспечению перевода неявных (скрытых) знаний в яв-

ные интеллектуальные ресурсы, которые могут быть в дальнейшем использованы для совершенствования деятельности организации.

Применение концептуальной карты в области управления знаниями непосредственно связано с таким понятием как «спираль знаний», предложенным Икуджиро Нонака в 1990 г. [412]. Спираль знаний, как можно видеть на рисунке 98, моделирует последовательное преобразование знаний организации в результате цикла из четырех процессов: социализации, экстернализации, комбинации и интернализации. Далее мы более подробно рассмотрим каждый из этапов этого цикла.

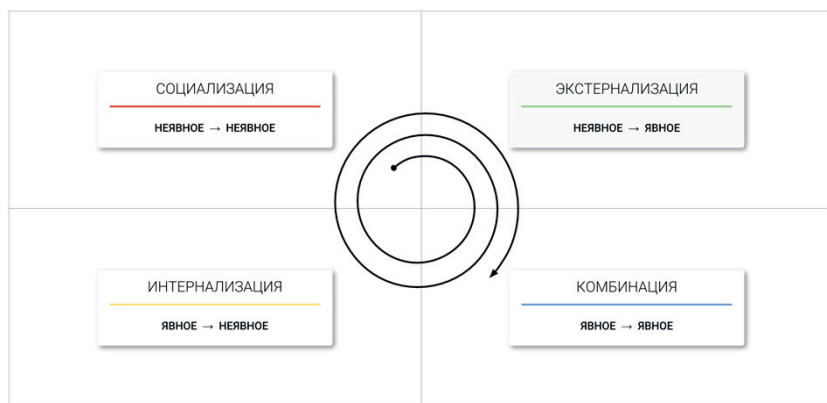


Рис. 98. Спираль знаний

Социализация и применение концептуальных карт

В результате социализации неявные (скрытые) знания одного человека преобразуются в неявные знания другого. К примеру, если человек наблюдает чью-либо деятельность в организации, он так или иначе формирует у себя представление об этой деятельности и о способах ее выполнения. Так у него формируется собственное неявное знание.

Процесс социализации характерен своей неформальностью: знания не переводятся в явную форму, не документируются и не контролируются. Поэтому в этом процессе организация фактически не приобретает знания в виде собственного интеллектуального актива. Несмотря на то, что социализация самый неэффективный

процесс с точки зрения приобретения знаний, он при этом самый распространенный: из-за неправильной организации управления знаниями, они передаются в основном скрыто и невербально.

Социализация знаний происходит за счет работы сотрудников в одной среде, с общими задачами и нормами. Концептуальная карта, организуя единую среду работы сотрудников организации и обеспечивая единый подход к выполнению задач, позволяет организовать контроль за созданием знаний, сотрудники отражают свои знания на карте, тем самым формализуя, проявляя их. Таким образом доля процесса социализации уменьшается и основными процессами становятся экстернализация, комбинация и интернализация, которые обеспечивают формирование интеллектуальных активов и способствуют развитию организации.

Экстернализация и применение концептуальных карт

Процесс экстернализации заключается в преобразовании неявных знаний в явные при помощи языка, метафор и аналогий. Другими словами, экстернализация — это кодификация человеческого опыта. Чаще всего этот процесс происходит в результате объединения различных неявных знаний в определенный информационный объект (картинку, текст, речь), что позволяет осуществить их явную передачу другим людям. К примеру, сотрудник, обладающий неявными знаниями об эффективном осуществлении какой-либо рабочей операции, может поделиться этими знаниями на собрании или в отчете, тем самым сделав свои знания явными.

Формирование нового концепта и связывание его с другими концептами на карте при помощи отношений является частным случаем экстернализации. Неявное знание в организации может со временем отслеживаться и переноситься сотрудниками организации на концептуальную карту, где за счет связи с другими, уже существующими знаниями, оно становится явным. Другими словами, создавая новый концепт на карте, мы поясняем его через другие концепты и делаем его понятным, пригодным для понимания другими людьми.

«Явность» знания обеспечивается за счет возможностей организации элементов концептуальной карты: иерархического представления, различных отношений и видов связей. Таким образом, за счет уже существующей структуры знаний, отображенных на концепту-

альной карте, неявные знания преобразуются в формат, понятный и готовый к распространению в организации, и становятся интеллектуальным ресурсом организации.

Комбинация и применение концептуальных карт

В процессе комбинации явные знания преобразуются в явные. Это происходит за счет совмещения, комбинирования различных типов явных знаний. К примеру, создание учебника по биологии, являющегося явным знанием, требует совмещения различных явных знаний о животных, растениях и прочих объектах изучения биологии, которые содержатся в виде научных трудов, энциклопедий, т. е. также являются явными знаниями.

Элементы концептуальной карты могут формироваться как в процессе экстернализации, так и в процессе комбинации. Явные знания, содержащиеся в документах, книгах и других носителях, могут быть преобразованы, соответственно используемой методологии (нотации), в элементы концептуальной карты. Кроме того, каждое новое явное знание в виде элемента карты добавляется к уже существующим знаниям, тем самым формируя общую систему знаний, которая также является явным знанием об организации в целом. Это еще одно проявление процесса комбинации.

Интернализация и применение концептуальных карт

Интернализация противоположна экстернализации. В ходе этого процесса явные знания преобразуются в неявные. Частный случай такого процесса – обучение. Интернализация происходит в результате усвоения человеком (сотрудником) некоторых знаний и применения этих знаний на практике. Несмотря на то, что неявные знания формируются у сотрудника, они все равно являются ресурсом компании, так как сотрудник может их применять для достижения целей компании.

Концептуальная карта может являться единым и общедоступным источником знаний организации, с помощью которого происходит обучение новых сотрудников, формирование целостного представления о деятельности и методологии организации. Также при помощи концептуальных карт происходит передача экспертных знаний и их сохранение в организации.

Поддержка коллективной работы с помощью концептуальных карт

Система управления знаниями, включающая концептуальную карту, помимо организации процессов преобразования знаний в ресурсы организации позволяет также осуществлять поддержку коллективной работы в организации. В первую очередь это касается создания и поддержки единых стандартов, методологии и корпоративной культуры в организации за счет того, что весь коллектив взаимодействует с единой концептуальной картой. По этой же причине в коллективе формируется единое представление и понимание знаний и стратегий организации. Наконец, такая система исключает наличие процессов повторного формирования знаний, так как все сотрудники имеют доступ ко всем существующим знаниям в организации в любое время.

Концептуальные карты как инструмент обучения

Одним из основных способов применения концептуальной карты является обучение. Помимо повышения степени усвоения материала за счет графического представления информации и использования связей и отношений, концептуальная карта также может быть использована для формирования навыков критического мышления и метакогниции (умения учиться). На примере концептуальной карты человек формирует у себя умения структурирования информации и выделения ключевых фактов.

Концептуальная карта может использоваться не только как способ передачи знаний обучаемой аудитории, но и как средство познания. Как показывает опыт Тайваньского национального института [413], студенты, обучавшиеся при помощи коллективного создания концептуальных карт на основе проходимого материала, усваивали этот материал лучше, чем студенты, обучавшиеся в традиционной форме. При такой организации обучения студент учится выделять главную информацию из общей совокупности информации, включать эти мысли в общую структуру и связывать с уже выделенной информацией.

Также концептуальная карта позволяет лучше усваивать материал и не забывать уже пройденный за счет естественного встраивания новой информации в уже существующую структуру знаний.

Концептуальные карты как инструмент проектирования

Являясь диаграммой с возможностью различного представления концептов, реализации иерархических и прочих отношений между ними, концептуальная карта может использоваться для целей проектирования. В отличие от других популярных средств проектирования, как, например, UML, концептуальная карта не имеет жесткой методологии, а ее нотация намного более гибкая, чем у большинства видов диаграмм. Это является как преимуществом, так и недостатком.

С одной стороны, концептуальная карта гибче в отношении реализации взаимосвязи компонентов и их представления, однако в то же время концептуальная карта не может использоваться как унифицированный стандарт проектирования. Отсутствие унификации приводит к тому, что каждая проектная команда по сути должна явно или неявно сформировать свою методологию проектирования в соответствии со своей задачей, предметной областью и личными предпочтениями. Формирование такой методологии требует волнительных трудозатрат и затрудняет обмен результатами проектирования между разными командами. Тем не менее, концептуальная карта — это «чистый лист», и на его основе можно создавать узкоспециализированные стандарты для частных целей проектирования.

В рамках банковской деятельности концептуальные карты можно использовать либо для ситуативных задач проектирования, либо для задач проектирования при решении задач, для которых нет общепринятых нотаций диаграмм. При решении этих задач гибкость, которую дают концептуальные карты, будет перевешивать минусы их применения вместо стандартных средств. Кроме того, по мере развития практики использования концептуальных карт в рамках деятельности банка можно сформировать внутренние стандарты на использование концептуальных карт для типовых задач, в которых практика их применения оказалась эффективной. Формирование таких внутренних стандартов будет компромиссом между бесстандартными схемами использования концептуальных карт и использованием стандартных диаграмм других типов.

**Концептуальные карты как инструмент
конструирования онтологий**

В разделе «Технологии семантических сетей и их применение» мы подробно рассматривали одну из ключевых семантических технологий — онтологии. Кроме возможности использования концептуальных карт как инструмента управления знаниями причиной включения концептуальных карт в круг прикладных семантических технологий является возможность их использования для построения онтологий.

Необходимо отметить, что создание онтологий является одной из практически неизбежных задач при внедрении семантических технологий в банке или другой крупной организации. Это связано с тем, что, с одной стороны, для внедрения семантических технологий необходимо очень плотное покрытие сущностей, с которыми оперирует организация, их определением в рамках онтологий. С другой стороны, если для определения сущностей из банковской предметной области или других общепринятых предметных областей можно использовать стандартные онтологии или онтологии, построенные на основе Вики-графа, то для определения сущностей, связанных со спецификой работы конкретного банка (или другой организации), неизбежно придется выполнять работы по созданию определений уникальных классов и объектов. И именно для этой задачи может быть очень полезным использование технологии построения онтологий на базе концептуальных карт.

Рассматривая применение концептуальных карт как инструмента конструирования онтологий, необходимо, прежде всего, не вдаваясь в детали вывода конкретных понятий, определить взаимосвязь понятия концептуальной карты с понятием онтологии. Во взаимоотношении с понятием концептуальной карты онтология может быть определена как комплексное представление знаний, определяющее не только отдельные концепты, но также особенности их взаимосвязи, определенные правила формирования и ограничения. Соответственно понятие концептуальной карты в таком ключе может быть определено как графический инструмент представления концептов и связей между ними.

Использование концептуальных карт как инструмента для создания онтологий не имеет четких границ, поэтому для обозначения роли карт можно выделить два основных направления: концептуальные карты как инструмент косвенного или прямого конструирования онтологий.

Концептуальные карты как инструмент косвенного конструирования онтологий

В качестве примера использования концептуальных карт как инструмента косвенного конструирования онтологий можно рассмотреть использование концептуальной карты как посредника между экспертом и инженером на этапе сбора знаний. В данном случае концептуальная карта напрямую не является инструментом создания онтологии, а играет роль своеобразного канала связи, обеспечивающего как эффективную организацию совместной работы группы экспертов, так и оптимальное взаимодействие данной группы непосредственно с инженерами знаний [414].

Концептуальные карты как исключительно графический инструмент могут также косвенно использоваться в качестве способа непосредственной визуализации, упрощающей работу в процессе создания онтологий.

Концептуальные карты как инструмент прямого конструирования онтологий

Использование концептуальных карт как инструмента прямого конструирования онтологий возможно с использованием средств соответствующих языков описания онтологий. Например, в работе [415] представлен автоматический инструмент преобразования концептуальной карты в онтологию, описанную на языке OWL.

Одной из успешных практик прямого конструирования онтологий из концептуальных карт является создание онтологии на базе множества концептуальных карт, разработанных для определяемой предметной области. Для реализации такого подхода полезно придерживаться следующих рекомендаций по созданию «хорошо сконструированных» концептуальных карт:

- концепт самого верхнего уровня определяет фокус-вопрос карты;
- каждые три взаимосвязанных концепта формируют утверждение;
- карта представляет конкретную предметную область;
- карта имеет иерархическую структуру;
- используются краткие обозначения.

Создание онтологии на базе множества концептуальных карт предполагает следующую технологию:

- Шаг 1: Эксперты в рассматриваемой предметной области создают множество «хорошо сконструированных» концептуальных карт.
 - Шаг 2: С использованием определенных эвристических преобразований на базе этих концептуальных карт генерируются онтологии отдельных предметных областей или их подразделов (например, в формате OWL).
 - Шаг 3: Созданные онтологии совмещаются (сшиваются) в виде единой онтологии.
- о Данный алгоритм приведен на схеме, изображенной на рисунке 99.

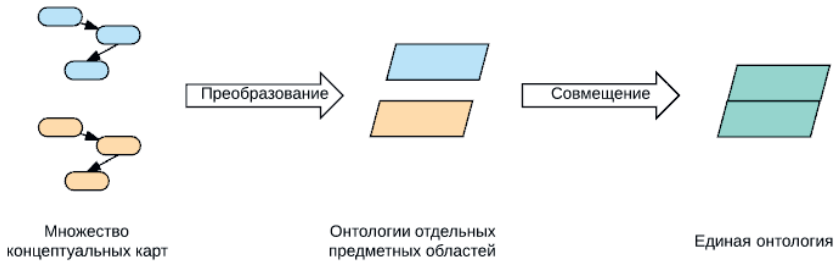


Рис. 99. Схема создания онтологии на базе множества концептуальных карт

Таким образом, концептуальные карты могут эффективно применяться как в качестве инструмента непосредственного создания онтологий, так и в качестве инструмента, использующегося косвенно в процессе конструирования.

8.3. Инструменты работы с концептуальными картами

Рассмотрим сам процесс создания концептуальных карт. Несмотря на то, что фактически концептуальные карты можно создавать, используя лист бумаги и письменные принадлежности, делать это без специального программного инструментария крайне неудобно. Причина кроется в самом алгоритме построения подобных карт, описанном в начале данного раздела. Дело в том, что процесс создания концептуальных карт предполагает прохождение определенного числа итераций, в ходе которых происходит дополнение и обновление набора обозначенных на карте концептов и отношений. Таким

образом, использование компьютерных технологий для построения концептуальных карт является наиболее удобным и эффективным. Это привело к появлению большого количества прикладных программ и инструментов, связанных с построением и редактированием концептуальных карт.

Все подобные инструменты можно разбить на две основные группы (класса):

- десктопные приложения;
- веб-сервисы.

Десктопные приложения представляют из себя полноценные классические приложения, в то время как группа веб-сервисов состоит из инструментов, использующих веб-технологии, основное взаимодействие с которыми реализуется посредством работы через веб-браузер. В следующих подразделах будут рассмотрены самые популярные инструменты каждой из групп.

Десктопные приложения для построения концептуальных карт

За счет простоты концептуальных карт их построение возможно в большинстве графических редакторов, не поддерживающих возможность создания карт напрямую. Однако в данном разделе были отобраны инструменты, непосредственно сфокусированные на поддержке процесса создания концептуальных карт.

На текущий момент наибольшее число инструментов по созданию концептуальных карт являются десктопными приложениями. К наиболее известным представителям данной группы относятся:

- SmartTools;
- Visual Understanding Environment (VUE);
- MindMapper;
- Xmind.

SmartTools – приложение, рассчитанное на использование в общеобразовательных учреждениях, исследовательских лабораториях и в процессе корпоративного обучения. В нем реализована поддержка группового онлайн-редактирования и построения смежных связей между концептуальными картами.

VUE – инструмент, рассчитанный на использование в общеобразовательных учреждениях в процессе обучения и проведения

ности, как построение бизнес-диаграмм, работа в режиме презентации, интеграция с Microsoft Office и Evernote.

Особенность данного класса инструментов заключается в наличии широкого функционала и отсутствии необходимости постоянного подключения к сети Интернет. Однако, большинство десктопных приложений не поддерживает возможность группового просмотра и редактирования концептуальных карт в режиме онлайн: из представленных выше инструментов лишь SmartTools реализует функции онлайн-просмотра и группового редактирования с использованием технологии SmartServer.

Веб-сервисы для построения концептуальных карт

Так как большинство десктопных приложений не поддерживает возможность группового просмотра и редактирования концептуальных карт в режиме онлайн, их использование при применении концептуальных карт для проектирования и управления знаниями в групповой деятельности весьма затруднительно. Для использования в командных проектах наиболее эффективными являются инструменты, использующие технологии веб, так называемые веб-приложения (веб-сервисы). Из этого можно заключить, что в большинстве проектов по использованию концептуальных карт в банках в первую очередь следует ориентироваться на веб-приложения.

Среди инструментов по работе с концептуальными картами мы рассмотрим следующие веб-сервисы:

- Lucidchart;
- Mindomo;
- TiddlyMap.

Lucidchart – веб-сервис на основе облачных технологий, представляющий собой универсальный инструмент для построения схем и диаграмм (скриншот примера работы с инструментом представлен на рисунке 101). Основной особенностью инструмента является поддержка построения большого числа диаграмм, включая концептуальные и интеллект-карты, блок-схемы и UML-диаграммы. Инструмент реализует возможность групповой работы в режиме онлайн и работы с проектами Microsoft Visio.

Mindomo – веб-приложение, рассчитанное на использование преподавателями и студентами в процессе обучения и проведение

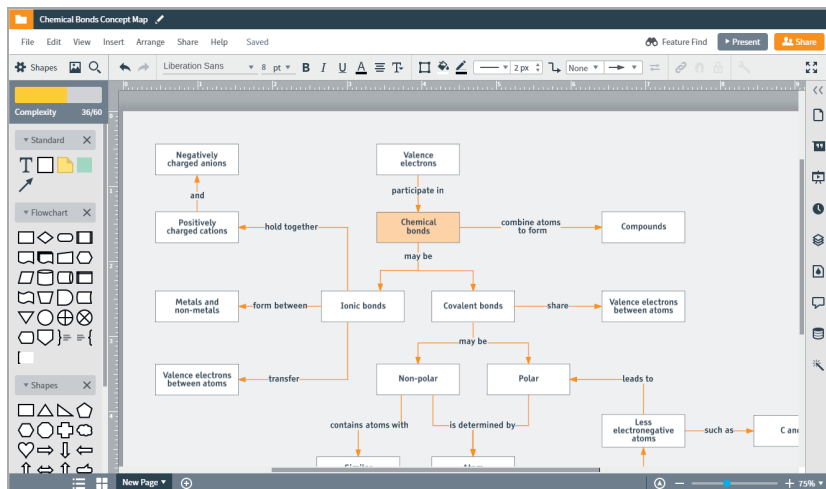


Рис. 101. Пример интерфейса веб-сервиса Lucidchart

исследовательской работы. Основными особенностями приложения являются поддержка режима презентации, возможность групповой работы в режиме онлайн и просмотра истории изменений, а также наличие большого количества типовых ознакомительных шаблонов.

TiddlyMap является специфическим веб-приложением, находящимся в экосистеме плагинов TiddlyWiki – веб-инструмента, позволяющего реализовать вики-сайт в виде одной html-страницы (дополнительная спецификация веб-инструмента TiddlyWiki представлена в разделе «Вики-технология и Википедия»). Основная особенность инструмента TiddlyMap заключается в реализации возможности работы с концептуальными картами в интеграции с концепцией вики (скриншот примера работы с инструментом представлен на рисунке 102). Каждый концепт, представленный с помощью подобной концептуальной карты, может быть описан отдельной Вики-страницей, построенной с использованием технологий TiddlyWiki. Для построения концептуальных карт в TiddlyMap используются богатые возможности JavaScript-библиотеки динамической визуализации Vis.js, реализующей как основной функционал по взаимодействию пользователя с концептуальной картой, так и дополнительные возможности, включая анимацию, гибкую графическую настройку узлов карты и т. д.

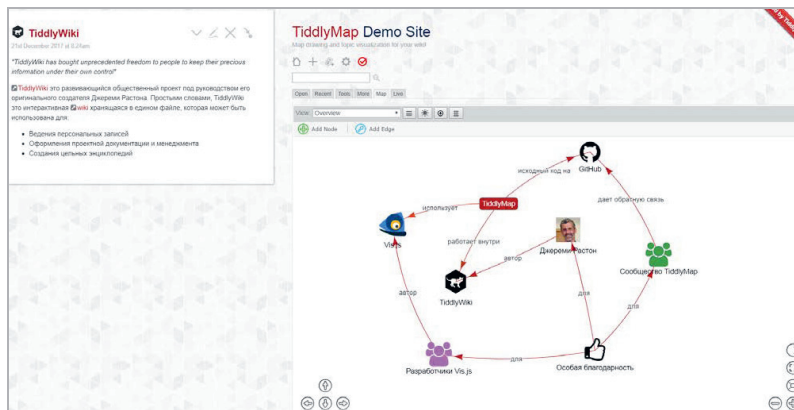


Рис. 102. Пример работы инструмента TiddlyMap

Подобная интеграция идей концептуальных карт и Вики-страниц позволяет существенно повысить эффективность использования концептуальных карт с позиции инструмента управления знаниями.

Инструменты на основе веб-технологий, в отличие от десктопных приложений, позволяют наиболее полно раскрыть функционал концептуальных карт как инструмента проектирования и управления знаниями за счет реализации возможностей группового онлайн-построения карт, а также их фактической независимости от платформы конечного устройства. Все это делает веб-инструменты наиболее подходящим вариантом при использовании концептуальных карт в ходе деятельности, предполагающей групповую работу. Однако большинству инструментов данного типа для корректной работы необходима поддержка постоянного подключения к сети. Также в сравнении с ведущими десктопными приложениями веб-инструменты, как правило, реализуют меньшее количество возможностей по работе с концептуальными картами.

8.4. Пример адаптации технологии концептуальных карт для создания прикладного семантической технологии для банка

В данном разделе мы рассмотрим практическую задачу, актуальную для большинства банков, и подходы к ее решению с использованием технологий интеллектуальных карт.

Каждый современный банк активно использует большое количество автоматизированных информационных систем (АИС) и непрерывно реализует проекты по развитию этих систем и внедрению новых АИС. Задачи анализа и проектирования доработок АИС, интеграции АИС и создания новых АИС обычно реализуются с использованием формальных языков моделирования бизнес-логики и информационных систем. Чаще всего для этого используются диаграммы UML, диаграммы семейства IDEF и их аналоги. При этом в большинстве банковских АИС основную роль играют сущности, представляющие ключевые бизнес-сущности банковской деятельности, такие как клиент, договор, счет, проводка и так далее. Однако при проектировании каждой системы для аналитиков и архитекторов обычно недоступно не только эталонное описание этих бизнес-сущностей, но и информационная база о фактических способах реализации этих типов в различных системах, уже эксплуатируемых в банке.

Таким образом для каждой АИС проектируются все новые версии стандартных сущностей клиента, договора, счета и т. д., т. е. тратится немало времени для каждой системы на разработку того, что уже разрабатывалось. Кроме того, из-за отсутствия метамодели ключевых бизнес-сущностей в случае интеграции систем в рамках АИС всего банка оказывается, что одни и те же сущности имеют разную структуру (разные наборы свойств, названия свойств, операции), что приводит к большим затратам временных и трудовых ресурсов на стыковку несогласованных объектов.

Данный кейс, типичный для большинства банков, демонстрирует прикладную задачу, требующую создания базы знаний. От данной базы знаний помимо хранения информации о референсных реализациях бизнес-сущностей и информации о фактической реализации таких метатипов в реальных системах может ожидать реализация различных сценариев поиска метатипов. Кроме обычных подходов в виде полнотекстового поиска для такой системы могут реализовываться персонализированные интеллектуальные сценарии поиска объектов.

По сути эти сценарии могут работать как автоматизированный ассистент аналитика или архитектора. В результате заполнения пользователем базы знаний тематических анкет, прохождения адаптивных вопрос-ответных сценариев (визардов или мастеров: опросников, в которых последующие вопросы выбираются в зависимости

от предыдущих ответов пользователя), статистических рекомендательных алгоритмов пользователю могут подбираться в базе знаний интересующие его сущности, группы сущностей и специфические взаимосвязи. Таким образом, база знаний может получить функционал экспертной системы. Применение подобных прикладных систем, основанных на семантических технологиях, может являться важным этапом адаптации семантических подходов для практики разработки банковских информационных систем.

Использование инструментов формальных языков моделирования для создания базы знаний

Как мы видим из рассмотренной выше задачи, такие прикладные технологии, основанные на семантических подходах, как системы управления знаниями и экспертные системы, имеют вполне актуальные приложения как в банках, так и во многих других областях. В этом разделе мы рассмотрим несколько примеров попыток создания баз знаний непосредственно на основе использования формальных языков моделирования бизнес-логики и информационных систем. Вот список наиболее популярных методологий (нотаций), используемых в этих целях:

- UML;
- IDEF0;
- IDEF1X;
- DFD;
- ERM.

Для удобства далее мы будем рассматривать только диаграммы UML, преимущественно диаграммы классов языка UML. Наш интерес к таким решениям объясняется тем, что в случае продуктивности такого подхода рассматриваемую нами задачу можно было бы решить, работая в том же типе диаграмм, в котором строятся описания АИС.

Инструменты проектирования на основе использования формальных языков моделирования бизнес-логики и информационных систем изначально не были предназначены для проектирования или поддержки баз знаний. Тем не менее в связи с рядом причин они все же стали использоваться и в этой области. Эти причины можно проследить на примере UML как наиболее распространенного и известного инструмента построения моделей:

- UML сам по себе является достаточно популярным языком и международным стандартом построения моделей и проектирования информационных систем;
- в UML имеется четкая и строгая система типов, что и позволяет ему являться унифицированным стандартом;
- UML работает с объектно-ориентированной парадигмой. Это определяет возможность генерации кода на основе некоторых моделей, таким образом процесс проектирования значительно упрощается;
- UML достаточно прост в изучении;
- UML популярен и существует достаточно давно, поэтому присутствует огромное количество программного обеспечения для работы с ним.

В целом все эти причины верны и для остальных, указанных выше, формальных нотаций.

Среди примеров использования UML в области управления знаниями и экспертных систем можно выделить опыт Томского политехнического института [416]. Суть данного проекта заключается в том, что для систем искусственного интеллекта стандарт ISO 13606, определяющий основные медицинские типы и отношения между ними, не подходил: набор видов отношений между типами был узок и не позволял реализовывать формирование адекватного решения машиной. Решением данной проблемы являлось создание информационной модели, описывающей уже существующие медицинские типы, но уже с использованием языка UML и его более широкого набора связей и отношений.

Еще одним примером использования UML является работа [417], демонстрирующая вариант создания и управления базой знаний на базе нотации UML. В данной работе показано, каким образом используется UML в подобных задачах, а также каким образом на основе UML-модели можно создать базу знаний на OWL.

Несмотря на свою популярность, инструменты формальных языков моделирования имеют ряд существенных недостатков при использовании их для создания баз знаний:

- у большинства из них отсутствуют средства, обеспечивающие коллективную работу;
- в них отсутствуют инструменты генерации онтологий (в приведенном выше примере показано, что создание такого инструмента довольно трудозатратно);

- крайне строгая формализация как типов, так и связей: строгая нотация сильно ограничивает возможности передачи специфики отношений объектов;
- сильно осложненный процесс комбинирования (объединения) двух моделей: из-за сильной формализованности нотаций для совмещения моделей может понадобиться перепроектировать каждую из них.

Таким образом, становится очевидно, что применение инструментов формальных языков моделирования не подходит для проектирования и поддержки работы базы знаний.

Подходы к адаптации технологии концептуальных карт

В данном разделе мы рассмотрим, каким образом можно применить технологию построения концептуальных карт для решения рассматриваемой задачи создания базы знаний и экспертной системы, содержащей метамодель формальных описаний различных АИС банка.

Суммируя все, о чем говорилось ранее, можно выделить следующие ключевые возможности и преимущества технологии концептуальных карт:

- возможность участвовать в процессе управления знаниями: карта может быть организована в виде базы знаний, с возможностью обеспечения процессов управления этими знаниями;
- отсутствие строгих определений или нотаций: структура концептуальной карты существенно более гибка и менее формализована, поэтому можно создать собственный набор отношений и типов, удовлетворяющий нуждам конкретной задачи, например, как показывалось выше, с помощью концептуальной карты можно реализовать ту же UML-нотацию;
- простота создания онтологий на основе концептуальной модели, сформулированной в виде концептуальной карты;
- наглядность и понятность (концептуальная карта проста для изучения), а главное — ориентированность на визуальное представление, что делает ее отличным интерфейсом для базы знаний;
- вследствие отсутствия нотации и методологии совмещение различных моделей не вызывает никаких конфликтов по структуре.

Разумеется, отдельно взятая концептуальная карта не может являться полноценной экспертной системой. Поэтому необходимо создать

систему, включающую в качестве компонентов как инструментарий по работе с концептуальной картой, так и программное обеспечение, реализующее дополнительные возможности, необходимые для построения экспертной системы.

Предлагаемая адаптация технологии концептуальных карт для создания экспертной системы для банка должна состоять из базы знаний, организованной в виде интерактивной концептуальной карты, и дополнительного блока информации, содержащегося на уровне каждого концепта. Таким образом каждый элемент карты (концепт) сможет иметь определенную совокупность свойств, наименование и описание. По сути такая база знаний будет близка по структуре к графу свойств, описанному в разделе, посвященном технологиям Вики-графа.

Для реализации данного решения одной концептуальной карты недостаточно. Для более содержательного формирования и изучения объектных моделей концепты на карте должны иметь подробное описание. Кроме того, это описание должно быть структурированным, исчерпывающим и понятным. Примером таких описаний может служить энциклопедическая статья, которая к тому же использует перекрестные ссылки на другие понятия и объекты, описываемые в энциклопедии. Современным информационным аналогом энциклопедии является ранее описанная Википедия: она организует данные в виде разделов, предоставляет широкие возможности по форматированию текста и работе с мультимедиа, а также основным функциональным компонентом Википедии является механизм взаимосвязи различных статей через ссылки.

Для реализации этой части решения отлично подходит упомянутая выше технология TiddlyMap, которая совмещает в себе функционал концептуальной карты и описательных вики-подобных страниц для каждого элемента этой карты. Эта технология является плагином для TiddlyWiki – проекта с открытым кодом, основной особенностью которого является размещение всего контента сайта на одной html-странице, что обеспечивает мобильность системе, создаваемой на базе этого проекта.

Таким образом, в нашем распоряжении будет представление базы знаний в виде концептуальной карты со структурированным описанием концептов на естественном языке на основе вики-разметки. Данное решение позволит совместить преимущества концептуальной карты, такие как:

- максимально гибкая структура для структурирования сложных систем;
- наглядность представления сложных отношений;
- простота редактирования;
- со структурированным описанием концептов на естественном языке на основе вики-разметки, что позволит:
- сопровождать концепты хорошо оформленным и структурированным текстом со всеми возможностями гипертекстовых документов (т. е. со ссылками внутри документов, между документами и ссылками на внешние ресурсы);
- использовать механизм шаблонов, позволяющий удобно создавать унифицированные хорошо организованные документы со сложной структурой;
- построить простой и эффективный механизм категоризации страниц с помощью иерархических категорий;
- привязать к документу произвольный набор именованных свойств, необходимый для построения интеллектуальных сценариев навигации по концептуальной сети (напр., для реализации интеллектуальных сценариев подбора бизнес-сущностей).

Для организации быстрого перемещения по концептуальной карте, так как она может быть довольно объемной, необходимо реализовать поиск интересующих пользователя концептов. Такой поиск может быть организован в виде опросных форм или мастеров (wizards), которые в вопросно-ответной форме обеспечивают поиск схожих концептов (объектов, процессов), что позволяет аналитику проводить интервью участников бизнес-процесса на основе этих вопросов и составлять объектную модель на основе найденных в базе знаний объектов и концептов. Сама реализация мастеров заключается в создании соответствующего графического интерфейса, что в целом сводится к созданию дополнительных программных компонентов и совмещению их с технологией TiddlyMap. В силу того, что это технология с открытым кодом, реализация решения сводится к созданию плагина к TiddlyWiki на основе TiddlyMap с добавлением функционала опросных форм.

Важным компонентом опросных форм является алгоритмическая организация структуры концептов. Суть этого компонента заключается в организации элементов концептуальной карты таким образом, чтобы путем выбора определенных ответов в вопросных

Глава 8

формах пользователь мог сужать множество элементов (концептов), постепенно сводя его к одному элементу. Помимо этого, данный компонент должен предоставлять возможность автоматического создания вопросно-ответной логики на основе концептов карты, связей между ними и их описаний. Этот компонент может быть реализован как дополнительный язык программирования высокого уровня или же построен с использованием семантических технологий, о которых говорилось ранее (OWL, RDF).

Таким образом предлагаемое решение представляет из себя нечто среднее между базой знаний и экспертной системой. Ее функционал предоставляет пользователям организованную совокупность объектных моделей, с помощью которых повышается эффективность аналитики и снижаются временные потери на проектирование компонентов корпоративной информационной системы банка. Предложенное технологическое решение демонстрирует возможность синергии вики-технологий и концептуальных карт для построения базы знаний, которая позволяет создавать ценные приложения для решения банковских задач.

ЛИТЕРАТУРА

1. «Стратегия развития информационного общества в Российской Федерации на 2017–2030 годы», утвержденная указом Президента РФ от 9 мая 2017 г. № 203.
2. Sound Practices: Implications of Fintech Developments for Banks and Bank Supervisors: Consultative Document. Basel Committee on Banking Supervision. 2017. URL: <https://www.bis.org/bcbs/publ/d415.pdf>.
3. Financial Technology Startup Highlights – Q1 2018 // Venture Scanner. 30.04.2018. URL: <https://www.venturescanner.com/financial-technology>.
4. Blockchain Technology Sector Overview – Q1 2018 // Venture Scanner. 22.04.2018. URL: <https://www.venturescanner.com/blockchain-technology>.
5. *Салтыкова М.* США и Европа: Вместо блокчейна – регтех. Почему банки инвестируют в финтех, но обходят стороной блокчейн // Rusbase. 01.03.2018. URL: <https://rb.ru/analytics/banki-fintech-blockchain>.
6. 2017 Fintech 100: Leading Global Fintech Innovators. KPMG. URL: <https://home.kpmg.com/content/dam/kpmg/it/pdf/2018/03/H2-Fintech-Innovators-2017.pdf>.
7. Fintech: Карта российского рынка (2018). Rusbase. URL: <https://rb.ru/fintech>.
8. *Nielsen J. F.* Internet technology and customer linking in Nordic banking // International Journal of Service Banking. 2002. V. 13.

- № 2. P. 475–496. URL: <https://www.emeraldinsight.com/doi/abs/10.1108/09564230210447940>.
9. *Sathye M.* Adoption of internet banking by Australian consumers: An empirical investigation // *The International Journal of Bank Marketing*. 1999. V. 17. № 7. P. 324–339. URL: <https://www.emeraldinsight.com/doi/abs/10.1108/02652329910305689>.
 10. *Diemers D., Lamaa A., Salamat J., Steffens T.* Developing a FinTech ecosystem in the GCC. Dubai: Strategy, 2015. URL: <http://www.strategyand.pwc.com/media/file/Developing-a-FinTech-ecosystem-in-the-GCC.pdf>.
 11. *Lee I. Shin Y. Sh.* Fintech: Ecosystem, business models, investment decisions, and challenges // *Business Horizons*. 2018. V. 61. № 1. P. 35–46. URL: <http://www.sciencedirect.com/science/article/pii/S0007681317301246>.
 12. *Walchek S.* The unbundling of finance // *TechCrunch*. 29.05.2015. URL: <https://techcrunch.com/2015/05/29/the-unbundling-of-finance>.
 13. Проект Федерального закона Российской Федерации «О цифровых финансовых активах». Опубликован 25.01.2018. URL: https://www.cbr.ru/Content/Document/File/36003/cfa_project.pdf.
 14. Проект Федерального закона Российской Федерации «Об альтернативных способах привлечения инвестиций (краудфандинге)». Опубликован 25.01.2018. URL: https://www.cbr.ru/analytics/standart_acts/others/20180125_02.pdf.
 15. *Crowdfunding's Potential for the Developing World*. 2013. infoDev, Finance and Private Sector Development Department. Washington, D. C.: World Bank, 2013. URL: <http://documents.worldbank.org/curated/en/409841468327411701/pdf/840000WP0Box380crowdfunding0study00.pdf>.
 16. *Алексеевских А.* Рынок краудфандинга вырос на 83% // *Известия*. 28.03.2018. URL: <https://iz.ru/724772/anastasiia-alekseevskikh/ry-nok-kraudfandinga-vyros-na-83>.
 17. Поток. URL: <https://business.potok.digital>.
 18. *Гайдаев В.* Интеллект может быть искусственным, но разум нужен человеческий // *Коммерсант*. 28.03.2018. URL: <https://www.kommersant.ru/doc/3580435>.
 19. *Markswebb Mobile Banking Rank 2017*. URL: <http://markswebb.ru/e-finance/mobile-banking-rank-2017>.
 20. *Durand D.* Risk elements in consumer instalment financing // *NBER Books*. 1941. V. 8. P. 105–142.

21. *Hand D. J., Henley W. E.* Statistical classification methods in consumer credit scoring: a review // *Journal of the Royal Statistical Society: Series A (Statistics in Society)*. 1997. V. 160. № 3. P. 523–541.
22. *García V., Marqués A. I., Sánchez J. S.* An insight into the experimental design for credit risk and corporate bankruptcy prediction systems // *Journal of Intelligent Information Systems*. 2015. V. 44. № 1. C. 159–189.
23. *Lessmann S., Baesens B., Seow H. V., Thomas L. C.* Benchmarking state-of-the-art classification algorithms for credit scoring: An update of research // *European Journal of Operational Research*. 2015. V. 247. № 1. P. 124–136.
24. *Hand D. J., Kelly M. G.* Superscorecards // *IMA Journal of Management Mathematics*. 2002. V. 13. № 4. P. 273–281.
25. *Pavlidis N. G., Tasoulis D. K., Adams N. M., Hand D. J.* Adaptive consumer credit classification // *Journal of the Operational Research Society*. 2012. V. 63. № 12. P. 1645–1654.
26. *Yap B. W., Ong S. H., Husain N. H. M.* Using data mining to improve assessment of credit worthiness via credit scoring models // *Expert Systems with Applications*. 2011. V. 38. № 10. P. 13274–13283.
27. *Khemais Z., Nesrine D., Mohamed M.* Credit Scoring and Default Risk Prediction: A Comparative Study between Discriminant Analysis & Logistic Regression // *International Journal of Economics and Finance*. 2016. V. 8. № 4. P. 39.
28. *Li Z.* Reject Inference in Credit Scoring Using Support Vector Machines // *SSRN*. 2016.
29. *Louzada F., Anacleto-Junior O., Candolo C., Mazucheli J.* Poly-bagging predictors for classification modelling for credit scoring // *Expert Systems with Applications*. 2011. V. 38. № 10. P. 12717–12720.
30. *Fisher R. A.* The use of multiple measurements in taxonomic problems // *Annals of eugenics*. 1936. V. 7. № 2. P. 179–188.
31. *Eisenbeis R. A.* Problems in applying discriminant analysis in credit scoring models // *Journal of Banking & Finance*. 1978. V. 2. № 3. P. 205–219.
32. *Mylonakis J., Diacogiannis G.* Evaluating the likelihood of using linear discriminant analysis as a commercial bank card owners credit scoring model // *International business research*. 2010. V. 3. № 2. P. 9.
33. *Akkoç S.* An empirical comparison of conventional techniques, neural networks and the three stage hybrid Adaptive Neuro Fuzzy Inference System (ANFIS) model for credit scoring analysis: The case of

- Turkish credit card data // *European Journal of Operational Research*. 2012. V. 222. № 1. P. 168–178.
34. *Falangs K., Glen J.J.* Heuristics for feature selection in mathematical programming discriminant analysis models // *Journal of the Operational Research Society*. 2010. V. 61. № 5. P. 804–812.
 35. *Breiman L., Friedman J., Stone C.J., Olshen R.A.* Classification and regression trees. CRC Press, 1984.
 36. *Loh W. Y.* Fifty years of classification and regression trees // *International Statistical Review*. 2014. V. 82. № 3. P. 329–348.
 37. *Lessmann S., Baesens B., Seow H. V., Thomas L. C.* Benchmarking state-of-the-art classification algorithms for credit scoring: An update of research // *European Journal of Operational Research*. 2015. V. 247. № 1. P. 124–136.
 38. *Finlay S.* Multiple classifier architectures and their application to credit risk assessment // *European Journal of Operational Research*. 2011. V. 210. № 2. P. 368–378.
 39. *Hu Q., Che X., Zhang L., Zhang D., Guo M., Yu D.* Rank entropy-based decision trees for monotonic classification // *IEEE Transactions on Knowledge and Data Engineering*. 2012. V. 24. № 11. P. 2052–2064.
 40. *Zhang D., Zhou X., Leung S. C., Zheng J.* Vertical bagging decision trees model for credit scoring // *Expert Systems with Applications*. 2010. V. 37. № 12. P. 7838–7843.
 41. *Hayashi Y., Tanaka Y., Takagi T., Saito T., Iiduka H., Kikuchi H., Mitra S.* Recursive-rule extraction algorithm with J48graft and applications to generating credit scores // *Journal of Artificial Intelligence and Soft Computing Research*. 2016. V. 6. № 1. P. 35–44.
 42. *Vapnik V.N., Vapnik V.* Statistical learning theory. N. Y.: Wiley, 1998. V. 1.
 43. *Bellotti T., Crook J.* Support vector machines for credit scoring and discovery of significant features // *Expert Systems with Applications*. 2009. V. 36. № 2. P. 3302–3308.
 44. *Chen W., Ma C., Ma L.* Mining the customer credit using hybrid support vector machine technique // *Expert Systems with Applications*. 2009. V. 36. № 4. P. 7611–7616.
 45. *Ling Y., Cao Q., Zhang H.* Credit scoring using multi-kernel support vector machine and chaos particle swarm optimization // *International Journal of Computational Intelligence and Applications*. 2012. V. 11. № 3. P. 12500198:1-12500198:13.

46. *Friedman N., Geiger D., Goldszmidt M.* Bayesian network classifiers // *Machine learning*. 1997. V. 29. № 2–3. P. 131–163.
47. *Pearl J.* Probabilistic reasoning in intelligent systems: networks of plausible inference. Morgan Kaufmann, 1988.
48. *Giudici P.* Bayesian data mining, with application to benchmarking and credit scoring // *Applied Stochastic Models in Business and Industry*. 2001. V. 17. № 1. P. 69–81.
49. *Gemela J.* Financial analysis using Bayesian networks // *Applied Stochastic Models in Business and Industry*. 2001. V. 17. № 1. P. 57–67.
50. *Antonakis A. C., Sfakianakis M. E.* Naïve Bayes as a means of constructing application scorecards // *Advances in Doctoral Research in Management*. 2008. V. 2. P. 47–62.
51. *Antonakis A. C., Sfakianakis M. E.* Assessing naive Bayes as a method for screening credit applicants // *Journal of applied Statistics*. 2009. V. 36. № 5. P. 537–545.
52. *Wu W. W.* Improving classification accuracy and causal knowledge for better credit decisions // *International journal of neural systems*. 2011. V. 21. № 4. P. 297–309.
53. *Zhu H., Beling P. A., Overstreet G. A.* A Bayesian framework for the combination of classifier outputs // *Journal of the Operational Research Society*. 2002. V. 53. № 7. P. 719–727.
54. *West D.* Neural network credit scoring models // *Computers & Operations Research*. 2000. V. 27. № 11. P. 1131–1152.
55. *Ong C. S., Huang J. J., Tzeng G. H.* Building credit scoring models using genetic programming // *Expert Systems with Applications*. 2005. V. 29. № 1. P. 41–47.
56. *Breiman L.* Bagging predictors // *Machine learning*. 1996. V. 24. № 2. P. 123–140.
57. *Wolpert D. H.* Stacked generalization // *Neural networks*. 1992. V. 5. № 2. P. 241–259.
58. *Vukovic S., Delibasic B., Uzelac A., Suknovic M.* A case-based reasoning model that uses preference theory functions for credit scoring // *Expert Systems with Applications*. 2012. V. 39. № 9. P. 8389–8395.
59. *Marqués A. I., García V., Sánchez J. S.* Two-level classifier ensembles for credit risk assessment // *Expert Systems with Applications*. 2012. V. 39. № 12. P. 10916–10922.
60. *Akkoç S.* An empirical comparison of conventional techniques, neural networks and the three stage hybrid Adaptive Neuro Fuzzy Inference System (ANFIS) model for credit scoring analysis: The case of

- Turkish credit card data // *European Journal of Operational Research*. 2012. V. 222. № 1. P. 168–178.
61. *Clemen R. T., Murphy A. H., Winkler R. L.* Screening probability forecasts: contrasts between choosing and combining // *International Journal of Forecasting*. 1995. V. 11. № 1. P. 133–145.
 62. *DeGroot M., Eriksson E. A.* Probability forecasting, stochastic dominance, and the Lorenz curve // *Bayesian Statistics*. 1985. V. 2. P. 99–118.
 63. *DeGroot M. H., Fienberg S. E.* The comparison and evaluation of forecasters // *The statistician*. 1983. P. 12–22.
 64. *Zhu H., Beling P. A., Overstreet G. A.* A Bayesian framework for the combination of classifier outputs // *Journal of the Operational Research Society*. 2002. V. 53. № 7. P. 719–727.
 65. Формы отчетности кредитных организаций. Банк России. URL: <https://www.cbr.ru/credit/forms.asp>.
 66. Basel Committee on Banking Supervision. Customer due diligence for Banks. Zürich, 2001. P. 3.
 67. Федеральный закон от 10.07.2002 № 86-ФЗ (ред. от 29.12.2014) «О Центральном банке Российской Федерации (Банке России)» (принят ГД ФС РФ 27.06.2002) // СПС «Консультант плюс».
 68. *Hawawini G., Keim D.* On the predictability of common stock returns: Worldwide evidence // *Handbooks in Operations Research and Management Science*. V. 9 (Finance) / Eds R. Jarrow et al. Amsterdam: Elsevier Science, 1995. P. 497–544.
 69. *Fama E. F.* The behavior of stock market price // *The Journal of Business*, 1965. V. 38. № 1. P. 34–105.
 70. *Basu S.* The investment performance of common stocks in relation to their price-earnings ratios: A test of the efficient market hypothesis // *Journal of Finance*. 1977. V. 32. P. 663–682.
 71. *Campbell J.* Stock returns and the term structure // *Journal of Financial Economics*. 1987. V. 18. P. 373–399.
 72. *Dourra H., Siy P.* Investment using technical analysis and fuzzy logic // *Fuzzy Sets and Systems*. 2002. V. 127. P. 221–240.
 73. *Fama E. F.* Efficient capital markets // *Journal of Finance*. 1991. V. 46. № 5. P. 1575–1617.
 74. *Fama E. F., French K.* Dividend yields and expected stock returns // *Journal of Financial Economics*. 1998. V. 22. P. 3–25.
 75. *Fama E. F., French K.* Permanent and temporary components of stock prices // *Journal of Political Economy*. 1998. V. 96. P. 246–273.

76. *Fama E. F., Schwert W.* Asset returns and inflation // *Journal of Financial Economics*. 1977. V. 5. P. 115–146.
77. *Edwards R., Magee J., Bassetti W. H. C.* *Technical Analysis of Stock Trends*. Boca Raton: CRC Press, 2013.
78. *Epps T. W., Epps M.* The stochastic dependence of security price changes and transactions volumes: Implications for the mixture of distribution hypothesis // *Econometrica*. 1976. V. 44. P. 305–321.
79. *Plummer T.* *Forecasting Financial Markets: Technical Analysis and the Dynamics of Price*. N. Y.: Wiley, 1990.
80. *Smirlock M., Starks L.* An empirical analysis of the stock price–volume relationship // *Journal of Banking and Finance*. 1990. V. 12. P. 31–42.
81. *Treynor J., Ferguson R.* In defense of technical analysis // *Journal of Finance*. 1985. V. 40. № 3. P. 757–773.
82. *Zarrabi N., Snaith S., Coakley J.* FX technical trading rules can be profitable sometimes! // *International Review of Financial Analysis*. 2017. V. 49. P. 113–127.
83. *Fang J., Qin Y., Jacobsen B.* Technical market indicators: An overview // *Journal of Behavioral and Experimental Finance*. 2014. V. 4. P. 25–56.
84. *Da Costa T. R. C. C., Nazario R. T., Bergo G. S. C., Sobreiro V. A., Kumura H.* Trading system based on the use of technical analysis: A computational experiment // *Journal of Behavioral and Experimental Finance*. 2015. V. 6. P. 42–55.
85. *Escobar A., Moreno J., Munera S.* A technical analysis indicator // *Electronic Notes in Theoretical Computer Science*. 2013. V. 292. P. 27–37.
86. *Путко Б. А., Диденко А. С., Дубовиков М. М.* Модель волатильности обменного курса валют (RUR/USD), построенная на основе фрактальных характеристик финансового ряда // *Прикладная эконометрика*. 2014. № 36 (4). С. 79–87.
87. *Arévalo A., Niño J., Hernández G., Sandoval J.* High-frequency trading strategy based on deep neural networks // *Lecture Notes in Computer Science*. 2016. V. 9773. P. 424–436.
88. *Atsalakis G.* Exchange rate forecasting by neuro-fuzzy techniques // *Journal of Financial Decision Making*. 2006. V. 1. № 2. P. 15–26.
89. *Chen Y., Yanga B., Abraham A.* Flexible neural trees ensemble for stock index modelling // *Neurocomputing*. 2006. V. 70. P. 697–703.
90. *Cervello-Royo R., Guijarro F., Michniuk K.* Stock market trading rule based on pattern recognition and technical analysis: Forecasting the djia index with intraday data // *Expert Systems with Applications*. 2015. V. 42. № 14. P. 5963–5975.

91. *Chen T.-I., Chen F.-Y.* An intelligent pattern recognition model for supporting investment decisions in stock market // *Information Sciences*. 2016. V. 346. P. 261–274.
92. *Chiang W.-C., Enke D., Wu T., Wang R.* An adaptive stock index trading decision support system // *Expert Systems with Applications*. 2016. V. 59. P. 195–207.
93. *Chourmouziadis K., Chatzoglou P. D.* An intelligent short-term stock trading fuzzy system for assisting investors in portfolio management // *Expert Systems with Applications*. 2016. V. 43. P. 298–311.
94. *Enke D., Thawornwong S.* The use of data mining and neural networks for forecasting stock market returns // *Expert Systems with Applications*. 2005. V. 29. P. 927–940.
95. *Enke D., Mehdiyev N.* Stock market prediction using a combination of stepwise regression analysis, differential evolution-based fuzzy clustering and a fuzzy inference neural network // *Intelligent Automation and Soft Computing*, 2013. V. 19. № 4. P. 636–648.
96. *Halliday R.* Equity trend prediction with neural networks // *Research Letters in the Information and Mathematical Sciences*. 2004. V. 6. P. 135–149.
97. *Niaki S. T. A., Hoseinzade S.* Forecasting S&P 500 index using artificial neural networks and design of experiments // *Journal of Industrial Engineering International*. 2013. V. 9. № 1. P. 1–9.
98. *Patel J., Shah S., Thakkar P., Kotecha K.* Predicting stock market index using fusion of machine learning techniques // *Expert Systems with Applications*. 2015. V. 42. P. 2162–2172.
99. *Qiu M., Song Y., Akagi F.* Application of artificial neural network for the prediction of stock market returns: The case of the Japanese stock market // *Chaos, Solitons and Fractals*. 2016. V. 85. P. 1–7.
100. *Zhong X., Enke D.* Forecasting daily stock market return using dimensionality reduction // *Expert Systems with Applications*. 2017. V. 67. P. 126–139.
101. *Huang C.-Y., Lin P. K. P.* Application of integrated data mining techniques in stock market forecasting // *Cogent Economics and Finance*. 2014. V. 2. № 1. P. 92905–92921.
102. *Doesken B., Abraham A., Thomas J., Paprzycki M.* Real stock trading using soft computing models // *International Conference on Information Technology: Coding and Computing (ITCC 05)*. 2005. V. 2. P. 162–167.
103. *Fernandez-Rodriguez F., Gonzalez-Martel C., Sosvilla-Rivebo S.* On the profitability of technical trading rules based on artificial neural net-

- works: Evidence from the Madrid stock market // *Economics Letters*. 2000. V. 69. P. 89–94.
104. *Chong E., Han C., Park F. C.* Deep Learning Networks for Stock Market Analysis and Prediction: Methodology, Data Representations and Case Studies // *Expert Systems with Applications*. 2017. V. 83. P. 187–205.
105. *Atsalakis G. S., Valavanis K. P.* Forecasting stock market short-term trends using a neuro-fuzzy based methodology // *Expert Systems with Applications*. 2009. V. 36. P. 10696–10707.
106. *Nair B. B., Mohandas V. P., Sakthivel N. R.* A Genetic Algorithm Optimized Decision TreeSVM based Stock Market Trend Prediction System // *International Journal on Computer Science and Engineering*. 2010. V. 2. № 9. P. 2981–2988.
107. *Volkova E. S., Gisin V. B., Soloviev V. I.* Data mining techniques: Modern approaches to application in credit scoring // *Digest Finance*. 2017. V. 22. № 4 (244). P. 400–412.
108. *Волкова Е. С., Гисин В. Б., Соловьев В. И.* Современные подходы к применению методов интеллектуального анализа данных в задаче кредитного скоринга // *Финансы и кредит*. 2017. Т. 23. № 34 (754). С. 2044–2060.
109. *Волкова Е. С., Гисин В. Б., Соловьев В. И.* Методы теории нечетких множеств в кредитном скоринге // *Финансы и кредит*. 2017. Т. 23. № 35 (755). С. 2088–2106.
110. *Radosteva M., Soloviev V., Ivanyuk V., Tsvirkun A.* Use of neural network models in market risk management // *Advances in Systems Science and Applications*. 2018. V. 18. № 2. P. 53–58.
111. *Elizarov M., Ivanyuk V., Soloviev V., Tsvirkun A.* Identification of high-frequency traders using fuzzy logic methods // 10th International Conference Management of Large-Scale System Development (MLSD). Piscataway, USA: IEEE, 2017. P. 1–4.
112. *Atsalakis G. S., Valavanis K. P.* Surveying stock market forecasting techniques. Part II: Soft computing methods // *Expert Systems with Applications*. 2009. V. 36. P. 5932–5941.
113. *Soloviev V.* Forecasting stock market turnovers with boosted decision trees // 11th International Conference on Application of Information and Communication Technologies (AICT). Piscataway, USA: IEEE, 2017. P. 140–143.
114. *Russakovsky O., Deng J., Su H., Krause J., Satheesh S., Ma S., Bernstein M.* ImageNet large scale visual recognition challenge. *International Journal of Computer Vision*. 115 (3). P. 211–252.

115. *Krizhevsky A., Sutskever I., Hinton G. E.* Image Net classification with deep convolutional neural networks // F. Pereira, C.J.C. Burgess, L. Bottou, K. Q. Weinberger (Eds). Advances in neural information processing systems, 25 (p. 1097–1105). Red Hook, N. Y.: Curran, 2012.
116. *Simonyan K., Zisserman A.* Very deep convolutional networks for large-scale image recognition, 2014. arXiv:1409.1556.
117. *Zeiler M. D., Fergus R.* Visualizing and understanding convolutional networks // Proceedings of the European Conference on Computer Vision (p. 818–833). Berlin: Springer, 2014.
118. *Szegedy C., Ioffe S., Vanhoucke V.* Inception-v4, Inception-Resnet and the impact of residual connections on learning, 2016. arXiv:1602.07261.
119. *Karpathy A.* CS231n: Convolutional neural networks for visual recognition, 2016.
120. *LeCun Y., Bottou L., Bengio Y., Haffner P.* Gradient-based learning applied to document recognition // Proceedings of the IEEE. 1998. V. 86 (11). P. 2278–2324.
121. *LeCun Y., Bengio Y., Hinton G.* Deep learning. Nature. 2015. V. 521 (7553). P. 436–444.
122. *Hinton G. E., Osindero S., Teh Y.* A fast learning algorithm for deep belief nets // Neural Computation. 2006. V. 18 (7). P. 1527–1554.
123. *Hinton G. E., Salakhutdinov R. R.* Reducing the dimensionality of data with neural networks // Science. 2006. V. 313 (5786). P. 504–507.
124. *Bengio Y., Lamblin P., Popovici D., Larochelle H.* Greedy layer-wise training of deep networks // J. C. Platt, D. Koller, Y. Singer, S. T. Roweis (Eds). Advances in neural information processing systems. 19 (p. 2814–2822). Red Hook, N. Y.: Curran, 2006.
125. *Deng L., Yu D.* Deep learning: Methods and applications // Foundations and Trends in Signal Processing. 2014. V. 7 (3–4). P. 197–387.
126. *Chellapilla K., Puri S., Simard P.* High performance convolutional neural networks for document processing // Proceedings of the 10th International Workshop on Frontiers in Handwriting Recognition. Los Alamitos, CA: IEEE Computer Society, 2006.
127. *Ranzato M. A., Huang F. J., Boureau Y., LeCun Y.* Unsupervised learning of invariant feature hierarchies with applications to object recognition // Proceedings IEEE Conference on Computer Vision and Pattern Recognition (p. 1–8). Los Alamitos, CA: IEEE Computer Society, 2007.
128. *Lin M., Chen Q., Yan S.* Network in network, 2013. arXiv:1312.4400.

129. *Zeiler M. D., Fergus R.* Stochastic pooling for regularization of deep convolutional neural networks. 2013. arXiv:1301.3557.
130. *Gong, Y., Wang L., Guo R., Lazebnik S.* Multi-scale orderless pooling of deep convolutional activation features // Proceedings of the European Conference on Computer Vision (p. 392–407). Berlin: Springer. 2014. September.
131. *Szegedy C., Vanhoucke V., Ioffe S., Shlens J., Wojna Z.* Rethinking the Inception architecture for computer vision, 2015. arXiv:1512.00567.
132. *He K., Zhang X., Ren S., Sun J.* Deep residual learning for image recognition, 2015. arXiv:1512.03385.
133. *Xu B., Wang N., Chen T., Li M.* Empirical evaluation of rectified activations in convolutional network, 2015. arXiv:1505.00853v2.
134. *Tang Y.* Deep learning using linear support vector machines, 2013. arXiv:1306.0239.
135. *Zhao Q., Griffin L. D.* Suppressing the unusual: Towards robust CNNs using symmetric activation functions, 2016. arXiv:1603.05145v1.
136. *Hinton G. E., Srivastava N., Krizhevsky A., Sutskever I., Salakhutdinov R. R.* Improving neural networks by preventing co-adaptation of feature detectors, 2012. arXiv:1207.0580.
137. *Glorot X., Bengio Y.* Understanding the difficulty of training deep feedforward neural networks // Proceedings of the 13th International Conference on Artificial Intelligence and Statistics, 2010. P. 249–256.
138. *Szegedy C., Zaremba W., Sutskever I., Bruna J., Erhan D., Goodfellow I., Fergus R.* Intriguing properties of neural networks // Proceedings of the 1st International Conference on Learning Representations, 2014 (p. 1–10). N. p.: Computational and Biological Learning Society.
139. *Bengio Y.* Learning deep architectures for AI. Foundations and Trends in Machine Learning. 2009. V. 2 (1). P. 1–127.
140. *Schmidhuber J.* Deep learning in neural networks: An overview // Neural Networks. 2015. V. 61. P. 85–117.
141. *Deng L.* A tutorial survey of architectures, algorithms, and applications for deep learning // APSIPA Transactions on Signal and Information Processing. 2014. V. 3 (2). P. 1–29.
142. *Guo Y., Liu Y., Oerlemans A., Lao S., Wu S., Lew M. S.* Deep learning for visual understanding: A review // Neurocomputing. 2016. V. 187. P. 27–48.
143. *Gu S., Rigazio L.* Towards deep neural network architectures robust to adversarial examples, 2014. arXiv:1412.5068.

144. *Srinivas S., Sarvadevabhatla R. K., Mopuri K. R., Prabhu N., Kruthiventi S. S., Babu R. V.* A taxonomy of deep convolutional neural nets for computer vision, 2016. arXiv:1601.06615.
145. *Rawat W., Wang Z.* Deep Convolutional Neural Networks for Image Classification: A Comprehensive Review // *Neural Computation*. 2017. V. 29 (9). P. 2352–2449.
146. *Yu D., Wang H., Chen P., Wei Z.* Mixed pooling for convolutional neural networks // *Proceedings of the 9th International Conference on Rough Sets and Knowledge Technology* (p. 364–375). Berlin: Springer, 2014.
147. *Nair V., Hinton G. E.* Rectified linear units improve restricted Boltzmann machines // *Proceedings of the 27th International Conference on Machine Learning* (p. 807–814). N. p.: International Machine Learning Society, 2010.
148. *LeCun Y., Boser B., Denker J. S., Henderson D., Howard R. E., Hubbard W., Jackel L. D.* Handwritten digit recognition with a back-propagation network // *D. S. Touretzky* (Ed.). *Advances in neural information processing systems*, 2 (p. 396–404). Cambridge, MA: MIT Press, 1989.
149. *LeCun Y., Boser B., Denker J. S., Henderson D., Howard R. E., Hubbard W., Jackel L. D.* Backpropagation applied to handwritten zip code recognition. *Neural Computation*. 1989. V. 1 (4). P. 541–551.
150. *Ciresan D. C., Meier U., Masci J., Gambardella Maria L., Schmidhuber J.* Flexible, high performance convolutional neural networks for image classification // *Proceedings of the International Joint Conference on Artificial Intelligence* (vol. 1, p. 1237–1242). Menlo Park, CA: AAAI Press, 2011.
151. *Szegedy C., Liu W., Jia Y., Sermanet P., Reed S., Anguelov D., Rabinovich A.* Going deeper with convolutions // *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (p. 1–9). Los Alamitos, CA: IEEE Computer Society, 2015.
152. *Floreano D., Mattiussi C.* Bio-inspired artificial intelligence: Theories, methods, and technologies. Cambridge, MA: MIT Press, 2008.
153. *Goodfellow I., Bengio Y., Courville A.* Deep learning. Cambridge, MA: MIT Press, 2016.
154. *Hubel D. H., Wiesel T. N.* Receptive fields of single neurones in the cat's striate cortex // *Journal of Physiology*. 1959. V. 148 (1). P. 574–591.
155. *Hubel D. H., Wiesel T. N.* Receptive fields, binocular interaction and functional architecture in the cat's visual cortex, 1962.

156. *Fukushima K.* Self-organization of a neural network which gives position-invariant response // Proceedings of the 6th International Joint Conference on Artificial Intelligence (vol. 1, p. 291–293). San Francisco: Morgan Kaufmann, 1979.
157. *Fukushima K.* Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position // Biological Cybernetics. 1980. V. 36 (4). P. 193–202.
158. *Fukushima K., Miyake S.* Neocognitron: A new algorithm for pattern recognition tolerant of deformations and shifts in position // Pattern Recognition. 1982. V. 15 (6). P. 455–469.
159. *LeCun Y.* Generalization and network design strategies // R. Pfeifer, Z. Schreter, F. Fogelman, L. Steels (Eds). Connections in perspective (p. 143–155). Zurich, Switzerland: Elsevier, 1989.
160. *Dreyfus S.* The numerical solution of variational problems // Journal of Mathematical Analysis and Applications. 1962. V. 5 (1). P. 30–45.
161. *Linnainmaa S.* The representation of the cumulative rounding error of an algorithm as a Taylor expansion of the local rounding errors. Master's thesis, University of Helsinki, Finland, 1970.
162. *Werbos P.* Beyond regression: New tools for prediction and analysis in the behavioral sciences. Doctoral dissertation, Harvard University, 1974.
163. *Werbos P. J.* Applications of advances in nonlinear sensitivity analysis // R. F. Drenick, F. Kozin (Eds). System modeling and optimization (p. 762–770). Berlin: Springer, 1982.
164. *Rumelhart D. E., Hinton G. E., Williams R. J.* Learning representations by back-propagating errors // Nature. 1986. V. 323 (6088). P. 533–536.
165. *Simard P. Y., Steinkraus D., Platt J. C.* Best practices for convolutional neural networks applied to visual document analysis // Proceedings of the 7th International Conference on Document Analysis and Recognition (vol. 3, p. 958–963). Washington, DC: IEEE Computer Society, 2003. August.
166. *Decoste D., Schölkopf B.* Training invariant support vector machines // Machine Learning. 2002. V. 46 (1–3). P. 161–190.
167. *Chellapilla K., Shilman M., Simard P.* Optimally combining a cascade of classifiers // Proceedings of the 18th Annual Symposium on Electronic Imaging (p. 6067–6126). N.p.: International Society for Optical Engineering, 2006.
168. *Abdulkader A.* A two-tier Arabic offline handwriting recognition based on conditional joining rules // Proceedings of the 10th International

- al Workshop on Frontiers in Handwriting Recognition (p. 1–6). Los Alamitos, CA: IEEE Computer Society, 2006.
169. *Chellapilla K., Simard P.* Two-tier approach for Arabic offline handwriting recognition // Proceedings of the 10th International Workshop on Frontiers in Handwriting Recognition (p. 1–6). Los Alamitos, CA: IEEE Computer Society, 2006.
 170. *Ranzato M., Poultney C., Chopra S., LeCun Y.* Efficient learning of sparse representations with an energy-based model // P. B. Schölkopf, J. C. Platt, T. Hoffman (Eds). Advances in neural information processing systems, 19 (p. 1137–1144). Cambridge, MA: MIT Press, 2006.
 171. *Fei-Fei L., Fergus R., Perona P.* One-shot learning of object categories // IEEE Transactions on Pattern Analysis and Machine Intelligence. 2006. V. 28 (4). P. 594–611.
 172. *Zhang H., Berg A. C., Maire M., Malik J.* SVM-KNN: Discriminative nearest neighbor classification for visual category recognition // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (p. 2126–2136). Red Hook, N. Y.: Curran, 2006.
 173. *Weston J., Ratle F., Mobahi H., Collobert R.* Deep learning via semisupervised embedding // Proceedings of the 25th International Conference on Machine Learning (p. 1168–1175). N. p.: International Machine Learning Society, 2008.
 174. *Belkin M., Niyogi P., Sindhvani V.* Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. Journal of Machine Learning Research. 2006. V. 7. P. 2399–2434.
 175. *Collobert R., Sinz F., Weston J., Bottou L.* Large scale transductive SVMs // Journal of Machine Learning Research. 2006. V. 7. P. 1687–1712.
 176. *Oh K., Jung K.* GPU implementation of neural networks // Pattern Recognition. 2004. V. 37 (6). P. 1311–1314.
 177. *Steinkrau D., Simard P. Y., Buck I.* Using GPUs for machine learning algorithms // Proceedings of the 8th International Conference on Document Analysis and Recognition (p. 1115–1119). Washington, DC: IEEE Computer Society, 2005.
 178. *Scherer D., Müller A., Behnke S.* Evaluation of pooling operations in convolutional architectures for object recognition // Proceedings of the 20th International Conference on Artificial Neural Networks (p. 92–101). Berlin: Springer, 2010.
 179. *Boureau Y., Ponce J., LeCun Y.* A theoretical analysis of feature pooling in visual recognition // Proceedings of the 27th International Con-

- ference on Machine Learning (p. 111–118). N. p.: International Machine Learning Society, 2010.
180. *Jarrett K., Kavukcuoglu K., LeCun Y.* What is the best multi-stage architecture for object recognition? // Proceedings of the IEEE International Conference on Computer Vision (p. 2146–2153). Red Hook, N. Y.: Curran, 2009.
 181. *Ciresan D., Meier U., Schmidhuber J.* Multi-column deep neural networks for image classification // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (p. 3642–3649). Red Hook, N. Y.: Curran, 2012.
 182. *Srivastava N., Hinton G. E., Krizhevsky A., Sutskever I., Salakhutdinov R.* Dropout: A simple way to prevent neural networks from overfitting // Journal of Machine Learning Research. 2014. V. 15 (1). P. 1929–1958.
 183. *He K., Zhang X., Ren S., Sun J.* Delving deep into rectifiers: Surpassing human-level performance on imagenet classification // Proceedings of the IEEE International Conference on Computer Vision (p. 1026–1034). Red Hook, N. Y.: Curran, 2015.
 184. *Mallat S.* Group invariant scattering // Communications on Pure and Applied Mathematics. 2012. V. 65 (10). P. 1331–1398.
 185. *Wiatowski T., Bölcskei H.* A mathematical theory of deep convolutional neural networks for feature extraction. 2015. arXiv:1512.06293.
 186. *Girshick R., Donahue J., Darrell T., Malik J.* Rich feature hierarchies for accurate object detection and semantic segmentation // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (p. 580–587). Red Hook, N. Y.: Curran, 2014.
 187. *Yu W., Yang K., Bai Y., Yao H., Rui Y.* DNN flow: DNN feature pyramid based image matching // Proceedings of the British Machine Vision Conference (p. 1–10). Durham, UK: BMVA Press, 2014a.
 188. *Yu W., Yang K., Bai Y., Yao H., Rui Y.* Visualizing and comparing convolutional neural networks. 2014b. arXiv:1412.6631.
 189. *Lee H., Grosse R., Ranganath R., Ng A. Y.* Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations // Proceedings of the 26th International Conference Machine Learning (p. 609–616). N. p.: International Machine Learning Society, 2009.
 190. *Razavian A., Azizpour H., Sullivan J., Carlsson S.* CNN features off-the-shelf: An astounding baseline for recognition // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition

- Workshops (p. 806–813). Los Alamitos, CA: IEEE Computer Society, 2014.
191. *Lee C., Gallagher P. W., Tu Z.* Generalizing pooling functions in convolutional neural networks: Mixed, gated and tree // Proceedings of the 19th International Conference on Artificial Intelligence and Statistics (p. 464–472). 2016.
 192. *Laptev D., Savinov N., Buhmann J. M., Pollefeys M.* TI-Pooling: Transformation-invariant pooling for feature learning in convolutional neural networks. 2016. arXiv:1604.06318.
 193. *Jaderberg M., Simonyan K., Zisserman A.* Spatial transformer networks // C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, R. Garnett (Eds). Advances in neural information processing systems, 28 (p. 2017–2025). Red Hook, N. Y.: Curran, 2015.
 194. *Goodfellow I. J., Shlens J., Szegedy C.* Explaining and harnessing adversarial examples // Proceedings of the 3rd International Conference on Learning Representations (p. 1–11). N. p.: Computational and Biological Learning Society, 2015.
 195. *Uličný M., Lundström J., Bytner S.* Robustness of deep convolutional neural networks for image recognition // Proceedings of the 1st International Symposium on Intelligent Computing Systems (p. 16–30). Switzerland: Springer International Publishing, 2016.
 196. *Jin J., Dundar A., Culurciello E.* Robust convolutional neural networks under adversarial noise // Proceedings of the 4th International Conference on Learning Representations (p. 1–8). N. p.: Computational and Biological Learning Society, 2016.
 197. *Tabacof P., Valle E.* Exploring the space of adversarial images // Proceedings of the International Joint Conference on Neural Networks (p. 1–8). Red Hook, N. Y.: Curran, 2016.
 198. *Miyato T., Maeda S., Koyama M., Nakae K., Ishii S.* Distributional smoothing with virtual adversarial training // Proceedings of the 5th International Conference on Learning Representations (p. 1–12). N. p.: Computational and Biological Learning Society, 2016.
 199. *Sabour S., Cao Y., Faghri F., Fleet D. J.* Adversarial manipulation of deep representations // Proceedings of the 4th International Conference on Learning Representations (p. 1–18). N. p.: Computational and Biological Learning Society, 2016.
 200. *Papernot N., McDaniel P., Wu X., Jha S., Swami A.* Distillation as a defense to adversarial perturbations against deep neural networks // Proceedings of the 37th IEEE Symposium on Security and

- Privacy (p. 1–16). Los Alamitos, CA: IEEE Computer Society, 2016.
201. *Huang R., Xu B., Schuurmans D., Szepesvári C.* Learning with a strong adversary. 2016. arXiv:1511.03034v6.
 202. *Fawzi A., Fawzi O., Frossard P.* Analysis of classifiers' robustness to adversarial perturbations. 2015a. arXiv:1502.02590.
 203. *Fawzi A., Fawzi O., Frossard P.* Fundamental limits on adversarial robustness // Proceedings of the 32nd International Conference on Machine Learning (p. 1–7). N.p.: International Machine Learning Society. 2015b.
 204. *Bastani O., Ioannou Y., Lampropoulos L., Vytiniotis D., Nori A., Criminisi A.* Measuring neural net robustness with constraints // D. Lee, M. Sugiyama, U.V. Luxburg, I. Guyon, R. Garnett (Eds). Advances in neural information processing systems, 29 (p. 2613–2621). Red Hook, N.Y.: Curran, 2016.
 205. *Ning F., Delhomme D., LeCun Y., Piano F., Bottou L., Barbano P. E.* Toward automatic phenotyping of developing embryos from videos // IEEE Transactions on Image Processing. 2005. V. 14 (9). P. 1360–1371.
 206. *Wang J., Yang Y., Mao J., Huang Z., Huang C., Xu W.* CNN-RNN: A unified framework for multi-label image classification. 2016. arXiv: 1604.04573.
 207. *Vinyals O., Toshev A., Bengio S., Erhan D.* Show and tell: A neural image caption generator // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (p. 3156–3164). Red Hook, N.Y.: Curran, 2015.
 208. *Karpathy A., Fei-Fei L.* Deep visual-semantic alignments for generating image descriptions // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (p. 3128–3137). Red Hook, N.Y.: Curran, 2016.
 209. *Ba J., Mnih V., Kavukcuoglu K.* Multiple object recognition with visual attention // Proceedings of the 3rd International Conference on Learning Representations (p. 1–10). N.p.: Computational and Biological Learning Society, 2015.
 210. *Goodfellow I., Pouget-Abadie J., Mirza M., Xu B., Warde-Farley D., Ozair S., Bengio Y.* Generative adversarial nets // Z. Ghahramani, M. Welling, C. Cortes, N.D. Lawrence, K. Q. Weinberger (Eds). Advances in neural information processing systems, 27 (p. 2672–2680). Red Hook, N.Y.: Curran, 2014.

211. *Kingma D. P., Welling M.* Auto-encoding variational Bayes, 2014. arXiv:1312.6114v10.
212. *Bengio Y., Thibodeau-Laufer E., Alain G., Yosinski J.* Deep generative stochastic networks trainable by backprop // Proceedings of the 31st International Conference Machine Learning (p. 226–234). N. p.: International Machine Learning Society, 2014.
213. *Kulkarni T. D., Whitney W. F., Kohli P., Tenenbaum J.* Deep convolutional inverse graphics network // C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, R. Garnett (Eds). Advances in neural information processing systems, 28 (p. 2539–2547). Red Hook, N. Y.: Curran, 2015.
214. *Bachman P.* An architecture for deep, hierarchical generative models // D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, R. Garnett (Eds). Advances in neural information processing systems, 29 (p. 4826–4834). Red Hook, N. Y.: Curran, 2016.
215. *Snoek J., Larochelle H., Adams R. P.* Practical Bayesian optimization of machine learning algorithms // F. Pereira, C. J. C. Burges, L. Bottou, K. Q. Weinberger (Eds). Advances in neural information processing systems, 25 (p. 2951–2959). Red Hook, N. Y.: Curran, 2012.
216. *Kennedy J., Eberhart R. C.* A new optimizer using particle swarm theory // Proceedings of the 6th International Symposium on Micro Machine and Human Science (p. 39–43). Piscataway, N. J.: IEEE. 1995. October.
217. *Pinto L., Gupta A.* Supersizing self-supervision: Learning to grasp from 50k tries and 700 robot hours. 2015. arXiv:1509.06825.
218. *Finn C., Tan X. Y., Duan Y., Darrell T., Levine S., Abbeel P.* Deep spatial autoencoders for visuomotor learning. 2015. arXiv:1509.06113.
219. *Levine S., Finn C., Darrell T., Abbeel P.* End-to-end training of deep visuomotor policies // Journal of Machine Learning Research. 2016. V. 17 (39). P. 1–40.
220. *Chollet F.* Xception: Deep learning with depthwise separable convolutions. 2016. arXiv:1610.02357.
221. *Xie S., Girshick R., Dollár P., Tu Z., He K.* Aggregated residual transformations for deep neural networks. 2016. arXiv:1611.05431.
222. *Goodfellow I. J., Pouget-Abadie J., Mirza M., Xu B., Warde-Farley D., Ozair Sh., Courville A., Bengio Y.* Generative Adversarial Nets. 2014. arXiv:1406.2661.
223. *Luc P., Couprie C., Chintala S., Verbeek J.* Semantic Segmentation using Adversarial Networks. 2016. arXiv:1611.08408.

224. *Palazzo S., Spampinato C., Kavasidis I., Giordano D., Shah M.* Generative Adversarial Networks Conditioned by Brain Signals, 2017. URL: http://openaccess.thecvf.com/content_ICCV_2017/papers/Palazzo_Generative_Adversarial_Networks_ICCV_2017_paper.pdf (дата обращения: 21.10.2018).
225. *Arakaki T., Barello G., Ahmadian Y.* Capturing the diversity of biological tuning curves using generative adversarial networks, 2017. doi: 10.1101/167916 (дата обращения: 21.10.2018).
226. *Che Z., Cheng Y., Zhai S., Sun Z., Liu Y.* Boosting Deep Learning Risk Prediction with Generative Adversarial Networks for Electronic Health Records // IEEE International Conference on Data Mining (ICDM), 2017. Piscataway, N. J.: IEEE, 2017. P. 787–792.
227. *Erickson Z., Chernova S., Kemp C. C.* Semi-Supervised Haptic Material Recognition for Robots using Generative Adversarial Networks. 2017. arXiv:1707.02796.
228. *Esteban C., Hyland S. L., Rätsch G.* Real-valued (Medical) Time Series Generation with Recurrent Conditional GANs. 2017. arXiv:1706.02633.
229. *Chen X., Duan Y., Houthoofd R., Schulman J., Sutskever I., Abbeel P.* InfoGAN: Interpretable Representation Learning by Information Maximizing Generative Adversarial Nets // Advances in Neural Information Processing Systems. 2016. arXiv:1606.03657.
230. *Hinton G. E., Salakhutdinov R. R.* Reducing the dimensionality of data with neural networks // Science. 2006. V. 313. № 5786. P. 504–507.
231. *Mescheder L., Nowozin S., Geiger A.* Adversarial Variational Bayes: Unifying Variational Autoencoders and Generative Adversarial Networks. 2017. arXiv:1701.04722.
232. *Dumoulin V., Belghazi I., Poole B., Mastropietro O., Lamb A., Arjovsky M., Courville A.* Adversarially Learned Inference. 2016. arXiv:1606.00704.
233. *Donahue J., Krähenbühl P., Darrell T.* Adversarial Feature Learning. 2016. arXiv:1605.09782.
234. *Li C., Liu H., Chen C., Pu Y., Chen L., Heno R., Carin L.* ALICE: Towards Understanding Adversarial Learning for Joint Distribution Matching // Advances in Neural Information Processing Systems. 2017. arXiv:1709.01215.
235. *Reed S., Akata Z., Yan X., Logeswaran L., Schiele B., Lee H.* Generative adversarial text to image synthesis. 2016. arXiv:1605.05396.
236. *Isola P., Zhu J.-Y., Zhou T., Efros A. A.* Image-to-Image Translation with Conditional Adversarial Networks. 2016. arXiv:1611.07004.

237. *Ledig C., Theis L., Huszar F., Caballero J., Cunningham A., Acosta A., Aitken A., Tejani A., Totz J., Wang Z., Shi W.* Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network. 2016. arXiv:1609.04802
238. *Ren S., He K., Girshick R., Sun J.* Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks // *Advances in Neural Information Processing Systems*. 2015. arXiv:1506.01497.
239. *Karras T., Aila T., Laine S., Lehtinen J.* Progressive Growing of GANs for Improved Quality, Stability and Variation. 2017. arXiv:1710.10196.
240. *Arora S., Ge R., Liang Y., Ma T., Zhang Y.* Generalization and Equilibrium in Generative Adversarial Nets (GANs). 2017. arXiv:1703.00573.
241. *Arjovsky M., Bottou L.* Towards Principled Methods for Training Generative Adversarial Networks. 2017. arXiv:1701.04862.
242. *Lee J. D., Simchowitz M., Jordan M. I., Recht B.* Gradient Descent Converges to Minimizers. 2016. arXiv:1602.04915.
243. *Creswell A., White T., Dumoulin V., Arulkumaran K., Sengupta B., Bharath A. A.* Generative Adversarial Networks: An Overview // *IEEE Signal Processing Magazine*. 2018. V. 35. № 1. P. 53–65.
244. *Jost Z.* Overview of GANs (Generative Adversarial Networks). Part I, 2017. URL: <https://www.kdnuggets.com/2017/11/overview-gans-generative-adversarial-networks-part1.html> (дата обращения: 21.10.2018).
245. *Theis L., van den Oord A., Bethge M.* A note on the evaluation of generative modelsю 2015. arXiv:1511.01844.
246. *Radford A., Metz L., Chintala S.* Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks 2015. arXiv:1511.06434.
247. *Mirza M., Osindero S.* Conditional Generative Adversarial Nets. 2014. arXiv:1411.1784.
248. *Sutton R. S., Barto A. G.* Reinforcement Learning: An Introduction. Second Edition. Cambridge, USA: MIT Press, 2018.
249. *Moody J. E., Saffell M.* Reinforcement Learning for Trading // *Advances in Neural Information Processing Systems (NIPS)*. Cambridge: MIT Press, 1998. P. 917–923.
250. *Shen Y.* Risk-averse Reinforcement Learning for Algorithmic Trading // *IEEE Conference on Computational Intelligence for Financial Engineering & Economics (CIFER)*. Piscataway, N.J.: IEEE, 2014. P. 391–398.
251. *Dempster M. A. H., Leemans V.* An Automated FX Trading System Using Adaptive Reinforcement Learning // *Expert Systems with*

- Applications: Special Issue on Financial Engineering. 2006. V. 30. P. 543–552.
252. *Deng Y.* Deep Direct Reinforcement Learning for Financial Signal Representation and Trading // IEEE Transactions on Neural Networks and Learning Systems. 2007. V. 28. № 3. P. 653–664.
253. *Cumming J.* An Investigation into the Use of Reinforcement Learning Techniques within the Algorithmic Trading Domain: M. Eng. Thesis. London: Imperial College, 2015.
254. *Si W.* A Multi-objective Deep Reinforcement Learning Approach for Stock Index Future's Intraday Trading // 10th International Symposium on Computational Intelligence and Design (ISCID). Vol. 2. Piscataway, N.J.: IEEE, 2017. P. 431–436.
255. *Huang Ch. Y.* Financial Trading as a Game: A Deep Reinforcement Learning Approach. 2018. arXiv:1807.02787.
256. *Liang Zh., Jiang K., Chen H., Zhu J., Li Y.* Deep Reinforcement Learning in Portfolio Management. 2018. arXiv:1808.09940.
257. *Spooner Th., Fearnley J., Savani R., Koukorinis A.* Market Making via Reinforcement Learning. 2018. arXiv:1804.04216.
258. *Gao X.* Deep Reinforcement Learning for Time Series: Playing Idealized Trading Games. 2018. arXiv:1803.03916.
259. *Singh S. P., Barto A. G., Chentanez N.* Intrinsically Motivated Reinforcement Learning // Advances in Neural Information Processing Systems (NIPS). Cambridge: MIT Press, 2004. P. 1281–1288.
260. *Pavlov M., Kolesnikov S., Plis S. M.* Run, Skeleton, Run: Skeletal Model in a Physics-based Simulation. 2018. arXiv:1711.06922.
261. *Mnih V., Kavukcuoglu K., Silver D., Graves A., Antonoglou I., Wierstra D., Riedmiller M.* Playing Atari with Deep Reinforcement Learning. 2013. arXiv:1312.5602.
262. *Mittel A., Munukutla S., Yadav H.* Visual Transfer between Atari Games using Competitive Reinforcement Learning. 2018. arXiv:1809.00397.
263. *Plappert M., Andrychowicz M., Ray A., McGrew B., Baker B., Powell G., Schneider J., Tobin J., Chociej M., Welinder P., Kumar V., Zaremba W.* Multi-Goal Reinforcement Learning: Challenging Robotics Environments and Request for Research. 2018. arXiv:1802.09464.
264. *Van Otterlo M., Wiering M.* Reinforcement Learning and Markov Decision Processes // Reinforcement Learning. Adaptation, Learning and Optimization. Berlin: Springer, 2012. V. 12. P. 3–42.
265. *Curran W.* Dimensionality Reduced Reinforcement Learning for Assistive Robots // AAAI Fall Symposium on Artificial Intelligence 2016.

- Symposium on Artificial Intelligence for Human–Robot Interaction (Arlington, November 17–19, 2016). Arlington, USA: AAAI, 2016. P. 25–31.
266. *Hsu Sh.-Hs., Shen I-Ch., Chen B.-Yu.* Transferring Deep Reinforcement Learning with Adversarial Objective and Augmentation. 2018. arXiv:1809.00770.
267. *Busoniu L., Babuska R., de Schutter B., Ernst D.* Reinforcement Learning and Dynamic Programming Using Function Approximators. Boca Raton, USA: CRC Press, 2010.
268. *Watkins Ch. J. C. H., Dayan P.* Q-learning // Machine Learning. 1992. V. 8. № 3–4. P. 279–292.
269. *Mnih V., Kavukcuoglu K., Silver D., Rusu A. A., Veness J., Bellemare M. G., Graves A., Riedmiller M., Fidjeland A. K., Ostrovski G., Petersen S., Beattie Ch., Sadik A., Antonoglou I., King H., Kumaran Dh., Wierstra D., Legg Sh., Hassabis D.* Human-level Control through Deep Reinforcement Learning // Nature. 2015. V. 518. P. 529–533.
270. *Van Hasselt H., Guez A., Silver D.* Deep Reinforcement Learning with Double Q-Learning // Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence (AAAI-16). Arlington, USA: AAAI, 2016. P. 2094–2100.
271. *Schaul T., Quan J., Antonoglou I., Silver D.* Prioritized Experience Replay // International Conference on Learning Representations (ICLR). San Juan, Puerto Rico. 2016. May 2–4. arXiv:1511.05952.
272. *Wang Z., Schaul T., Hessel M., van Hasselt H., Lanctot M., de Freitas N.* Dueling Network Architectures for Deep Reinforcement Learning // Proceedings of The 33rd International Conference on Machine Learning (ICML). N. Y., 2016. June 19–24. arXiv:1511.06581.
273. *Fortunato M., Azar M. Gh., Piot B., Menick J., Osband I., Graves A., Mnih V., Munos R., Hassabis D., Pietquin O., Blundell Ch., Legg Sh.* Noisy Networks for Exploration // International Conference on Learning Representations (ICLR). Vancouver, 2018. April 30–May 3. arXiv:1706.10295.
274. *Bellemare M. G., Dabney W., Munos R.* A Distributional Perspective on Reinforcement Learning // Proceedings of the 34th International Conference on Machine Learning. 2017. arXiv:1707.06887.
275. *Hessel M., Modayil J., van Hasselt H., Schaul T., Ostrovski G., Dabney W., Horgan D., Piot B., Azar M., Silver D.* Rainbow: Combining Improvements in Deep Reinforcement Learning // Proceedings of the 31nd AAAI Conference on Artificial Intelligence (AAAI-18). New Orleans, 2018. February 2–7. arXiv:1710.02298.

276. *Biondo A. E., Pluchino A., Rapisarda A., Helbing D.* Are Random Trading Strategies More Successful than Technical Ones? 2013. arXiv:1303.4351,
277. *Nanda S., Peters D. J.* A Very Long-Term Buy-and-Hold Portfolio // The Journal of Portfolio Management. 2006. V. 32. № 3. P. 65–73.
278. *Shiryaev A., Xu Z., Zhou X. Yu.* Thou Shalt Buy and Hold // Quantitative Finance. 2008. V. 8. № 8. P. 765–776.
279. Digital Intelligence: The Heart of Successful Digital Transformation. Daniel Newman – Futurum, 2017.
280. *Коровкин В., Плаксенков Е., Кабакова О.* Цифровые платформы и экосистемы финансовой инклюзивности. Российский опыт. Отчет. Московская школа управления Сколково, 2015.
281. *Паркер Джеффри, Альтин М. ван, Санджит Ч.* Революция платформ: Как сетевые рынки меняют экономику – и как заставить их работать на вас / Пер. с англ. Е. Пономаревой. М.: Манн, Иванов и Фарберов, 2017.
282. *Will Chu.* Carrefour extend Blockchain use to dairy and meat product ranges 08-Mar-2018-URL: https://www.foodnavigator.com/News/Business/Carrefour-extend-Blockchain-use-to-dairy-and-meat-product-ranges?utm_source=copyright&utm_medium=OnSite&utm_campaign=copyright#.
283. Chainstep. Blockchain in use. URL: <https://www.chainstep.com/use-cases/?lang=en>.
284. Nir Kshetri Blockchain's roles in meeting key supply chain management objectives, International Journal of Information Management. 2018. V. 39. P. 80–89.
285. *Корчагин С.* О текущих трендах в развитии технологии блокчейн // Свободная мысль. 2016. № 4 (1658). С. 31–38.
286. *Труфанов С. А.* Будущее менеджмента, маркетинга и производства в контексте развития информационных технологий и эволюции поколений // Экономика. Управление. Финансы. 2017. № 3 (9). С. 45–56.
287. *Нигматулин Т. А., Краснова А. И., Лавринович А. А.* Перспективы использования технологии блокчейн таможенными органами Российской Федерации // Ученые записки Санкт-Петербургского им. В. Б. Бобкова Филиала Российской таможенной академии. 2016. № 4 (60). С. 11–14.
288. Информационная группа Интерфакс. В России с середины 2018 г. введут систему поштучного учета алкоголя. URL: <http://www.interfax.ru/business/591657> (дата обращения: 13 декабря 2017).

289. DHL. Trend Research. Sharing Economy logistics. Rethinking logistics with access over ownership. 2017. May. URL: http://www.dhl.com/content/dam/downloads/g0/about_us/logistics_insights/DHL-Trend_Report_Sharing_Economy.pdf.
290. Новоселова Л. «Токенизация» объектов гражданского права // Хозяйство и право. 2017. № 12. С. 29–44.
291. Larson R., Richards M. Boredom in the middle school years: Blaming schools versus blaming students // American Journal of Education. 1991. V. 99. P. 418–443.
292. Dunleavy J., Milton P. What did you do in school today? Exploring the concept of student engagement and its implications for teaching and learning in Canada // Canadian Education Association. 2009. P. 1–22.
293. Anderson J. R. Acquisition of cognitive skill // Psychological Review. 1982. V. 89. № 4. P. 369–406.
294. Mostow J., Hauptmann A., Chase L., Roth S. Towards a reading coach that listens: Automated detection of oral reading errors // Proceedings of the 11th National Conference on Artificial Intelligence (AAAI 1993), American Association for Artificial Intelligence. Palo Alto, USA: AAAI Press, 1993. P. 392–397.
295. Koedinger K. R., Anderson J. R. Intelligent tutoring goes to school in the big city // International Journal of Artificial Intelligence in Education. 1997. V. 8. P. 30–43.
296. VanLehn K., Lynch C., Schultz K., Shapiro J., Shelby R., Taylor L. The Andes physics tutoring system: Lessons learned // International Journal of Artificial Intelligence in Education. 2005. V. 15. № 3. P. 147–204.
297. Fredricks J. A., Blumenfeld P. C., Paris A. H. School engagement: Potential of the concept, state of the evidence // Review of Educational Research. 2004. V. 74. № 1. P. 59–109.
298. Fredricks J. A., McColskey W. The measurement of student engagement: A comparative analysis of various methods and student self-report instruments // Handbook of Research on Student Engagement. N. Y.: Springer, 2012. P. 763–782.
299. Anderson A., Christenson S., Sinclair M., Lehr C. Check and connect: The importance of relationships for promoting engagement with school // Journal of School Psychology. 2004. V. 42. P. 95–113.
300. Harris L. A phenomenographic investigation of teacher conceptions of student engagement in learning // Australian Educational Researcher. 2008. V. 5. № 1. P. 57–79.

Литература

301. *Малошонок Н. Г.* Вовлеченность студентов в учебный процесс в российских вузах // Высшее образование в России. 2014. № 1. С. 37–44.
302. *Beck J.* Engagement tracing: Using response times to model student disengagement // Proceedings of the 2005 Conference on Artificial Intelligence in Education: Supporting Learning through Intelligent and Socially Informed Technology. Amsterdam: IOS Press, 2005. P. 88–95.
303. *Johns J., Woolf B.* A dynamic mixture model to detect student motivation and proficiency // Proceedings of the 21st National Conference on Artificial Intelligence (AAAI 2006), American Association for Artificial Intelligence. Palo Alto, USA: AAAI Press, 2006. P. 2–8.
304. *Pope A., Bogart E., Bartolome D.* Biocybernetic system evaluates indices of operator engagement in automated task // Biological Psychology. 1995. V. 40. P. 187–195.
305. *Fairclough S., Venables L.* Prediction of subjective states from psychophysiology: A multivariate approach // Biological Psychology. 2006. V. 71. P. 100–110.
306. *Kapoor A., Picard R.* Multimodal affect recognition in learning environments // Proceedings of the 13th Annual ACM International Conference on Multimedia (MULTIMEDIA 2005). N. Y.: ACM, 2005. P. 677–682.
307. *McDaniel B., D’Mello S., King B., Chipman P., Tapp K., Grässer A.* Facial features for affective state detection in learning environments // Proceedings of the 29th Annual Conference of the Cognitive Science Society. Austin: Cognitive Science Society, 2007. P. 467–472.
308. *D’Mello S., Craig S., Grässer A.* Multimethod assessment of affective experience and expression during deep learning // International Journal of Learning Technology. 2009. V. 4. № 3. P. 165–187
309. *D’Mello S., Graesser A.* Multimodal semi-automated affect detection from conversational cues, gross body language and facial features // User Modeling and User-Adapted Interaction. 2010. V. 20. № 2. P. 147–187.
310. *Grafsgaard J., Fulton R., Boyer K., Wiebe E., Lester J.* Multimodal analysis of the implicit affective channel in computer-mediated textual communication // Proceedings of the 14th ACM international conference on Multimodal interaction (ICMI 2012). N. Y.: ACM, 2012. P. 145–152.
311. *Whitehill J., Serpell Z., Lin Yi-Ch., Foster A., Movellan J. R.* The faces of engagement: Automatic recognition of student engagement from fa-

Литература

- cial expressions // IEEE Transactions on Affective Computing. 2014. V. 5. № 1. P. 86–98.
312. How to Detect Faces in Image // MicrosoftDocs on GitHub.
313. *Соловьев В., Куклина Д., Славгородский А., Пухов И., Тутко М.* Мониторинг вовлеченности студентов в учебный процесс // Открытые системы. СУБД. 2018. № 2. С. 28–32.
314. *Бондар И. А.* Взаимодействие формальных и неформальных институтов в процессе трансформации экономических отношений // Вестник Саратовского государственного социально-экономического университета. 2010. № 5. URL: <https://cyberleninka.ru/article/n/vzaimodeystvie-formalnyh-i-neformalnyh-institutov-v-protseste-transformatsii-ekonomicheskikh-otnosheniy> (дата обращения: 21.12.2018).
315. *Дуглас Н.* Институты, институциональные изменения и функционирование экономики. М.: Фонд экономической книги «Начала», 1997. Т. 2.
316. *Натхов Т.* Классификация институтов и социальный контракт. 2007. URL: <http://www.hse.ru/data/785/069/1236/Text.pdf>.
317. *Аузан А. А.* (ред.). Институциональная экономика: новая институциональная экономическая теория: Учебник. М.: Инфра-М, 2005.
318. *Тамбовцев В. Л.* Институты-как-равновесия vs институты-как-правила // Журнал экономической теории. 2013. № 4. С. 111–122.
319. *Варнавский А. В.* Токен/криптовалюта: технологическое содержание и экономическая сущность // Финансы: теория и практика. 2018. № 5.
320. *Бутрюмова Н. Н., Назаров М. Г., Фияксель Э. А.* Законодательные барьеры ведения инновационной деятельности в России // Инновации. 2016. № 4 (210). URL: <https://cyberleninka.ru/article/n/zakonodatelnye-bariery-vedeniya-innovatsionnoy-deyatelnosti-v-rossii> (дата обращения: 21.12.2018).
321. *Нисан Н., Ронен А.* Алгоритмический механизм проектирования // Игры и экономическое поведение. 2001. Т. 35. № 1–2. С. 166–196.
322. Правовая информация / Отв. ред. А. Ф. Шебанов, А. Р. Шляхов, С. С. Москвин. М.: Наука, 1974.
323. *Жуков В. И.* Правовая информация: ее сущность и типология // Проблема совершенствования советского законодательства / Отв. ред. И. Н. Кузнецов. М.: Изд-во ВНИИ, 1985.

324. *Иванов И. Г.* Принципы построения формально-алгоритмических моделей нормативно-правовых документов // Известия ЮФУ. Технические науки. 1998. № 1.
325. *Иванов И. Г.* Принципы формирования множества субъектов и объектов правоотношений при процедурном моделировании регулятивных функций // Известия ЮФУ. Технические науки. 2006. № 9–2. URL: <https://cyberleninka.ru/article/n/principy-formirovaniya-mnozhestva-subektov-i-obektov-pravootnosheniya-pri-protsedurnom-modelirovanii-regulyativnyh-funktsiy> (дата обращения: 12.05.2018).
326. *Мазуренко А. П.* К вопросу о методах правотворческой политики // ЮП. 2007. № 5. URL: <https://cyberleninka.ru/article/n/k-voprosu-o-metodah-pravotvorcheskoj-politiki> (дата обращения: 06.05.2018).
327. *Гаврилов О. А.* Информатизация правовой системы России. М., 1998.
328. *Bodard F., Hella M., Pouillet Y., Stenne P.* A Prototype ADP System to Assist Judicial Decision Making. URL: <http://www.crid.be/pdf/public/5817.pdf>.
329. *Beckstein C., Stolle R., Tobermann G.* Meta-Programming for Generalized Horn Clause Logic. URL: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.47.8538&rep=rep1&type=pdf>.
330. *Thomasset C., Paquin L. C.* Expert Systems in Law and the Representation of Legal Knowledge: Can we isolate it from the Why and the Who? URL: http://multimedia.uqam.ca/profs/kdz/FTPRoot/web-server/profs/lcp/publi_pdf/Florence_89%20.pdf.
331. *Meyer B.* Object-oriented software construction. V. 2. N. Y.: Prentice Hall, 1988.
332. *Szabo N.* Smart contracts: building blocks for digital markets // EXTROPY: The Journal of Transhumanist Thought. 1996.16.
333. *Godlevskiy A., Letichevskiy A., Peschanenko V., Letychevskiy O., Morokhovets M., Skobelev V., Poltorackiy M.* Formalization and Algebraic Verification of Legal Requirements. URL: <http://ceur-ws.org/Vol-1844/10000524.pdf>.
334. Официальный сайт Wikimedia Foundation. URL: <https://wikimediafoundation.org/wiki/Home>.
335. Логическая и визуальная разметки. URL: https://ru.wikipedia.org/wiki/Язык_разметки#Логическая_и_визуальная_разметки.
336. Вики-разметка. URL: <http://wiki.risk.ru/index.php/Вики-разметка>.

Литература

337. Wikitext, Links_and_URLs. URL: https://en.wikipedia.org/wiki/Help:Wikitext#Links_and_URLs.
338. Правила и указания Википедии. URL: https://ru.wikipedia.org/wiki/Википедия:Правила_и_указания.
339. Top Ten Wiki Engines. URL: <http://wiki.c2.com/?TopTenWikiEngines>.
340. Официальный сайт WikiMatrix. URL: <http://www.wikimatrix.org>.
341. Comparison of wiki software, Wikipedia. URL: https://en.wikipedia.org/wiki/Comparison_of_wiki_software.
342. Алтайский государственный технический университет. «База знаний». URL: http://wiki.mvtom.ru/index.php/База_знаний.
343. Официальный сайт Atlassian Confluence. URL: <https://ru.atlassian.com/software/confluence>.
344. Почта банк создал «Базу знаний». 21.04.2017. URL: <https://www.pochtabank.ru/news/423979>.
345. Знание – сила: как Альфа-банк минимизировал риски, создав корпоративную википедию. 02.03.2016. URL: <https://alfabank.ru/press/monitoring/2016/3/2/35499.html>.
346. List of the different language editions of Wikipedia, Wikipedia. URL: https://en.wikipedia.org/wiki/List_of_Wikipedias#Detailed_list.
347. The fundamental principles of Wikipedia, Wikipedia. URL: https://en.wikipedia.org/wiki/Wikipedia:Five_pillars.
348. Wolfram|Alfa search engine. URL: <https://www.wolframalpha.com/input/?i=wikipedia.org>.
349. Pageviews Analysis tool for Wikimedia Foundation wikis. URL: <https://tools.wmflabs.org/siteviews>.
350. Награды Русской Википедии. URL: https://ru.wikipedia.org/wiki/Русская_Википедия#Награды.
351. Wikimania. URL: <https://wikimania2017.wikimedia.org/wiki/Wikimania>.
352. Meta-Wiki. URL: https://meta.wikimedia.org/wiki/Main_Page.
353. Media-Wiki. URL: <https://www.mediawiki.org/wiki/MediaWiki>.
354. Общие минимальные требования Википедии, Википедия, URL: https://ru.wikipedia.org/wiki/Википедия:Минимальные_требования#Общие_требования.
355. Каталог шаблонов Википедии, Википедия. URL: https://ru.wikipedia.org/wiki/Википедия:Шаблоны#Каталог_шаблонов.
356. Alexa. The top 500 sites on the web. URL: <https://www.alexa.com/topsites>.

357. *Ярцева В. Н.* Лингвистический энциклопедический словарь. М.: Сов. энциклопедия, 1990.
358. *Gunter C. A., Scott D. S.* Semantic Domains // Handbook on theoretical computer science. Vol. B: Formal models and semantics (B). 1990. P. 633–674.
359. *Berners-Lee T.* Semantic Web Road map. 1998. 14.10. URL: <https://www.w3.org/DesignIssues/Semantic.html>.
360. *Berners-Lee T., Hendler J., Lassila O.* The Semantic Web // Scientific American. 2001. May. P. 29–37.
361. Resource Description Framework. 25.02.2014. URL: <https://www.w3.org/RDF>.
362. *Manola F., Miller E.* RDF Primer. W3C, 10.02.2004. URL: <https://www.w3.org/TR/2004/REC-rdf-primer-20040210>.
363. *Cyganiak R., Wood D., Lanthaler M.* RDF Primer, 25.02.2014. URL: <https://www.w3.org/TR/2014/REC-rdf11-concepts-20140225>.
364. Web Ontology Language. 27.10.2009. URL: <https://www.w3.org/OWL>.
365. *Martin D., Hodgson R., Horrocks I., Yendluri P.* OWL 2 Web Ontology Language. Document Overview (Second Edition). 11.12.2012. URL: <https://www.w3.org/TR/2012/REC-owl2-overview-20121211>.
366. *Hitzler P., Krötzsch M., Parsia B.* OWL 2 Web Ontology Language. Primer (Second Edition). 11.12.2012. URL: <https://www.w3.org/TR/2012/REC-owl2-primer-20121211>.
367. *Smith M. K., Welty Ch., McGuinness D. L.* OWL Web Ontology Language Guide, 10.02.2004. URL: <https://www.w3.org/TR/2004/REC-owl-guide-20040210>.
368. *Harris S., Seaborne A.* SPARQL 1.1 Query Language. 21.03.2013. URL: <https://www.w3.org/TR/2013/REC-sparql11-query-20130321>.
369. DCMi Metadata Terms. DCMi Usage Board. 14.06.2012. URL: <http://www.dublincore.org/documents/2012/06/14/dcmi-terms>.
370. Metadata Basics. DCMi Usage Board. URL: <http://dublincore.org/metadata-basics>.
371. Dublin Core Metadata Element Set, Version 1.1: Reference Description. DCMi Usage Board, 14.06.2012. URL: <http://dublincore.org/documents/2012/06/14/dces>.
372. What is the Open Source Metadata Framework? URL: <http://ibiblio.org/osrt/omf>.
373. PBCore: The Challenge of Adopting a Descriptive Metadata Standard for Public Media, 16.02.2011. URL: <http://pbcore.org/tag/dublin-core>.

374. Open Packaging Format (OPF) 2.0.1 v1.0.1. URL: http://www.idpf.org/epub/20/spec/OPF_2.0_latest.htm.
375. A unified use of Metadata in eXo Platform. Brice Revenant. 27.12.2008. URL: <https://www.exoplatform.com/blog/2008/12/27/a-unified-use-of-metadata-in-exo-platform>.
376. XML Watch: Finding friends with XML and RDF. Edd Dumbill. 01.06.2002. URL: <https://web.archive.org/web/20091223003446/http://www.ibm.com/developerworks/xml/library/x-foaf.html>.
377. XML Watch: Support online communities with FOAF. 01.08.2002. URL: <https://web.archive.org/web/20100307223814/http://www.ibm.com/developerworks/xml/library/x-foaf2.html>.
378. DataSources. URL: <https://web.archive.org/web/20100226072731/http://wiki.foaf-project.org/w/DataSources>.
379. Yandex Search Supports World Standards. URL: https://yandex.com/company/press_center/press_releases/2008/2008-08-15.
380. FOAF. James Walker. 2014. 19.10. URL: <https://www.drupal.org/project/foaf>.
381. Sir Tim Berners-Lee. Talks with Talis about the Semantic Web (transcript of an interview recorded on 7 February 2008). Paul Miller. 07.02.2008. URL: https://web.archive.org/web/20130510134842/http://talis-podcasts.s3.amazonaws.com/twt20080207_TimBL.html.
382. DBpedia Version 2014 released. Chris Bizer. 2014. 09.09. URL: <http://blog.dbpedia.org/2014/09/09/dbpedia-version-2014-released>.
383. AlchemyAPI. URL: <http://wiki.dbpedia.org/projects/alchemyapi>.
384. Natural Language Understanding. 2017. 18.12. URL: <https://console.bluemix.net/docs/services/natural-language-understanding/getting-started.html>.
385. Yahoo! Content Analysis API. URL: <http://wiki.dbpedia.org/projects/yahoo-content-analysis-api>.
386. Qeupy: python to do question answering. Elías Andrawos. 2013. 20.09. URL: <http://www.machinalis.com/blog/qeupy>.
387. Искусственный интеллект в поиске. Как Яндекс научился применять нейронные сети, чтобы искать по смыслу, а не по словам. Блог компании Яндекс. 2016. 02.11. URL: <https://habrahabr.ru/company/yandex/blog/314222>.
388. Как работает нейронный машинный перевод? Блог компании Lokalise. 2017. 28.07. URL: <https://habrahabr.ru/company/lokalise/blog/334342>.

389. A Semantic Bank. Dave McComb. URL: <https://semanticarts.com/blog/a-semantic-bank>.
390. The EDM Council. URL: <http://www.edmcouncil.org>.
391. The Semantic Bank with FIBO. Shannon Walker, Enterprise Data World 2016 Conference. 2016. 20.04. URL: <http://edw2016.dataversity.net/sessionPop.cfm?confid=94&proposalid=8976>.
392. Banking on FIBO: Financial Institutions Turn to Semantic Standard. Jennifer Zaino, Enterprise Data World. 2016 Conference, 06.10.2016. URL: <http://www.dataversity.net/banking-fibo-financial-institutions-turn-standard-value-compliance>.
393. *Пайгородский А. М.* Модели случайных графов. М.: МЦНМО, 2011.
394. *Page L., Brin S., Motwani R., Winograd T.* The PageRank Citation Ranking: Bringing Order to the Web. Technical report, Stanford Digital Library Technologies Project, 1998.
395. *Albert R., Jeong H., Barabasi L.-A.* Diameter of the World-Wide Web // Nature. 1999. September. V. 401. P. 130–131.
396. *Barabasi L.-A., Albert R.* Emergence of scaling in random networks // Science. 1999. October. V. 286. P. 509–512.
397. *Barabasi L.-A., Albert R., Jeong H.* Scale-free characteristics of random networks: the topology of the world-wide web // Physica. 2000. V. A281. P. 69–77.
398. NetworkX – Software for complex networks. URL: <https://networkx.github.io>.
399. Igraph – The network analysis package. URL: <http://igraph.org>.
400. Paul Weinberg James Groff Andrew Oppel. SQL. The Complete Reference, Third Edition, 2010.
401. Property Graph Model. URL: <https://github.com/opencypher/openCypher/blob/master/docs/property-graph-model.adoc#pgm-definitions-node>.
402. Property Graphs: The Swiss Army Knife of Data Modeling. URL: <http://www.dataversity.net/property-graphs-swiss-army-knife-data-modeling>.
403. *Robinson I., Webber J., Eifrem E.* Graph Databases. Neo Technology Inc., 2015.
404. Database schema of MediaWiki. URL: https://upload.wikimedia.org/wikipedia/commons/9/94/MediaWiki_1.28.0_database_schema.svg.
405. The Definitive Guide to Graph Databases for the RDBMS Developer. Neo4J, 2016.

Литература

406. Wikimedia Downloads. URL: <https://dumps.wikimedia.org/backup-index.html>.
407. *Watts D. J., Strogatz S.* Collective dynamics of ‘small-world’ networks // *Nature*. 1998. June. V. 393. P. 440–442.
408. *Велихов П. Е.* Меры семантической близости статей Википедии и их применение к обработке текстов // *Информационные технологии и вычислительные системы*, 2009. № 1. С. 23–37.
409. *Yeh E., Ramage D., Manning C. D., Agirre E., Soroa A.* Wikiwalk: Random walks on Wikipedia for semantic relatedness // *TextGraphs Workshop*. 2009. P. 41–49.
410. *Варламов М. И., Коршунов А. В.* Расчет семантической близости концептов на основе кратчайших путей в графе ссылок Википедии // *Машинное обучение и анализ данных*. 2014. Т. 1. № 8. С. 1107–1125.
411. *Novak J. D., Canas A. J.* The theory Underlying Concept Maps and How to Construct and Use Them. 2006.
412. *Nonaka I.* Management of Knowledge-Creation. 1990.
413. *Liu Sh.-H., Lee G.-G.* Using a concept map knowledge management system to enhance the learning of biology // *Computers Education*. 2013. V. 68. P. 105–16.
414. *Castro A. G., Rocca-Serra P., Stevens R., Taylor C., Nashar K., Ragan M. A., Sansone S.-A.* The use of concept maps during knowledge elicitation in ontology development processes – the nutrigenomics use case // *BMC Bioinformatics*. 2006. V. 7. P. 267.
415. *Brilhante V., Macedo G. T., Macedo S. F.* Heuristic transformation of well-constructed conceptual maps into OWL preliminary domain ontologies // *Proceedings of the Second Workshop on Ontologies and their Applications*. 2006.
416. Tomsk Polytechnic University. Arranging ISO 13606 Archetypes into a Knowledge Base Using UML Connectors. 2014.
417. *Dorodnykh N. O., Yurin A. Yu.* Using UML Class Diagrams for Design of Knowledge Bases of Rule-Base Expert Systems // *Software Engineering*. 2015. V. 4. P. 3–9.

Научное издание

ПАРАДИГМЫ ЦИФРОВОЙ ЭКОНОМИКИ

**Технологии искусственного интеллекта
в финансах и финтехе**

Под редакцией

М. А. Эскиндарова, В. И. Соловьева

Оригинал-макет и верстка – *С. С. Фёдоров*

Корректор – *О. В. Шапошникова*

Издательство «Когито-Центр»

129366, Москва, ул. Ярославская, д. 13

Тел.: +7 (495) 540-57-27

E-mail: post@cogito-shop.com, cogito@bk.ru

www.cogito-centre.com

Сдано в набор 18.01.19. Подписано в печать 02.02.19
Формат 60×90/16. Бумага офсетная. Печать офсетная
Гарнитура NewtonС. Усл. печ. л. 20,4. Уч.-изд. л. 20
Тираж 500 экз. Заказ 2698

Отпечатано в ПАО «Т8 Издательские Технологии»
109316, г. Москва, Волгоградский проспект, д. 42, корп. 5, ком. 6

