Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Томский государственный университет систем управления и
радиоэлектроники» (ТУСУР)

На правах рукописи

Грибков Егор Игоревич

НЕЙРОСЕТЕВЫЕ МОДЕЛИ НА ОСНОВЕ СИСТЕМЫ ПЕРЕХОДОВ ДЛЯ ИЗВЛЕЧЕНИЯ СТРУКТУРИРОВАННОЙ ИНФОРМАЦИИ О ПРОДУКТАХ ИЗ ТЕКСТОВ ПОЛЬЗОВАТЕЛЕЙ

Специальность 05.13.17 —
«Теоретические основы информатики»

Диссертация на соискание учёной степени
кандидата технических наук

Научный руководитель:
доктор технических наук, профессор
Ехлаков Юрий Поликарпович

Томск — 2020

Оглавление

Стр.

Стр.

Введение

На протяжении последнего десятилетия в связи с ростом доступности интернета и созданием множества интернет-ресурсов, в частности социальных сетей и форумов, наблюдается бурное развитие методов для автоматизации обработки текстов естественного языка, в том числе текстов, в которых потребитель выражает свое мнение о приобретенном товаре или услуге. Это приводит к лавинообразному распространению позитивной и негативной информации о потребительских свойствах продукта, что может как увеличить, так и обрушить продажи, повредить репутации компании, сделать её продукцию менее конкурентоспособной в будущем. От того, насколько быстро и точно выявляются сильные и слабые стороны собственных продуктов и продуктов конкурентов, зависит успешность бизнеса компании и её дальнейшая судьба. Основной формой передачи информации в интернете являются текст. Однако несмотря на письменную форму тексты в интернете чаще всего носят неформальный характер и изобилуют расхождениями с письменной нормой (употребление сленга, жаргонизмов, просторечных слов, игнорирование правил пунктуации, использование редкуции [1; 2]). Отмеченные особенности текстовой информации пользователей, а также большие объемы порождаемой пользователями информации делают ручной анализ трудновыполнимой задачей. В этой связи автоматизация обработки и анализа содержимого текстов в части наличия и извлечения информации о товарах является актуальной задачей.

Для качественного решения этой задачи широкое распространение получили методы на основе машинного обучения. Общие вопросы применения методов машинного обучения для обработки текстов исследовались в работах таких ученых, как К. Дайер, Й. Голдберг, К. Д. Мэннинг, Н. Клахбеннер, З. Янг. При анализе пользовательских текстов в литературе набольшее внимание уделяется вопросам анализа тональности — эмоционального окраса текста. Анализ тональности пользовательских текстов является предметом интереса в работах Б. Пана [3; 4], П. Д. Терни [5], Н. В. Лукашевич [6—8], Е. В. Тутубалиной [9; 10], В. Вана. В наиболее распространенной формулировке определяется тональность всего текста или его структурной части (предложения, параграфа). Вместе с тем в работах Т. Т. Тета, И. Андроцопулиоса, Д. Вагнера и др. отмечается, что методы анализа тональности дают слишком обобщенную

оценку объектам интереса пользователя и не предоставляют детальной информации об аспектах – составных частях, атрибутах или характеристиках оцениваемых пользователями объектов. В своих работах они предлагают расширенную постановку задачи, получившей название аспектно-ориентированного анализа тональности (АОАТ), в которой требуется определять в текстах аспекты и их тональность. Решение задачи в такой постановке позволяет получить представление о сложных продуктах с большим количеством эксплуатационных характеристик, в которых потребители могут положительно оценивать одни качества, но высказывать смешенные эмоции относительно других в пределах одного текста. Дополнительно данная тематика развивалась в работах С. Джаббары, П. Симиано, А. Катияра, О. Ирсоя, рассматривающих проблемы извлечения из текстов пользователей мнений, представленных в виде составных объектов, содержащих помимо аспектов связанные с ними оценочные высказывания, например «батарея хорошо держит заряд», «ножки стула слишком короткие».

Однако существующие методы извлечения структурированной информации из текстов пользователей производят извлечение отдельных составляющих и их объединение в результирующую структуру с помощью многокомпонентных моделей [11; 12], которые настраиваются на решение задачи независимо друг от друга. Это приводит к эффекту распространения ошибки между отдельными компонентами и отрицательно сказывается на точности модели [13—15]. Устранению данных недостатков посвящены методы предсказания, позволяющие предсказывать структуру объекта в рамках единой модели с учетом структурных взаимосвязей между сущностями в тексте. Это устраняет риск распространения ошибки при передаче промежуточных результатов между компонентами и повышает конечную точность предсказаний. В частности, широкое применение получил подход на основе систем переходов [16—18], сводящий задачу предсказания объекта со сложной структурой к предсказанию последовательности действий, в результате исполнения которых будет получен искомый объект. Данный подход отличается линейной сложностью получения предсказаний, гибкостью при определении структуры объектов и возможностью использования стандартного аппарата нейронных сетей для извлечения признаков из текста.

Таким образом, использование методов предсказания в задачах, связанных с извлечением структурированной информации из текстов пользователей, является весьма актуальным.

Целью диссертационной работы является развитие методов предсказания составных объектов с использованием нейронных сетей в части извлечения структурированной информации из пользовательских текстов на естественном языке.

Для достижения цели необходимо решить следующие **задачи**.

1. Выявить специфику задачи извлечения и анализа структурированной информации из текстов пользователей с точки зрения методов обработки естественного языка.

2. Провести анализ современных методов обработки естественного языка на основе машинного обучения.

3. Разработать нейросетевую модель на основе системы переходов для извлечения составных объектов и их атрибутов из текстов отзывов на естественном языке.

4. Разработать нейросетевую модель и алгоритм для извлечения и анализа пользовательских мнений из текстов отзывов о потребительских свойствах товаров и подготовить обучающий набор данных.

5. Разработать нейросетевую модель и алгоритм обработки запросов пользователей на этапе эксплуатации и сопровождения программного продукта, подготовить обучающий набор данных.

6. Провести экспериментальное исследование предложенных моделей на материале подготовленных наборов данных.

7. Реализовать предложенные модели и алгоритмы в виде модулей программного комплекса и провести их практическую апробацию и внедрение.

Объектом исследования являются неструктурированные тексты на естественном языке, в которых пользователи высказывают свои мысли, мнения, замечания об опыте эксплуатации различных продуктов, а также свои пожелания и запросы производителям.

Предметом исследования являются модели для извлечения и анализа структурированной информации из текстов на естественном языке на основе нейронных сетей с применением подхода на основе системы переходов.

Теоретическую и методологическую базу исследования составили труды ведущих российских и зарубежных специалистов в области обработ-

ки естественного языка, лингвистики, машинного обучения и нейросетевых методов. Информационной базой являются материалы, опубликованные в периодической печати, учебной и научной литературе, сети Интернет.

Методы исследования. Диссертационная работа опирается на методы обработки естественного языка, построения и обучения нейронных сетей, методы предсказания структурированных данных.

Область исследования диссертационной работы соответствует указанным в паспорте специальности 05.13.17 «Теоретические основы информатики» пунктам:

– п. 5 «Разработка и исследование моделей и алгоритмов анализа данных, обнаружения закономерностей в данных и их извлечениях разработка и исследование методов и алгоритмов анализа текста, устной речи и изображений»;

– п. 6 «Разработка методов, языков и моделей человекомашинного общения; разработка методов и моделей распознавания, понимания и синтеза речи, принципов и методов извлечения данных из текстов на естественном языке».

Научная новизна полученных в диссертационной работе результатов.

1. Предложена оригинальная нейросетевая модель на основе системы переходов для извлечения составных объектов и их атрибутов из текстов на естественном языке, позволяющая одновременно предсказывать структуру объекта и значения его атрибутов, с возможностью адаптации под конкретные задачи через задание множеств, описывающих семантику фрагментов и атрибутов.

2. На основе предложенной модели разработана оригинальная модель для извлечения и анализа мнений из текстов пользовательских отзывов о продуктах, отличающаяся от известных моделей использованием подхода на основе системы переходов и позволяющая получить лучшие показатели качества извлечения частей составных объектов: 0,795 F_1 – при определении фрагментов, 0,723 F_1 – при определении отношений, 0,631 F_1 – при определении атрибутов.

3. На основе предложенного метода разработана оригинальная модель для анализа запросов пользователей на этапе эксплуатации и сопровождения программного продукта, отличающаяся от известных моделей использованием подхода на основе системы переходов и позволяющая получить лучшие показатели качества извлечения частей составных объектов: 0,633 F_1 – при извлечении фрагментов, 0,693 F_1 – при извлечении отношений.

Теоретическая ценность работы заключается в развитии методов обработки естественного языка, в частности методов предсказания объектов со сложной структурой с использованием моделей на основе системы переходов и нейросетевого подхода в задачах, связанных с обработкой текстов мнений пользователей о продуктах.

Практическая значимость работы обуславливается возможностью использования разработанных моделей и программных средств в следующих случаях:

1) при анализе текстов отзывов пользователей о продуктах маркетологами компаний как для определения сильных и слабых сторон собственных продуктов и продуктов фирм-конкурентов, а также последующей модификации комплекса маркетинговых мероприятий для улучшения положения продукта на рынке;

2) на этапе эксплуатации и сопровождения программного продукта специалистами службы технической поддержки пользователей для обеспечения эксплуатации программного продукта в соответствии с его техническими характеристиками и развития продукта в соответствии с предложениями пользователей и требованиями рыночной ситуации;

3) при сравнении альтернативных предложений потенциальными покупателями товаров интернет-магазина «AliExpress» посредством сервиса «Quiddi.ru» с целью выбора товара, оценка качества которого в наибольшей степени подкреплена информацией их отзывов других покупателей.

Результаты диссертационного исследования использованы:

– в ФГБОУ ВО «ТУСУР» при выполнении государственного задания Министерства науки и высшего образования РФ, проект FEWM-2020-0036 «Методологическое и инструментальное обеспечение принятия решений в задачах управления социально-экономическими системами и процессами в гетерогенной информационной среде»;

– в учебном процессе кафедры автоматизации обработки информации (АОИ) ТУСУРа при чтении курса лекций и проведении практических занятий по дисциплинам «Интеллектуальные вычислительные системы», «Анализ больших данных» при подготовке магистров по направлению 09.04.04 — «Программная инженерия»;

– при реализации коммерческих продуктов компании ООО «ТомскСофт»: программной системы для извлечения и анализа мнений о потреби-

тельских свойствах товаров «Quiddi Semantics» (свидетельство о регистрации программы для ЭВМ №2019612276 от 14.02.2019),программной системы для обработки запросов пользователей на этапе эксплуатации и сопровождения программного продукта «Quiddi Support Analyst» (свидетельство о регистрации программы для ЭВМ №2020614799 от 24.04.2020).

Достоверность полученных результатов обусловлена корректным применением аппарата нейронных сетей при разработке модели, а также количественным сравнением предложенных моделей с аналогами. Адекватность предложенных в работе моделей и алгоритмов подтверждается результатами их практического использования в коммерческих программных продуктах компании ООО «ТомскСофт».

Публикации. Основные результаты по теме диссертации изложены в 10 печатных изданиях, 4 из которых изданы в журналах, рекомендованных ВАК [19—22], 4 — в тезисах докладов [23—26], получены 2 свидетельства о регистрации программ для ЭВМ [27; 28].

Апробация результатов работы Основные результаты диссертационной работы докладывались на конференциях различного уровня. Среди них:

1) всероссийская научная конференция молодых ученых «Наука. Технологии. Инновации» (03–07 декабря 2018 г., г. Новосибирск, НГТ);

2) международная научно-практическая конференция «Электронные средства и системы управления» (20–22 ноября 2019 г., г. Томск, ТУСУР);

3) международная научно-техническая конференция студентов, аспирантов и молодых ученых «Научная сессия ТУСУР» (2019-2020 гг., г. Томск, ТУСУР).

Личный вклад. Автором самостоятельно выполнены анализ современных методов обработки естественного языка на основе машинного обучения, предметной области, теоретическое и экспериментальное исследование разработанных моделей и алгоритмов, проектирование и реализация подсистем обучения моделей и анализа текста в составе программных систем «Quiddi Semantics» и «Quiddi Support Analyst». Совместно с научным руководителем разработаны содержательная и математическая постановки задач, предложены структуры классификаторов типов мнений пользователей и типов информативных фраз о программном обеспечении. Сервис для разметки текстов разработан Трошиным М.В., подсистемы для сбора информации и обращений пользователей разработаны совместно Трошиным М.В. и Пекарских Е.А.

Основные положения, выносимые на защиту.

1. Нейросетевая модель на основе системы переходов для извлечения составных объектов и их атрибутов из текстов на естественном языке, позволяющая одновременно извлекать фрагменты объектов и определять взаимосвязи между ними с возможностью адаптации к конкретной предметной области через задание множеств, определяющих смысловое наполнение фрагментов составных объектов и из атрибутов.

2. Нейросетевая модель для извлечения и анализа пользовательских мнений из текстов отзывов о потребительских свойствах товаров, разработанная на основе предложенной модели общего вида и обеспечивающая точность определения фрагментов 0,795 F_1, отношений — 0,723 F_1, атрибутов — 0,631 F_1.

3. Нейросетевая модель для обработки запросов пользователей на этапе эксплуатации программного продукта, разработанная на основе предложенной модели общего вида и обеспечивающая точность определения фрагментов 0,633 F_1, отношений — 0,693 F_1.

Объем и структура работы. Диссертация состоит из введения, трёх глав, заключения и двух приложений. Полный объём диссертации составляет 128 страниц, включая 37 рисунков и 20 таблиц. Список литературы содержит 127 наименований.

Глава 1. Методы машинного обучения в задачах информационной поддержки процессов эксплуатации и сопровождения продуктов

1.1 Особенности текстов пользователей о продуктах на этапах эксплуатации и сопровождения

Конкурентоспособность продуктов (товаров) зависит от множества факторов, связанных с позиционированием продукта на рынке, соответствия потребительских свойств продукта требованиям конечных потребителей, умения производителей реагировать на просьбы потребителей, а также времени реакции на эти просьбы. В моделях жизненного цикла продукта эти вопросы рассматриваются на этапах эксплуатации и сопровождения продукта и сводятся к решению следующих задач [29]:

– техническая поддержка пользователей для обеспечения эксплуатации продукта в соответствии с его техническими характеристиками;

– развитие (модификация) продукта в соответствии с предложениями пользователей и требованиями рыночной ситуации;

– модификация комплекса маркетинговых мероприятий для улучшения положения продукта на рынке.

Для организации процесса *технической поддержки* пользователей в компаниях создаются специальные подразделения по взаимодействию с конечными пользователями, отличающиеся видом используемой информации (звук, текст), временными задержками (синхронное и асинхронное общение) между получением запроса и реакцией на него. В задачи службы технической поддержки входят следующие функциональные обязанности: отвечать на запросы конечных пользователей по проблемам эксплуатации продуктов, связанных с их некачественной работой, ошибками в технической документации, недостаточной квалификацией конечных пользователей. Кроме того, у конечных пользователей могут возникать пожелания по улучшению технических и эксплуатационных характеристик продуктов.

Модификация продукта предполагает улучшение дизайна, функциональных и нефункциональных характеристик продукта. Придавая продукту новые потребительские свойства, компания зарабатывает репутацию инноватора и

закрепляет лояльность целевых сегментов, для которых эти новые свойства считаются важными. Однако следует заметить, что если компания не будет постоянно стремиться к сбору и анализу пользовательских мнений о продукте, то одноразовая модификация продукта вряд ли окупится в долгосрочной перспективе.

Модификация комплекса маркетинговых мероприятий направленна на улучшение понимания компанией своих клиентов: какие товары они предпочитают, какие положительные и отрицательные характеристики приобретаемых товаров они выделяют. Опираясь на эти данные, компания может изменять один или несколько элементов маркетинга (цену, рекламу, стимулирование сбыта и т. д.), предпринимать другие действия, направленные на повышение лояльности покупателей и улучшение конкурентоспособности продукта.

Одним из направлений, способствующих эффективному реагированию бизнеса на качество решения задач, связанных с процессами эксплуатации и сопровождения продукта, является сбор и анализ вопросов, мнений и пожеланий пользователей о проблемах, качестве и потребительских свойствах продуктов, выявленных в процессе их использования (эксплуатации). В классическом маркетинге эти вопросы решаются в рамках системы управления маркетинговой информацией как системы взаимосвязи людей, технических средств и методических приемов для сбора, анализа и оценки информации для планирования, реализации и контроля маркетинговых мероприятий [29].

Мнением в данной работе называется оценочное суждение пользователя о продукте или его характеристике, выраженное в виде текстовой информации на естественном языке и имеющее определенную эмоциональную окраску. Основными источниками мнений потребителей в классическом маркетинге являются такие методы маркетинговых исследований как анкетирование и организация фокус-групп. К существенным недостаткам данных методов можно отнести:

1) временные затраты, так как подготовка анкет, проведение очного или заочного анкетирования и последующая обработка результатов может занимать продолжительное время;

2) значительные финансовые затраты, связанные с проведением исследования самостоятельно или наймом профессионального маркетингового агентства;

3) ограниченную широту охвата из-за ограниченности ресурсов.

Такое положение дел ограничивает доступность маркетинговых исследований для небольших компаний, желающих организовать мониторинг положения продукта на рынке и не имеющих для реализации достаточного количества денежных средств. Вместе с тем развитие интернета породило большое число новых источников маркетинговой информации, из которых можно узнать мнения пользователей о конкретных продуктах. К ним следует отнести три группы интернет-ресурсов: социальные сети, страницы интернет-магазинов, торговые агрегаторы.

Социальные сети позволяют пользователям обмениваться друг с другом сообщениями в разных форматах. Некоторые ресурсы («Вконтакте», «Facebook») позволяют создавать полноценные публикации и объединяться в сообщества по интересам. Другие же предназначены в первую очередь для обмена короткими сообщениями.

Многие интернет-магазины предоставляют возможность пользователям оставлять отзывы о приобретаемых товарах, задавать вопросы об особенностях эксплуатации и делиться мнениями об опыте использования. Для выделения особенно ценных отзывов или мнений часто вводится система рейтингов, в рамках которой сообщество оценивает полезность предоставленной информации.

Торговые агрегаторы, ярким представителем которых в России является «Яндекс Маркет», собирают информацию о продуктах из множества магазинов на одном ресурсе и позволяют пользователям находить наилучшие предложения. На этом ресурсе доступна также система структурированных отзывов, в которой пользователь может выделить плюсы и минусы товара, предоставить развернутый комментарий. Наличие таких площадок позволяет постоянно расширять аудиторию и увеличивать объемы информации об обратной связи.

Использование пользовательского контента с этих ресурсов в качестве дополнительной маркетинговой информации имеет несколько существенных плюсов. Во-первых, возможно получение большого количества мнений без особых затрат со стороны заинтересованных лиц. Так, современные социальные площадки типа Twitter позволяют получать постоянный поток сообщений пользователей с помощью доступных API (Application Program Interface). Если такой возможности нет, то не представляет большой технической сложности создать средства, собирающие необходимую информацию с веб-страниц. Во-вторых, существенно упрощается проведение повторного или уточняющего анализа, так как нет необходимости в организации повторного сбора

фокус-групп, проведения собеседований и обработки результатов. В-третьих, собранные таким образом данные могут быть более репрезентативными, так как размер выборки превышает таковые при интервьюировании.

Основной объем пользовательского контента в рассмотренных источниках составляет текстовая информация, иногда сопровождаемая изображениями товара. Составляемые пользователями тексты, как правило, можно отнести к разговорному стилю речи. Для него характерна экспрессивность, наличие большого количества эмоционально насыщенной лексики, неоднородность используемой лексики. Синтаксическая структура предложений обычно проста и не содержит сложных причастных и деепричастных оборотов, однако осложняется свободным порядком слов. В текстах пользователи часто опускают знаки препинания или заменяют их путем структурирования текста с помощью знаков переноса строки, табуляции и прочих. Характерным является присутствие большого числа опечаток и ошибок при написании слов. Отмеченные особенности текстовой информации пользователей, а также большие объемы порождаемой пользователями информации делают ручной анализ трудновыполнимой задачей. В этой связи автоматизация обработки и анализа содержимого текстов на естественном языке на предмет наличия и извлечения пользовательских мнений является актуальной задачей.

1.2 Методы анализа мнений пользователей на естественном языке

1.2.1 Анализ мнений на уровне текста и его элементов

Вопросы обнаружения, извлечения, анализа и оценки субъективной информации из текстов, представленных на естественном языке, рассматриваются в контексте анализа тональности (sentiment analysis) или извлечения мнений (opinion mining). Под тональностью понимается определённое эмоциональное состояние, которое автор пытается выразить в тексте с целью отобразить испытываемое настроение либо вызвать его у читателя.

В настоящее время в литературе предложено достаточно большое количество работ, посвящённых методам анализа тональности текстов мнений

пользователей, различающихся глубиной анализа текстов, используемыми шкалами тональности, применяемыми моделями и признаками. На рисунке 1.1 приведена классификация методов по детальности анализа, на рисунке 1.2 – по используемым моделям и алгоритмам.

По глубине анализа существующие подходы можно условно разделить на две группы: анализ тональности текста на уровне текстов или его элементов; анализ тональности текста на уровне отдельных аспектов.

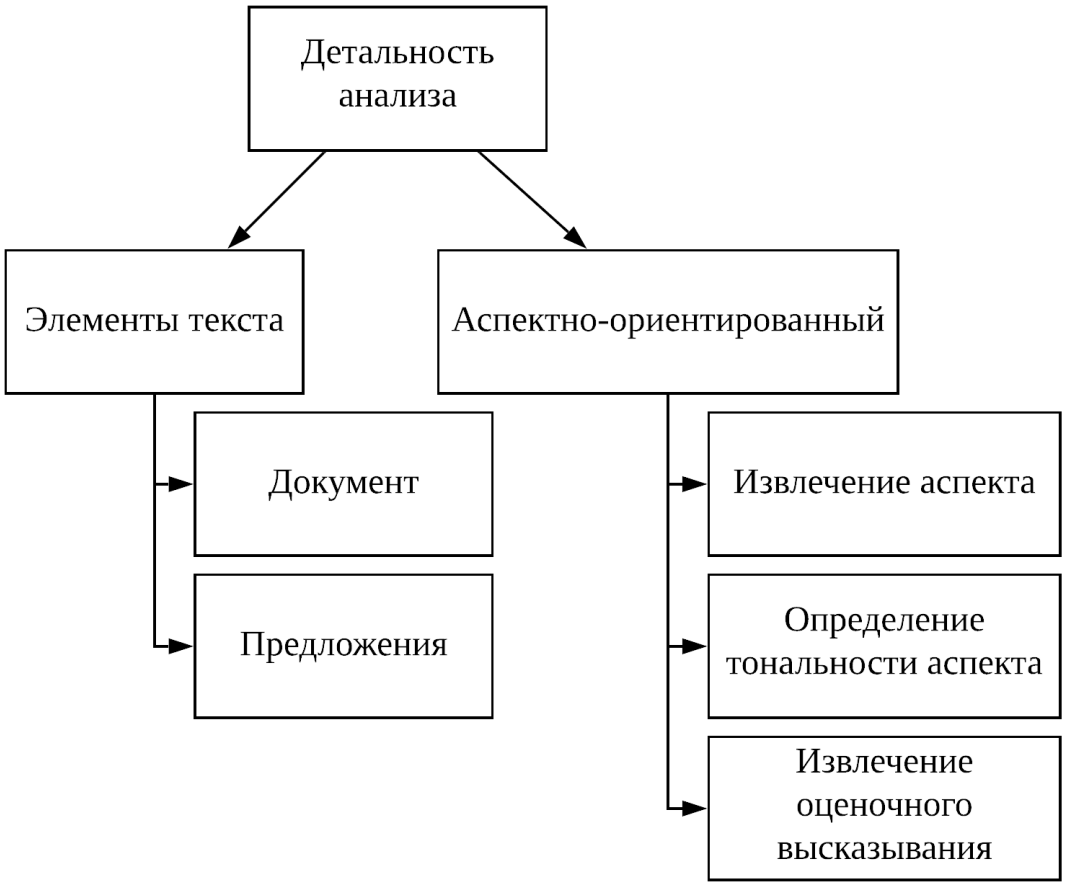


Рисунок 1.1 — Классификация задач анализа мнений по степени детализации

Большинство работ по анализу тональности текстов целиком рассматривает текст как целое в предположении, что он посвящен одной теме [3; 5; 30]. В работах данной группы тональность, как правило, определяется в виде категориальной переменной из множества {*положительная*, *нейтральная*, *негативная*}. В такой постановке описанные в литературе методы можно условно разделить на две крупных категории [31]:

1) методы на основе словарей тональной лексики (lexicon-based);
2) методы, использующие машинное обучение.

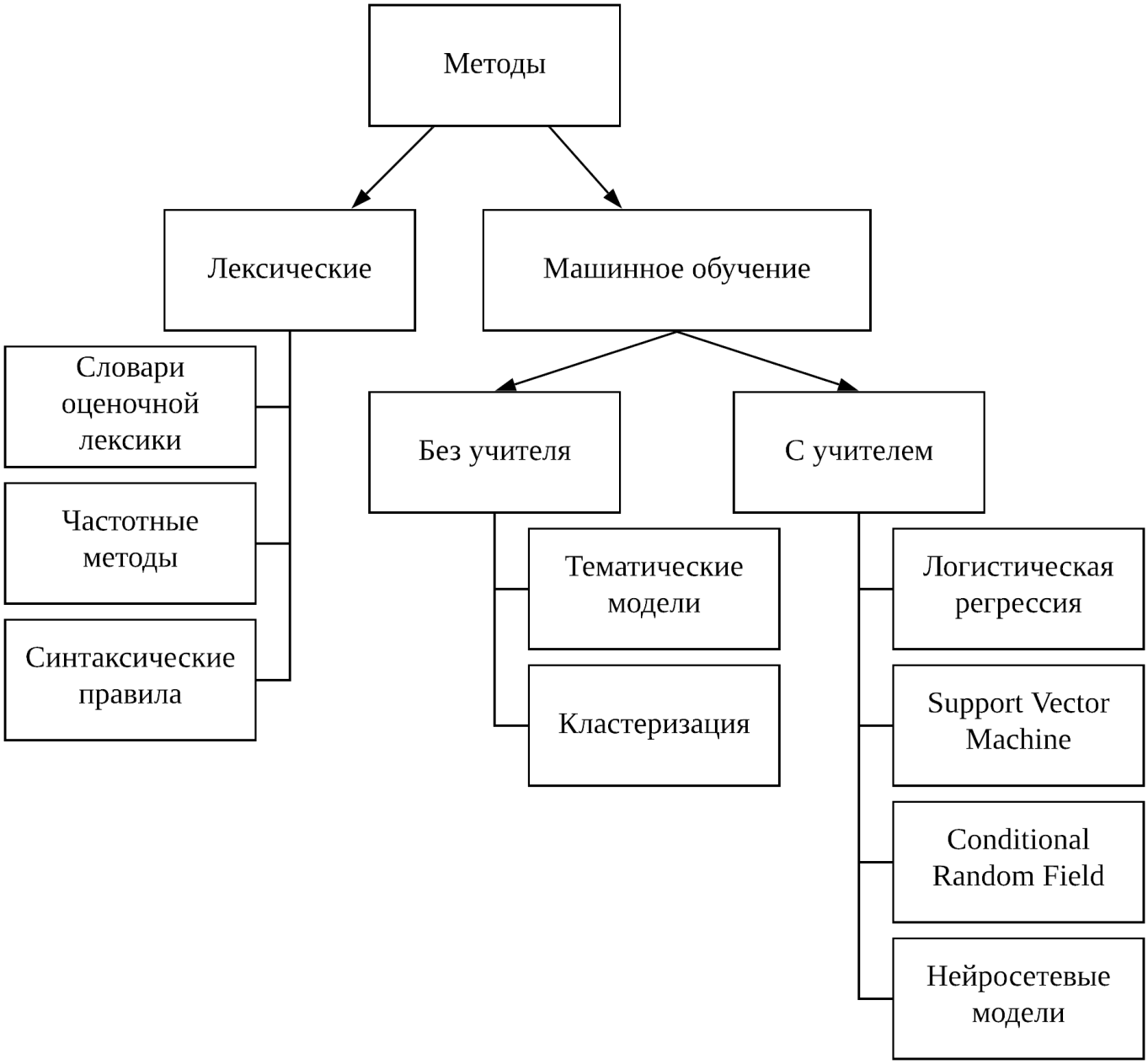


Рисунок 1.2 — Классификация задач анализа тональности по используемым методам

Группа методов на основе словарей тональной лексики опирается на идентификацию эмоционально окрашенной лексики в текстах и расчет оценки тональности на её основе. Самым простым способом получить набор эмоциональных слов и выражений является ручное составление лексикона. Однако такой способ требует большого количества времени на построение достаточно полного лексикона, пригодного для практического применения. В работах [32; 33] предлагается итеративный полуавтоматический подход к построению

лексикона. На первом этапе создается базовое множество слов с известной тональностью, которое затем расширяется с помощью поиска синонимов и антонимов в корпусе WordNet или тезаурусе [34] и добавления их в исходное множество. Итерации расширения множества повторяются до тех пор, пока есть новые слова для расширения множества. В работе [35] предлагается подход для построения лексикона на основе паттернов использования слов совместно со словами из исходного множества слов известной тональности. Для этого авторы предлагают решать задачу кластеризации прилагательных с учетом введенных ограничений на встречаемости, решением которой будут кластеры позитивных и негативных слов. В [36] предлагается подход, основанный на использовании синтаксического анализатора предложений. Авторы полагают, что каждому слову свойственна внутренняя тональность, которая распространяется на его соседей в дереве зависимостей. Используя синтаксические зависимости, можно определить тональность каждого слова и затем всего предложения. Достоинством такого подхода является возможность корректно учитывать отрицания и контекст предложения, однако для его использования необходим качественный синтаксический анализатор.

Развитие методов машинного обучения и появление общедоступных размеченных наборов данных привели к тому, что большая часть современных работ по анализу тональности так или иначе полагается на обучаемые модели. В таком случае анализ тональности формулируется как задача классификации или регрессии. Пусть задана коллекция обучающих текстов $D = \{x_i,y_i\}_{i=1}^{N}$, где $y_i \in Y$ – метка тональности для текста x_i. Задана функция получения метки тональности $y = f(x|\theta)$, параметризованная вектором θ. Необходимо найти вектор параметров $\hat{\theta}$, минимизирующий функцию потерь $L(x,y,\theta)_D$ на всем наборе данных D:

$$\hat{\theta} = \arg\min_{\theta} L(x,y,\theta)_D$$

В случае регрессии, как правило, используется $L(x,y,\theta) = \|f(x|\theta) - y\|_2$. Для классификации применяют разнообразные функции в зависимости от особенностей используемого алгоритма, но наиболее простой является индикаторная функция $L(x,y,\theta) = I(f(x|\theta) = y)$. Работы в данной категории можно рассматривать с двух сторон: по используемому методу машинного обучения и по набору используемых признаков. В одной из первых работ [4] приводится сравнение трех методов: Naïve Bayes (NB), Maximum Entropy (ME) и Support Vector

Machine (SVM), а также использование различных признаков и их комбинаций из набора элементов: слова, биграммы, части речи, прилагательные, позиция слова. Экспериментальное исследование авторов показало, что наилучшие результаты показывает SVM с bag-of-unigram признаками. В [37]исследуется проблема выбора оптимального набора признаков для задачи классификации текстов по эмоциональным категориям. Для выбора наиболее информативных признаков применяются методы взвешенного правдоподобия, взаимной точечной информации (pointwise mutual information), «нормализованное Google-расстояние». Обученный на очищенных признаках NB-классификатор имеет среднюю точность 71,35%, что позволяет говорить о полезности отсева неинформативных признаков. В последнее время широкое распространение получили методы машинного обучения на основе нейронных сетей. Авторы [38] затрагивают проблему сравнения методов классификации на основе SVM и нейронных сетей. В ходе экспериментальной проверки на множестве отзывов из магазина Amazon использование нейронной позволило авторам получить лучшие результаты на большинстве тестов. Применение свёрточных нейронных сетей в задачах классификации текста рассматривают авторы [39]. В работе сравниваются различные варианты свёрточных сетей, моделей древоподобного условного случайного поля (Tree-CRF) и SVM-классификатора. Модели на основе нейронных сетей показывают преимущества перед другими в шести задачах из семи.

Анализ мнений на уровне предложений подразумевает определение тональности в одном выбранном предложении. Так, в работе [32] авторы рассматривают алгоритм анализа пользовательских мнений на уровне отдельных предложений, где определяются эмоционально окрашенные предложения, которые затем используются для генерации краткого обзора продукта. В работе [40] предлагается подход к определению тональности предложений на основе байесовского классификатора. Существуют работы, предлагающие выявлять распределение тональности внутри одного предложения. Одной из знаковых работ в этом направлении является [41], где авторы предлагают определять тональность узлов двоичного дерева разбора предложения. Можно отметить, что использование этого подхода осложнено необходимостью подготавливать выборку размеченных деревьев. Существенным ограничением применимости методов машинного обучения с учителем является отсутствие данных с разметкой на уровне отдельных предложений. Авторы работы [42] рассматривают

подход на основе нейронных сетей, позволяющий определять тональность отдельных предложений при наличии данных о тональности всего текста.

1.2.2 Анализ мнений на уровне сущностей

Отдельное направление исследований, называемое аспектно-ориентированным анализом (aspect-oriented sentiment analysis), посвящено оценке мнений пользователей об отдельных аспектах текста. Аспектом называют слово или словосочетание, обозначающее признак, составную часть, характеристику, определенное качество товара или услуги. Пользовательские тексты могут содержать предложения, в которых сочетаются позитивные и негативные высказывания о различных аспектах. Например, в предложении «Хороший и дешевый телефон с приятным дисплеем, однако качество корпуса оставляет желать лучшего» отмечается положительный аспект «телефон» с характеристиками «хороший» и «дешевый», а также негативный аспект «качество корпуса» с характеристикой «оставляет желать лучшего».

В работе [43] предлагается выделить несколько этапов в аспектно-ориентированном анализе тональности: определение аспектов в текстах отзывов; определение тональности аспектов; группировка аспектов. На первом этапе происходит определение аспектов в текстах отзывов, после чего определяется их тональность, которая зависит от контекста, в котором употреблен термина. На третьем этапе производится группировка пар аспект-тональность и формируется отчет, по которому можно составить целостное представление о плюсах и минусах конкретных продуктов.

Определение аспектов

В работе [44] предлагается частотный метод обнаружения аспектов, заключающийся в поиске наиболее часто встречающихся в одном предложении групп существительных и последующем отсеве сначала групп слов, которые редко встречаются в предложениях рядом, и затем одиночных слов, которые часто встречаются в составе групп. Работа [45] дополняет рассмотренный подход методом поиска неявных аспектов на основе обнаружения ассоциативных правил. Неявным называется аспект, который не упоминается явно в тексте, например: «Хороший телефон, несмотря на тяжесть.». В данном примере присутствуют яв-

ный аспект «телефон» и неявный аспект «вес», индикатором которого является слово «тяжесть». Для частотных методов характерно большое число ложноположительных срабатываний, связанное с тем, что существительные и именные группы, имеющие высокую частотность в среднем (для текстов на определенном языке), могут быть ошибочно распознаны как аспекты. В этой связи в [46] рассматривается методика фильтрации найденных кандидатов в аспекты путем сравнения частоты встречаемости кандидата в текстах обзора с частой в большом корпусе произвольных текстов.

В отличие от частотных методов синтаксические подходы основываются на синтаксических отношениях между словами. Это позволяет обнаруживать не только часто употребляемые аспекты. В [47] рассматривается метод обобщения синтаксических паттернов, содержащихся в обучающей выборке. Авторы предлагают разделять паттерны и синтаксические деревья предложений из неразмеченной выборки на части и оценивать их похожесть. Если похожесть двух структур превышает некоторый заданный порог, то засчитывается обнаружение нового синтаксического паттерна. В работах [48—50] описываться подход, который рассматривает обнаружение аспектов и построение словаря оценочных слов как взаимосвязанные задачи. Для их решения предлагается использовать алгоритм двойного распространения, в котором сначала на основе имеющихся оценочных слов и синтаксических паттернов расширяют множество известных аспектов, затем расширенное множество используют для поиска новых оценочных слов.

В ряде работ [51; 52] задача поиска аспектов сводится к задаче разметки последовательности с помощью условных случайных полей и рекуррентных нейронных сетей. Основное внимание в этих работах уделяется вопросам обоснования выбора определенного набора признаков, таких как n-грамы, частеречная разметка, связи из деревьев зависимости и т.п. Эксперименты, проведенные в работе [52] на примере набора пользовательских отзывов с дорожки SemEval-2014, показывают превосходство моделей на основе рекуррентной нейронной сети Long-Short Term Memory (LSTM) над CRF-моделями при решении задачи обнаружения аспектов, даже несмотря на то, что в LSTM в качестве признаков используются только векторные представления слов. Добавление лингвистических признаков (части речи, информация о чанках) улучшает качество результатов, получаемых LSTM.

Широкое распространение получили методы поиска аспектов на основе модели латентного размещения Дирихле (Latent Dirichlet Allocation, LDA) [53]. LDA – это модель обучения без учителя (unsupervised learning), предназначенная для отыскания тем-топиков в наборе текстов. Топиком называется распределение, задающим вероятность порождения определенного слова в тексте определенной тематике. В изначальном варианте модель LDA не подходит для извлечения аспектов, так как получаемые ею топики не обязательно содержат в себе аспекты. В работе [54] предлагается подход, в котором выделяют два вида топиков: глобальные, описывающие текст всего отзыва, и локальные, описывающие отдельные предложения. Предполагается, что локальная модель выявляет аспекты, а глобальная — все остальные слова. Авторы [55] рассматривают модель LDA с разделенным словарем, одна часть которого содержит существительные, а вторая — слова, зависимые от них (прилагательные и т.п.). В [56] решается проблема «холодного старта», заключающаяся в том, что для получения надежных оценок параметров в LDA требуется большое количество текстов отзывов. Авторы предлагают дополнительно моделировать тональность продуктов и объединять товары в группы. В этом случае возможно настраивать параметры модели в рамках группы отзывов, что увеличивает статистическую эффективность методов.

Определение тональности аспектов

Работы, посвященные определению тональности аспектов, как правило, используют методы, применяемые при анализе тональности на уровне текстов и его элементов. Так, в работах [32; 57; 58] предлагаются методы на основе словарей оценочной лексики и других лексических источников. В работе [59] для предсказания тональности аспекта предлагается использовать SVM, признаками которого выступают слова оценочной лексики и аспекты из описаний плюсов и минусов продуктов.

Методы совместного поиска аспектов и определения их тональности

Рассмотренные ранее методы позволяли решать задачи поиска аспектов и определения их тональности раздельно как две независимые задачи. Однако в действительности эти задачи взаимосвязаны, и использование статистических закономерностей в рамках одной модели может улучшить качество решения обеих задач. Это делает актуальной разработку методов, позволяющих решать обе задачи одновременно. Авторы [12] рассматривают применение SVM для совместного определения аспектов и оценочной лексики в виде пар

«аспект–оценка». Особенностью данного подхода является способность предсказывать связанность аспекта и оценочного слова даже в тех случаях, когда между ними нет прямой синтаксической связи. Сравнение методов на основе условных случайных полей, Tree-CRF и Skip-Tree-CRF, в которых проблема поиска и оценки аспектов решается как задача разметки последовательности, проведено в работе [60]. Проведенные эксперименты показывают, что Skip-Tree-CRF имеет лучшие показатели точности по сравнению с другими моделями. Авторы объясняют это использованием зависимостей между скрытыми вершинами CRF. В [61] предлагается подход на основе скрытых марковских моделей, в котором одновременно моделируются отдельные слова и n-грамы частей речи. Дополнительно авторы используют подход самообучения модели, при котором модель обучается на примерах вне обучающего множества, предсказанных ею самой. Рассмотренные работы уделяют основное внимание поиску аспектов. Авторы [62] предлагают гибридную модель для поиска оценочных высказываний, связанных с конкретными аспектами. Это позволяет извлекать из текстов отзывов выражения, содержащие аспект и мнение пользователя о нем. Например, «firmware update without a hitch,», «firmware update without a hitch». Извлечение подобных структур позволяет извлекать более детализированную по сравнению с другими подходами информацию о продукте. В работе [63] был предложен набор отзывов из интернет-магазина Amazon, в котором были выделены аспекты, оценочные высказывания, а также связи между ними с указанием тональности образованный пары. Для извлечения авторы предлагают использовать графическую вероятностную модель, которая показывает точность извлечения связей по критерию F_1 равную 65%. Авторы отмечают сложность создания такого корпуса и использования методов машинного обучения для решения поставленной задачи, относительно низкая согласованность результатов двух асессоров, занимавшихся разметкой отзывов.

1.3 Нейросетевые модели для анализа текста на естественном языке

В последнее время широкое распространение получили методы обработки естественного языка на основе нейронных сетей. Это в первую очередь

обусловлено наличием эмпирических данных, показывающих высокое качество получаемых ими результатов и возможностью работать с «сырыми», минимально обработанными данными. Подтверждение эффективности использования нейронных сетей можно найти в работах, посвященных решению задач классификации текстов [38; 39; 64; 65], синтаксического анализа [16; 18; 66; 67], статистического машинного перевода [68—70].

В настоящее время широкое использование для решения задач обработки естественного языка получили методы, опирающиеся на два типа моделей:

– модели распределенных векторных представлений слов;

– модели для извлечения признаков из последовательностей на основе свёрточных и рекуррентных нейронных сетей.

Для применения нейросетевых моделей при обработке текстов, исходные данные необходимо представить в числовой форме. Одним из наиболее простых способов преобразования является преобразование в унитарный код, где каждому слову w с порядковым номером i из некоторого фиксированного словаря V в соответствие ставится $\overrightarrow{0}$-вектор размера $|V|$, в котором i-ый элемент равен 1. Однако в контексте моделей на основе нейронных сетей наилучшие результаты получают распределенные векторные представления [71; 72]. В данном случае каждому слову соответствует вектор $\mathbf{w} \in \mathbb{R}^n$, где $n \ll |V|$ ($n \in [100; 300]$). В отличие от унитарного кода, вектор w является «плотным» – т. е. не содержит большого количества нулевых элементов. В работах [73; 74] предложены методы получения распределенных векторных представлений слов путем обучения нейронной лингвистической модели, которая минимизирует ошибку предсказания контекста для каждого слова в большом корпусе. В процессе обучения векторные представления семантически похожих слов сближаются друг с другом в векторном пространстве. После обучения полученные представления могут использоваться как часть другой модели для решения прикладных задач [39; 75; 76].

Извлечение признаков из текста, представленного последовательностью слов, можно представить в виде следующего преобразования:

$$\mathbf{h}_i = F(E(w_1,w_2,...,w_N), i),$$

где $E(w_1,w_2,...,w_n)$ – последовательность векторных представлений слов, i – позиция слова в предложении. Более сложные признаки образуются путем ис-

пользования последовательности преобразований:

$$\mathbf{h}_i^1 = F_1(E(w_1,...,w_N), i),$$
$$\mathbf{h}_i^2 = F_1(h_1^1,...,h_N^1, i),$$
$$\mathbf{h}_i^j = F_j(h_1^{j-1},...,h_N^{j-1}, i)$$

Одним из основных требований к преобразованию F является возможность учитывать при формировании нового вектора признаков не только представление на i-й позиции, но и некоторый контекст, в котором это представление находится. Этому требованию удовлетворяют признаки, полученные с помощью свёрточных или рекуррентных нейронных сетей.

Свёрточные нейронные сети в настоящее время широко используются в задачах обработки естественного языка [39; 64; 77—79]. Как правило, при этом используются одномерные свертки с одной степенью пространственной свободы. Каждый вектор выходной последовательности на позиции i формируется на основе векторов входной последовательности, находящихся в окне размера k с центром i по следующей формуле:

$$\mathbf{h}_i^j = \sigma(\mathbf{W}\left[\mathbf{h}_{i-(k-1)/2}^{j-1}; ...; \mathbf{h}_{i+(k-1)/2}^{j-1}\right] + \mathbf{b}), \tag{1.1}$$

где $\mathbf{W}, \mathbf{b}$ – параметры свертки, σ – нелинейная функция активации. Для того чтобы после применения операции свертки к последовательности её длина не изменилась, используется паддинг — дополнение исходной последовательности нуль-векторами.

Наиболее распространенной функцией активации, применяемой при обучении свёрточных нейронных сетей, является rectified linear unit (ReLU):

$$\mathrm{ReLU}_i(x) = \max(0, x_i)$$

Авторы [80] показывают экспериментально, что замена tanh на ReLU ускоряет сходимость обучения в 6 раз. В работе [81] предлагается функция активации Maxout, представляющая собой обобщение ReLU с произвольным количеством линейных участков и параметрами, отвечающими за наклон каждого участка:

$$\mathrm{Maxout}_i(\mathbf{x}) = \max_{j\in[1,k]}(\mathbf{x}^T\mathbf{W}_{ij} + \mathbf{b}_{ij}),$$

где k – количество линейных участков, $\mathbf{W}$ и $\mathbf{b}$ – обучаемые параметры функции активации. Проведенные эксперименты показывают, что применение Maxout уменьшает ошибку классификации при обучении сети.

Существенного прогресса в обучении глубоких свёрточных сетей удалось достигнуть благодаря использованию архитектурного приёма сквозных соединений [82], при котором выход каждого слоя суммируется с его изначальным входом:

$$\mathbf{h}_i^j = \sigma(\mathbf{W}\mathbf{h}_i^{j-1} + \mathbf{b}) + \mathbf{h}_i^{j-1} \tag{1.2}$$

Этот прием позволяет обучать более глубокие сети, что в свою очередь повышает точность классификации изображений. Исследования показывают [83; 84], что такое решение оправдано в том числе при решении задач обработки естественного языка. Иллюстрация применения свёрточной нейронной сети с использованием сквозных соединений для извлечения признаков представлена на рисунке 1.3.

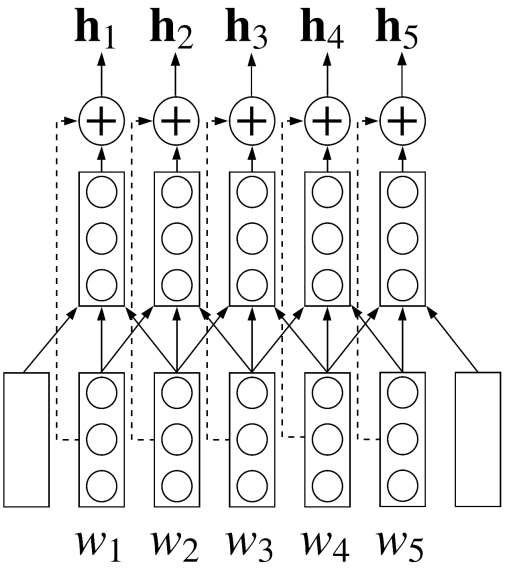


Рисунок 1.3 — Сверточная нейронная сеть с применением сквозных соединений

При использовании рекуррентных нейронных сетей (РНС) контекстные представления элементов последовательности образуются с помощью рекуррентных взаимосвязей следующего вида:

$$\mathbf{h}_i^j = \sigma(\mathbf{U}\mathbf{h}_{i-1}^j + \mathbf{W}\mathbf{h}_i^{j-1} + b)$$

В этом случае контекстная информация передается таким образом, что элемент i несет информацию обо всех предыдущих $i^\smile 1$ элементах. Для возможности учитывать контекст последующих элементов в [85] предложено использовать двунаправленные РНС, в которых итоговые контекстные вектора образуются

путем объединения векторов $\overrightarrow{\mathbf{h}}_i$ и $\overleftarrow{\mathbf{h}}_i$, полученных при прямом (слева направо) и обратном (справа налево) проходах:

$$\begin{aligned}
\overrightarrow{\mathbf{h}}_i^j &= \sigma(\overrightarrow{\mathbf{U}}\mathbf{h}_{i-1}^j + \overrightarrow{\mathbf{W}}\mathbf{h}_i^{j-1} + \overrightarrow{\mathbf{b}}),\\
\overleftarrow{\mathbf{h}}_i^j &= \sigma(\overleftarrow{\mathbf{U}}\mathbf{h}_{i-1}^j + \overleftarrow{\mathbf{W}}\mathbf{h}_i^{j-1} + \overleftarrow{\mathbf{b}}),\\
\mathbf{h}_i^j &= \left[\overrightarrow{\mathbf{h}}_i^j; \overleftarrow{\mathbf{h}}_i^j\right]
\end{aligned}$$

На практике при обучении рекуррентных сетей возникают проблемы затухающего (vanishing) и взрывного (exploding) градиента [86]. Затухающий градиент не позволяет сети выучить взаимосвязи между удаленными элементами последовательности, тогда как взрывной градиент приводит к слишком быстрому изменению весов и делает процесс обучения нестабильным. Для решения первой проблемы, в работе [87] был предложен особый тип рекуррентного блока Long Short-Term Memory (LSTM), описываемого следующим набором уравнений:

$$\begin{aligned}
\mathbf{i}_t &= \sigma(\mathbf{W}_{xi}\mathbf{x}_t + \mathbf{W}_{hi}\mathbf{h}_{t-1} + \mathbf{b}_i),\\
\mathbf{f}_t &= \sigma(\mathbf{W}_{xf}\mathbf{x}_t + \mathbf{W}_{hf}\mathbf{h}_{t-1} + \mathbf{b}_f),\\
\mathbf{o}_t &= \sigma(\mathbf{W}_{xo}\mathbf{x}_t + \mathbf{W}_{ho}\mathbf{h}_{t-1} + \mathbf{b}_o),\\
\bar{\mathbf{c}}_t &= \tanh(\mathbf{W}_{wc}\mathbf{x}_t + \mathbf{W}_{hc}\mathbf{h}_{t-1} + \mathbf{b}_c),\\
\mathbf{c}_t &= \mathbf{f}_t \otimes \mathbf{c}_{t-1} + \mathbf{i}_t \otimes \bar{\mathbf{c}}_t,\\
\mathbf{h}_t &= \mathbf{o}_t \otimes \tanh(\mathbf{c}_t)
\end{aligned} \tag{1.3}$$

В нем градиент для текущего внутреннего состояния является линейным по отношению к предыдущему, что позволяет исключить резкое уменьшение значения его нормы. Развитием LSTM является блок Gated Recurrent Unit (GRU) [68; 88], заданный следующим образом:

$$\begin{aligned}
\mathbf{z}_t &= \sigma(\mathbf{W}_{xz}\mathbf{x}_t + \mathbf{U}_{hz}\mathbf{h}_{t-1} + \mathbf{b}_z),\\
\mathbf{r}_t &= \sigma(\mathbf{W}_{xr}\mathbf{x}_t + \mathbf{U}_{hr}\mathbf{h}_{t-1} + \mathbf{b}_r),\\
\bar{\mathbf{h}}_t &= \tanh(\mathbf{W}\mathbf{x}_t + \mathbf{U}(\mathbf{r}_t \otimes \mathbf{h}_{t-1}) + \mathbf{b}),\\
\mathbf{h}_t &= (1 - \mathbf{z}_t) \odot \mathbf{h}_{t-1} + \mathbf{z}_t \odot \mathbf{h}_t
\end{aligned} \tag{1.4}$$

Авторы приводят в качестве преимущества перед LSTM простоту, меньший объем необходимых вычислений, сопоставимую с LSTM точность предсказания. Пример извлечения признаков с помощью двунаправленной рекуррентной нейронной сети представлен на рисунке 1.4.

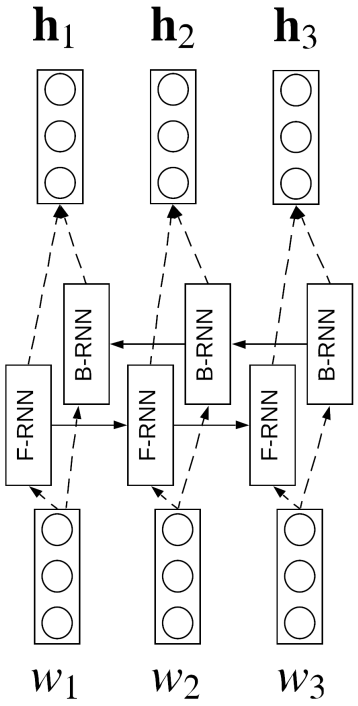


Рисунок 1.4 — Двунаправленная рекуррентная нейронная сеть

Авторы работы [89] предложили использовать две LSTM для машинного перевода: первая сеть (энкодер) представляла текст на одном языке в виде вектора фиксированного размера, после чего вторая (декодер), получая на вход вектор, предсказывала последовательность слов на другом языке. В работе [65] предложен оригинальный подход к классификации текстов с применением РНС и механизма внимания [69; 90], подразумевающий иерархический процесс формирования векторного представления классифицируемого документа: сначала РНС уровня предложений получают векторные представления составляющих текст предложений, которые затем агрегируют РНС уровня всего документа.

Проблема использования нейронных сетей в задачах предсказания структурированных объектов также хорошо представлена в научной литературе. Так, в [91] предложена модель для извлечения именованных сущностей, использующая LSTM для извлечения признаков и CRF в задаче NER. Работа [66] рассматривает графовый алгоритм синтаксического анализа текста, основанный на нейронных сетях и би-аффинном механизме внимания, достигающей высокой точности предсказания дуг и меток. В работах [16; 92] коллективом авторов предлагаются методы синтаксического разбора текстов на основе рекуррентных нейронных сетей с применением системы переходов. В частности, авторы предлагают использовать стековые LSTM для получения признако-

вых описаний элементов конфигурации, представленной стековой структурой данных. Рассматривается применение стековой LSTM к различным элементам конфигурации: стеку входных слов, стеку истории совершенных моделью действий и стеку найденных частей синтаксической структуры.

1.4 Применение нейронных сетей в задачах анализа мнений

В настоящее время модели на основе нейронных сетей широко применяются при решении самых разнообразных задач в области анализа мнений. Распространенным способом определения тональности текстов является обучение модели на основе свёрточных нейронных сетей [39; 77]. В работе [65] предложена иерархическая модель классификации, в которой с помощью рекуррентной нейронной сети с механизмом внимания (attention) получают представления отдельных предложений, а затем с помощью аналогичной сети верхнего уровня получают представление всего текста, которое передается в классификатор. Высокое качество результатов показывают нейронные сети при решении задач аспектно-ориентированного анализа тональности. Одна из первых работ в этой области [93] использует рекуррентную нейронную сеть для извлечения оценочных высказываний из текстов новостей. Авторы [52] исследуют возможности нескольких разновидностей РНС на задаче определения аспектных терминов в текстах отзывов, экспериментальное сравнение показывает, что наилучшим вариантом оказывается использование LSTM и дополнительных синтаксических признаков (информация о чанкинге и частеречной разметке). В [94] предлагается использовать вариант модели LSTM, позволяющей учитывать контекст, для определения тональности аспектов. Задача извлечения сущностей и связанных с ними оценочных высказываний, рассматривается в работе [95]. Авторы сравнивают модель CRF-LSTM с простой CRF и приходят к выводу, что первая модель в целом работает хуже, чем обычная CRF. Подход к извлечению аспектов из субтитров к видео с обзорами на различные продукты предложен в работе [96]. Авторы используют нейронную сеть LSTM с механизмом внимания и достигают лучшей по сравнению с аналогами точности без использования дополнительных признаков. Работа [97] рассматривает применение Deep Memory Networks (DMN) к задаче классифи-

кации тональности аспектов в предложениях. Вывод в данном типе нейронных сетей строится пошагово: на каждом шаге сеть получает распределение значимости отдельных предложений в формировании финального высказывания, причем при формировании распределения учитываются как входные данные, так и распределение с предыдущего шага. Авторы провели экспериментальное сравнение DMN с методами на основе SVM и LSTM, результаты показывают превосходство DMN с 9-ю шагами вывода без использования дополнительных признаков над аналогами. В [98] рассматривается похожая модель, в которую дополнительно включен компонент для обработки графа синтаксических зависимостей. Проведенная серия экспериментов показывает пользу использования синтаксических признаков в виде графа зависимостей. Оригинальный подход к аспектно-ориентированному анализу тональности предложен в [78]. Авторы предлагают отказаться от решения задачи поиска аспектов в тексте отзыва, мотивируя это тем, что для большинства товаров набор важных для пользователей характеристик известен заранее. По этой причине предлагается решать задачу классификации предложений, где классами выступают известные аспекты товаров, каждое предложение может относиться к нескольким аспектам одновременно. Для классификации используется сверточная нейронная сеть. После определения состава аспектов в предложении вторая сверточная нейронная сеть используется для классификации тональности.

В исследовании [11] рассматривается применение нейронных сетей к задаче извлечения аспектов и связанных с ними оценочных высказываний из текстов отзывов в наборе данных, предложенных в работе [63]. Авторами предлагается решение задачи извлечения пар «аспект-оценочное высказывание» в два этапа. Сначала с использованием комбинации свёрточной и рекуррентной нейронных сетей из текстов извлекаются аспекты и оценочные высказывания. Затем для оценочных высказываний производится оценка тональности с помощью рекуррентной нейронной сети. На последнем этапе производится попарная классификация аспектов и оценочных высказываний на предмет наличия связи. Работа [99] исследует извлечение аспектов и оценочных высказываний из отзывов посетителей ресторанов. Для этого авторы предлагают оригинальную модель с использованием нейронной сети и CRF, которая позволяет учитывать взаимное влияние аспектов и оценочных высказываний друг на друга. В данной работе не рассматриваются вопросы поиска явных связей между извлеченными сущностями.

Следует отметить, что в литературе, посвященной методам аспектно-ориентированного анализа тональности не уделяют достаточного внимания задаче определения структурных взаимосвязей между извлекаемыми сущностями. Можно отметить лишь несколько работ, в которых помимо аспектных терминов извлекаются связанные с ними оценочные высказывания [11; 62; 63; 99]. В работах [11; 63] рассматривается задача поиска связей между аспектами и оценочными высказываниями, однако оценка точности приведена только для ситуации, когда известны истинные расположения аспектов и оценочных высказываний. При эксплуатации модель может сталкиваться с ситуациями, когда границы сущностей определены неточно, и это ограничивает максимальную точность работы модели. Авторы [100] отмечают, что при оценке точности определения связей «от начала до конца», при котором одновременно предсказываются и сущности, и связи между ними, наблюдается ухудшение качества извлечении связей на величину порядка 0,3 F_1.

1.5 Модели на основе системы переходов в задачах обработки естественного языка

Традиционно в области машинного обучения рассматриваются задачи, в которых предсказанием является некоторое скалярное значение (регрессия) или распределение вероятности по множеству возможных значений одной переменной (классификация). Однако для многих задач в области обработки естественного языка, таких как статистический машинный перевод, частеречная разметка и определение синтаксической структуры предложений, результатом предсказания является объекты более сложные, такие как последовательности переменной длины, деревья и графы. Размер множества возможных значений выхода в таком случае может зависеть от размера входа экспоненциально. Например, в задаче разметки последовательности, где для каждого слова в предложении длинны n необходимо определить класс из множества Y с помощью функции $f(w_1,w_2,...,w_N) = y_1,y_2,...,y_N,y_i \in Y$, размер множества допустимых предсказаний равен $|Y|^n$. Таким образом, при решении практических задач свести предсказание сложной структуры к классификации путем перечисления всех возможных вариантов структуры практически

не представляется возможным. Из этого следует необходимость декомпозиции процесса предсказания согласно структуре объекта, ожидаемого на выходе. При разметке последовательностей это могут быть метки отдельных слов или перечисление фрагментов с определенной меткой, при определении деревьев зависимостей – связи между отдельными словами и их метки.

Распространённым способом преодоления этого ограничения является декомпозиция процесса предсказания объекта со сложной структурой на последовательность независимых предсказаний отдельных элементов данной структуры. В таком случае разметка последовательности длины n потребует решения n задач классификации. Если целью предсказания является более сложная структура, то исследователи зачастую прибегают к созданию конвейерных систем, состоящих из последовательно соединенных друг с другом моделей [101—103]. При этом возникает проблема распространения ошибки (англ. error propagation), когда ошибка на одном этапе конвейера распространяется далее, приводя к падению точности работы системы в целом.

В моделях на основе нейронных сетей распространенным способом смягчить данную проблему является использование общего набора параметров, разделяемого между частями моделями, специфичными для каждого отдельного этапа. В работах [101; 104; 105] отмечается, что использование общего набора параметров при извлечении признаков позволяет повысить точность по сравнению с использованием конвейерного соединения отдельных моделей. Однако при таком подходе разные части модели не могут в полной мере учитывать взаимные зависимости между выходом модели.

Одним из возможных способов учитывать взаимное влияние выходных переменных через общее разделяемое признаковое описание является использование моделей на основе системы переходов [17; 106], которые формулируют задачу предсказания структурированного объекта как предсказание последовательности переходов (transition), исполнение которой позволяет в итоге получить желаемый структурированный объект.

Формально система на основе системы переходов определяется кортежем $(C_t, Y, A(C_t))$, где C_t – конфигурация системы, Y – множество переходов, $A(C_t)$ – функция, задающая множество переходов, доступных для исполнения в текущей конфигурации. Конфигурация C_t содержит структуры данных, содержащих информацию о предсказываемом объекте и ходе предсказания состояния на шаге t. На каждом шаге на основании содержания C_t выбирается

некоторый переход из множества Y, который может вносить изменения в структуры данных из C. C_t и Y могут быть сконструированы таким образом, что в некоторых конфигурациях не все действия из Y могут быть применены к C_t. Такого рода ограничения на возможные переходы задаются с помощью функции $A(C_t) : C \mapsto Y^{`} \subseteq Y$. Выделяют также две специальных конфигурации: начальную C_0 и конечную C_T. Начальная конфигурация необходима для задания необходимых для совершения предсказаний исходных данных (например, текст документа), конечная определяет момент прекращения процесса предсказания и содержит все необходимые данные для восстановления исходного объекта. В таком случае процесс получения структурированного объекта o представляется последовательностью трансформаций конфигурации C при выполнении цепочки переходов $\mathbf{y} = (y_1,y_2,...,y_T)$:

$$C_0 \to y_1 \to C_1 \to y_2 \to ... \to y_T \to C_T.$$

Иначе говоря, структурированному объекту o можно поставить в соответствие пару $(C_0, \mathbf{y})$. Практическое применение модели на основе системы переходов требует задания каждого элемента кортежа согласно структуре результирующего объекта и особенностям решаемой прикладной задачи.

После задания модели на основе системы переходов необходимо выбрать алгоритм определения последовательности переходов $\mathbf{y}$ на основе имеющегося набора структурированных объектов. Пусть задана размеченная выборка, состоящая из множества пар (x_i,o_i), где x_i – входной текст, o_i – ассоциированный с текстом составной объект. Необходимо определить параметры условного распределения $p_\theta(o_i|x_i)$. Используя соответствие $o = (C_0, \mathbf{y})$, перепишем распределение в виде $p_\theta(y_1, y_2, ..., y_T|C_0)$. В общем случае переходы в последовательности $\mathbf{y}$ не являются независимыми, однако конфигурацию C возможно определить таким образом, чтобы она хранила в себе некоторую информацию об истории совершенных переходов и изменений состояния. В этом случае мы можем предположить взаимную независимость элементов $\mathbf{y}$ и декомпозировать вероятность в виде произведения отдельных условных вероятностей в следующем виде:

$$p_\theta(y_1,y_2,...,y_T|C_0) = \prod_t p_\theta(y_t|C_t)$$

Условная вероятность $p_\theta(y_t|C_t)$ как правило выражается через функцию признакового описания конфигурации:

$$p_\theta(y_t|C_t)_k \sim \mathbf{W}_k\phi(C_t) + \mathbf{b}_k$$

. Процесс предсказания составного объекта представлен на рисунке 1.5

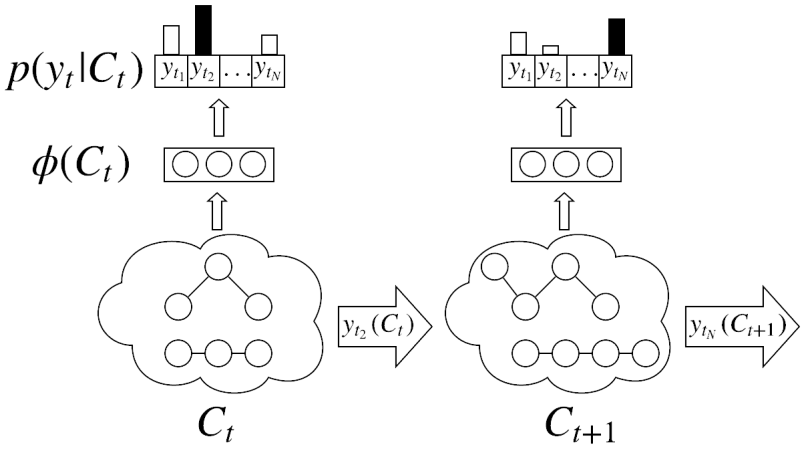


Рисунок 1.5 — Процесс предсказания составного объекта

Существенным недостатком такого представления задачи является предположение о независимости отдельных переходов в $\mathbf{y}$. Следствием этого является эффект распространения ошибки (error propagation), при котором ошибки на первых шагах предсказания сказываются на всем последующем результате. Использование данных методов позволяет улучшить точность предсказаний, однако зачастую приводит к усложнению модели и увеличению времени, затрачиваемого на получение результата. Кроме того, для каждого вида структур требуется создание эффективной процедуры точного вероятностного вывода или его приближения, тогда как сложность вывода в модели на основе системы переходов зависит линейно от размера входа.

В ряде работ [18; 107] отмечается, что использование более качественных признаков в моделях на основе системы переходов позволяет нивелировать разницу в точности предсказаний по сравнению с моделями, полагающимися на дорогостоящие процедуры поиска оптимального предсказания. Это делает привлекательным применение нейронных сетей в такого рода моделях для автоматического получения сложных признаков на основе данных.

Таким образом, успешное обучение и применение модели на основе системы переходов зависит следующих факторов:

1) степень соответствия каждого элемента (C,Y,$A(C)$) характерным для решаемой задачи структурным зависимостям;

2) выбор параметрической модели $p_\theta(y_t|C_t)$ и метода оценки её параметров;

3) выбор функции извлечения признаков конфигурации $\phi(C_t)$.

Выводы к первой главе

1. Сопровождение продукта после выхода на рынок является самым длинным этапом в рамках жизненного цикла, в ходе которого пользователи могут сталкиваться с различными трудностями при эксплуатации продукта, и производителю необходимо оперативно на них реагировать. Кроме непосредственного общения с пользователями о качестве товаров и услуг, дополнительным источником обратной связи являются тексты отзывов, оставленные ими в интернете. Характерными чертами данного типа текстов является наличие опечаток и орфографических ошибок, игнорирование знаков препинания, употребление сленговых выражений, фразеологизмов и сарказма. Ручной анализ большого количества доступных текстов представляет собой нетривиальную задачу, поэтому задача автоматизации такого анализа является актуальной для бизнес-сообщества.

2. В научной литературе обработке субъективной информации (в том числе и мнений) текстов на естественном языке посвящена область анализа тональности, в рамках которой представлены различные методы обработки текстов, различающихся уровнем детальности проводимого анализа и используемых моделей. Наиболее перспективной с точки зрения анализа пользовательских мнений является группа методов детального анализа тональности, которые позволяют анализировать отношение к продуктам, их составным частям и характеристикам. Однако современные методы анализа либо не учитывают структурные взаимосвязи между извлекаемыми сущностями, либо имеют низкую точность их извлечения.

3. Перспективным направлением решения различных прикладных задач обработки текстов на естественном языке является развитие методов на основе нейронных сетей. Это обусловлено успехами использования данного класса моделей при решении разнообразных задач обработки текстов на естественном языке, в том числе для анализа пользовательских мнений о продуктах и услугах.

4. При решении задач обработки естественного языка методами машинного обучения, в которых в качестве объекта предсказания выступают сущности с нетривиальной структурой, широкое распространение получили модели на основе системы переходов. Их использование позволяет декомпозировать процесс получения финального предсказания на независимые шаги, что дает возможность использовать при работе с объектами сложной структуры стандартный аппарат вероятностных моделей машинного обучения, в том числе и нейронных сетей.

5. Учитывая лингвистические особенности текстов на естественном языке о продуктах и услугах, для решения задачи извлечения из этих текстов структурированной информации о мнениях, запросах и пожеланиях потребителей, предлагается использовать подход на основе системы переходов, использующий рекуррентные и свёрточные нейронные сети для получения признаков из исходного текста.

Глава 2. Нейросетевые модели на основе системы переходов для извлечения структурированной информации о продуктах из текстов пользователей

2.1 Нейросетевая модель для извлечения составных объектов и их атрибутов из текстов на естественном языке

2.1.1 Архитектура нейронной сети и алгоритм извлечения составных объектов

Под составным объектом в тексте, представленным в виде последовательности слов $\mathbf{w} = (w_1,...,w_N)$, будем понимать пару $o = (SP, R)$, состоящую из множества фрагментов SP и множества связей между фрагментами R. Каждый элемент множества SP определяет некоторую часть составного объекта, представленную непрерывной последовательностью слов в тексте (фрагментом). Каждый из фрагментов в зависимости от специфики конкретной предметной области характеризуется определенной ролью в объекте. Для обозначения роли предлагается ввести множество типов фрагментов Lbl. Таким образом, фрагмент определяется парой $sp = (w_j, w_{j+1}, ..., w_{j+m}, label)$, где j – индекс начала фрагмента в тексте $\mathbf{w}$, m – длина фрагмента (в словах), $lbl \in Lbl$ – метка типа фрагмента. Элементы множества R задают структуру объекта в виде набора отношений между парами фрагментов (sp_1, sp_2). Дополнительно расширить смысловое наполнение объекта можно, дополнив каждое отношение набором атрибутов из множества $A = \{a_1, a_2,...,a_{n_A}\}$, где для каждого a определено множество значений атрибута $V(a) = \{v_1^a, ..., v_{n_{V(a)}}\}$. Таким образом, отношение задано тройкой следующего вида: $r = (sp_1, sp_2, av = \{(a_k, v_m^{a_k}, (a_p, v_g^{a_p}))\})$, $(\forall k,p : a_k \neq a_p, \text{если } k \neq p)$. Пример формальной структуры составного объекта приведен на рисунке 2.1.

Извлечение составных объектов с описанной структурой предлагается осуществлять с использованием нейросетевой модели на основе системы переходов [21; 25] со структурой, заданной кортежем $(C,Y,A(C))$, где C – конфигурация модели, Y – множество переходов, изменяющих конфигурацию,

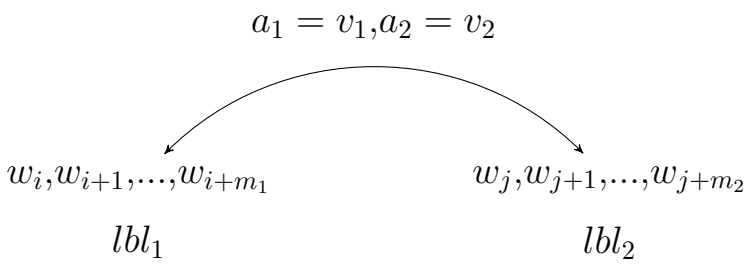


Рисунок 2.1 — Пример формальной структуры составного объекта

$A(C)$ – функция, задающая множество переходов, доступных для исполнения в текущей конфигурации.

Конфигурация модели задается кортежем $C = (B, S, L, H)$. Список B содержит в себе все необработанные на данный момент слова исходного текста. Во время работы допускается извлекать слова из начала списка до тех пор, пока он станет пустым. Список S содержит извлеченные на текущий момент составляющие объект фрагменты. Во время работы новые фрагменты добавляются в конец списка. Фрагмент в конце списка может быть сформирован не полностью и модифицироваться во время работы. Это необходимо для инкрементального построения фрагмента за несколько переходов. Фрагменты на последующих позициях должны быть полностью определены и вносить в них изменения не допускается. Список L хранит отношения между фрагментами из S и их атрибуты. Отношение в конце списка может быть определено не полностью. Элементы L могут ссылаться на одни и те же фрагменты из S несколько раз. Допускается создание петель – отношений, в которых началом и концом связи является один и тот же фрагмент. В то же время в S могут присутствовать элементы, не связанные ни с каким другим. История совершенных переходов H используется в роли дополняющего элемента, реализующего механизм памяти о принятых на прошлых шагах решениях.

Начальную и целевую конфигурации предлагается определить следующим образом:

$$C_0 = ((w_1, \ldots, w_N), \varnothing, \varnothing, \varnothing),$$
$$C_T = (\varnothing, (s_1, \ldots, s_{|S|}), (l_1, \ldots, l_{|L|}), (y_1, \ldots, y_T))$$

Множество доступных переходов задается в следующем виде:

$$Y = \{Shift, Start(lbl), Add(lbl), Link(n_1, n_2), Attr(a, v), End\}.$$

Опишем подробнее каждый переход и соответствующее ему изменение конфигурации C:

– $Shift$ извлекает из списка B первый элемент;

– $Start(e)$ определяет начало нового фрагмента с меткой типа e и помещает его в конец S. При этом элемент в начале B извлекается и становится первым словом в новом фрагменте;

– $Add(e)$ извлекает элемент из начала списка B и помещает его в находящийся на вершине стека S фрагмент;

– $Link(n_1,n_2)$ образует отношение между элементами S, находящиеся на позициях n_1 и n_2, оно добавляется в конец списка L;

– $Attr(a,v)$ присваивает атрибуту a значение v в отношении, находящейся в конце L;

– End заканчивает формирование предсказания.

Изменения, вносимые в конфигурацию C_t при совершении определенного перехода, представлены в таблице 2.1. Следует отметить, что способ задания множества Y накладывает ограничение на максимальное расстояние между фрагментами в S, которое может предсказать модель.

Таблица 2.1 — Изменение конфигурации при выполнении переходов

t	y_t	$t+1$
$w\|B$	$Shift$	B
$w\|B;S$	$Start(lbl)$	$B;S\|(\{w\};lbl)$
$w\|B;S\|(\{...,x\};lbl)$	$Add(lbl)$	$B;S\|(\{...,x,w\};lbl)$
L	$Link(n_1,n_2)$	$L\|(S_{n_1},S_{n_2},\varnothing)$
$L\|(S_{n_1},S_{n_2},\{...,(a',v')\})$	$Attr(a,v)$	$L\|(S_{n_1},S_{n_2},\{...,(a',v'),(a,v)\})$

Функция $A(C)$, определяющая допустимость совершения переходов в конкретной конфигурации, задана в виде набора условий и представлена в виде таблицы 2.2. Функция $A(C)$ задает недетерминированный конечный автомат, распознающий последовательности переходов, выполнение которых приводит к формированию составных объектов на выходе модели. Множество состояний автомата состоит из начального состояния $None$, конечного состояния End и набора состояний, соответствующих типам фрагментов из множества Lbl. На рисунке 2.2 приведен пример автомата для случая с двумя типами распознаваемых фрагментов.

Пример составного объекта в тексте и соответствующая ему последовательность переходов представлены на рисунке 2.3. Преобразования множества

Таблица 2.2 — Условия допустимости совершения переходов

Переход	Условие
$Shift$	$B \neq \varnothing$
$Start(lbl)$	$B \neq \varnothing$
$Add(lbl)$	$B \neq \varnothing \wedge S \neq \varnothing \wedge type(S_{-1}) = lbl$
$Link(n_1,n_2)$	$\exists S_{n_1} \wedge \exists S_{n_2} \wedge (n_1,n_2) \notin L$
$Attr(a,v)$	$L \neq \varnothing \wedge \forall v \nexists (a,v) \in L_{-1}$
End	$B = \varnothing$

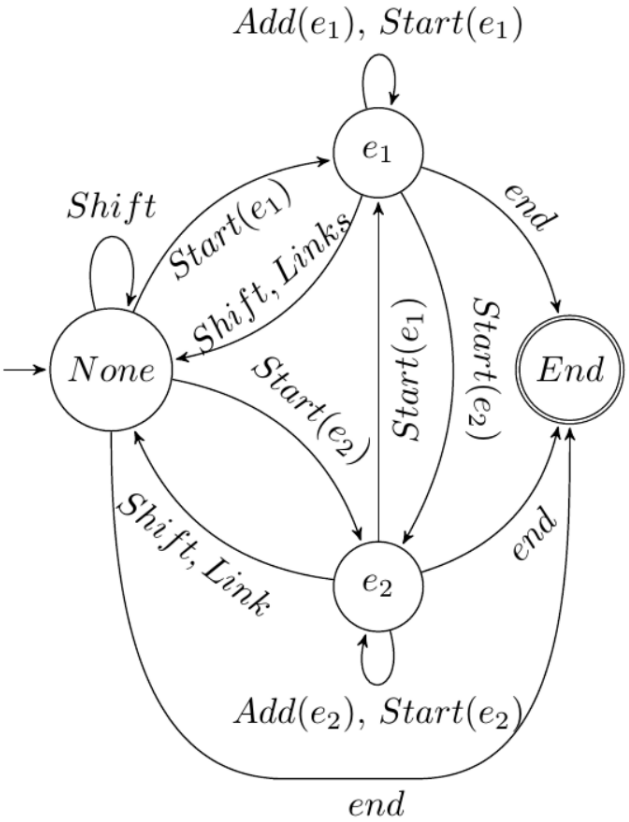
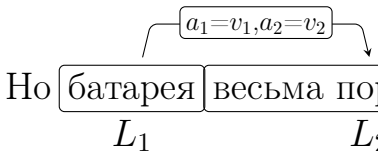


Рисунок 2.2 — Пример абстрактного автомата модели извлечения

составных объектов в тексте $\{o_j\}$ в последовательность действий $(y_1,y_2,...,y_T)$ и соответствующее ему обратное преобразование, необходимые для совершения предсказаний и обучения модели, описаны в алгоритмах 1 и 2.

Тогда процесс извлечения составных объектов и их атрибутов может быть представлен следующей последовательностью шагов:

Шаг 1 Определить на основе исходного текста **w** начальную конфигурацию C_0, задать $t = 1$.



$$\mathbf{y} = (Shift, Start(L_1), Start(L_2), Add(L_2), Link(1,2), Attr(a_1, v_1),$$

$$Attr(a_2, v_2), Shift, Shift, Shift, End)$$

Рисунок 2.3 — Составной объект в тексте и соответствующая ему последовательность переходов

Алгоритм 1: Преобразование множества составных объектов в последовательность переходов

Вход: Текст (w_1,w_2,...,w_N) и составные объекты $\{o_j\}$
Выход: Последовательность переходов ($y_1, y_2, ..., y_T$)
for $k \leftarrow 1$ **to** N **do**
  $trans_k \coloneqq (Shift)$;
end
$idxs \coloneqq$ *Номера фрагментов в порядке упоминания*;
foreach (I,R) $\in \{o_j\}$ **do**
  foreach $(i_1, i_2, AV) \in R$ **do**
    if $idxs(i_1)$ *не вставлен* **then** InsertSpan($trans, idxs(i_1), |i_1|, type(i_1)$);
    if $idxs(i_2)$ *не вставлен* **then** InsertSpan($trans, idxs(i_2), |i_2|, type(i_2)$);
    $i_r = \arg\max_{i \in \{i_{from}, i_{to}\}} idxs(i)$;
    Append($trans_{idxs(i_r)}, (Link(idxs(i_r) + 1 - idxs(i_{from}), idxs(i_r) + 1 - idxs(i_2)))$);
    foreach (a,v) $\in AV$ **do**
      Append($trans_{idxs(i_r)}, Attribute(a, v)$);
    end
  end
end
Раскрыть *trans* в последовательность $\{y_1, y_2, ..., y_T\}$;

Шаг 2 Рассчитать набор признаков конфигурации $\phi(C_{t-1})$.

Шаг 3 Подать набор признаков в классификатор f и предсказать наиболее вероятный переход с учетом условий допустимости из 2.2: $\hat{y}_t = f(\phi(C_{t-1}))$. Если $\hat{y}_t = End$, то $t = t + 1$ и переход на шаг 5.

Шаг 4 Получить конфигурацию C_t из C_{t-1} в соответствии с предсказанным переходом по правилам из таблицы 2.1 и перейти на шаг 2.

Алгоритм 2: Преобразование последовательности переходов в множество составных объектов

Вход: Последовательность переходов $(y_1, y_2, ..., y_T)$
Выход: Множество составных объектов $\{o_j\}$
$p \coloneqq 1, span \coloneqq \varnothing, s\_stack \coloneqq \varnothing, l\_stack \coloneqq \varnothing$;
foreach $y \in (y_1, y_2, ..., y_T)$ **do**
  switch y **do**
    case $Shift$ **do**
      if $span \neq \varnothing$ **then** Push($s\_stack, span$);
      $span \coloneqq \varnothing$;
    case $Start(lbl)$ **do**
      if $span \neq \varnothing$ **then** Push($s\_stack, span$);
      $span \coloneqq (p; lbl)$;
    case $Add(lbl)$ **do**
      Append($span, p$);
    case $Link(i_1, i_2)$ **do**
      Push($l\_stack, (s\_stack_{i_1}, s\_stack_{i_2}, \varnothing)$)
    case $Attr(a,v)$ **do**
      Добавить (a,v) *в* Top(l_stack);
    case End **do** ;
  end
  if $y \in \{Shift, Start(lbl), Add(Label)\}$ **then** $p \coloneqq p + 1$;
end
Объединить $l_1, l_2 \in l\_stack$ в объект o_j, если у l_1 и l_2 есть общий фрагмент.

Шаг 5 Преобразовать полученную последовательность переходов $\hat{\mathbf{y}}$ в составной объект.

В данной работе в качестве классификатора $f(\phi(C_t))$ предлагается использовать параметрический вероятностный классификатор:

$$\hat{y}_t = f(\phi(C_t)) = \arg\max_{y \in A(C_t)} p_\theta(y|\phi(C_t)).$$

Ключевой задачей при формировании модели на основе системы переходов является выбор способа параметризации классификатора $p_\theta(y_t|\phi(C_t))$ и задание функции извлечения признаков их конфигурации $\phi(C_t)$. Данное распределение предлагается задать следующим образом:

$$p_\theta(\hat{y}_t|\phi(C_t)) = \text{softmax}_{\hat{y}_t}(\mathbf{W}\phi(C_t) + \mathbf{b}). \tag{2.1}$$

Сложность выбора конкретного вида функции извлечения признаков $\phi(C_t)$ при работе с текстами заключается в необходимости представить кон-

фигурацию переменного и потенциально неограниченного размера вектором с конечным числом элементов.

Вектор признакового описания конфигурации $\phi(C_t)$ определяется как конкатенация признаковых описаний частей конфигурации B, S и H на шаге t:

$$\phi(C_t) = [\phi(B_t); \phi(S_t); \phi(H_t)] \tag{2.2}$$

Основой для получения признаковых описаний B и S служат контекстные векторные представления входной последовательности слов, помещенные в список B:

$$\mathbf{h}_i^B = F(E(w_1, w_2, ..., w_N), i) \tag{2.3}$$

В качестве преобразования F в работе предлагается использовать свёрточную (1.1) или рекуррентную (1.3 и 1.4) нейронную сеть. Вектор признакового описания списка слов $\phi(B_t)$ образован конкатенацией первых n_B контекстно-зависимых признаков элементов списка B на шаге t:

$$\phi(B_t) = \left[\mathbf{h}_{t(1)}^B; \mathbf{h}_{t(2)}^B; ...; \mathbf{h}_{t(n_B)}^B\right], \tag{2.4}$$

где $t(i)$ – позиция i-го элемента B в последовательности представлений h^B на шаге t.

Формирование $\phi(B_t)$ во времени можно представить как перемещение окна фиксированного размера n_B по последовательности h^B слева направо и конкатенация её элементов в данном окне. Для корректного формирования $\phi(B_t)$ в конец h^B добавляются $(n_B - 1)$ специальных замещающих векторов. Процесс формирования данного вектора признаков во времени для случая $n_B = 3$ показан на рисунке 2.4. Подобное решение позволяет модели работать с n-грамами слов и лучше обрабатывать длинные фразы.

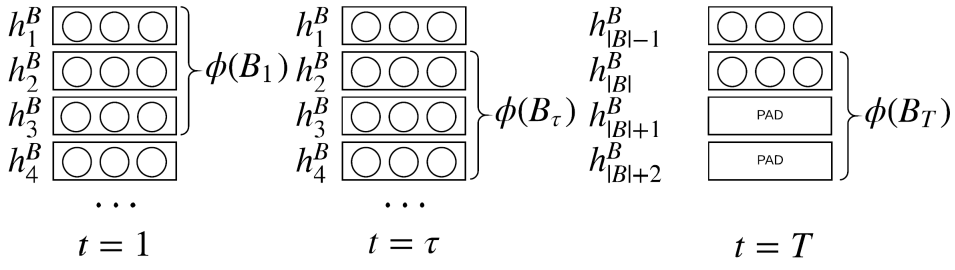


Рисунок 2.4 — Пример формирования $\phi(B_t)$

Для формирования вектора признакового описания списка извлечённых сущностей $\phi(S_t)$ сначала рассчитываются признаки индивидуальных элементов h^S по формуле:

$$\mathbf{h}_i^S = \mathbf{W}\left[\left(\frac{1}{e(i)-b(i)}\sum_{j=b(i)}^{e(i)}\mathbf{h}_j^B\right); E^{Label}(i)\right] + \mathbf{b}, \tag{2.5}$$

где $b(i)$ и $e(i)$ – индексы, соответствующие началу и концу i-го фрагмента текста в последовательности $\mathbf{h}^B$, $E^{Label}(i)$ - векторное представление типа i-го фрагмента. Затем n_S последних элементов последовательности конкатенируются, образуя вектор $\phi(S_t)$:

$$\phi(S_t) = \left[\mathbf{h}_{-1}^S; \mathbf{h}_{-2}^S; ...; \mathbf{h}_{-n_s}^S\right]. \tag{2.6}$$

Для формирования вектора признаков истории совершенных переходов $\phi(H_t)$ используется скрытое состояние последнего шага сети LSTM, примененной к последовательности векторных представлений действий:

$$\phi(H_t) = \text{LSTM}(E^{Act}(H_1),..., E^{Act}(H_t))_t \tag{2.7}$$

С учетом вышеизложенного, архитектура нейронной сети может быть представлена в виде диаграммы, представленной на рисунке 2.5.

Последовательность шагов для извлечения составных объектов и их атрибутов в соответствии с предложенной моделью (2.1—2.7) представлена в алгоритме 3. Схематичное изображение процесса предсказания показано на рисунке 2.6

2.1.2 Процедура обучения модели

Обучение предложенной модели предлагается осуществлять при помощи метода максимального правдоподобия. Пусть дана обучающая выборка, элементами которой являются пары из текста и содержащееся в нем множество составных объектов $D = \{(\mathbf{w}_i, \{o_j\}_i)\}$. Каждому $\{o_j\}_i$ соответствует истинная последовательность переходов $\mathbf{y} = (y_1, y_2, ..., y_T)$, полученная с помощью алгоритма 1. Имея на входе C_0, модель предсказывает последовательность оценок

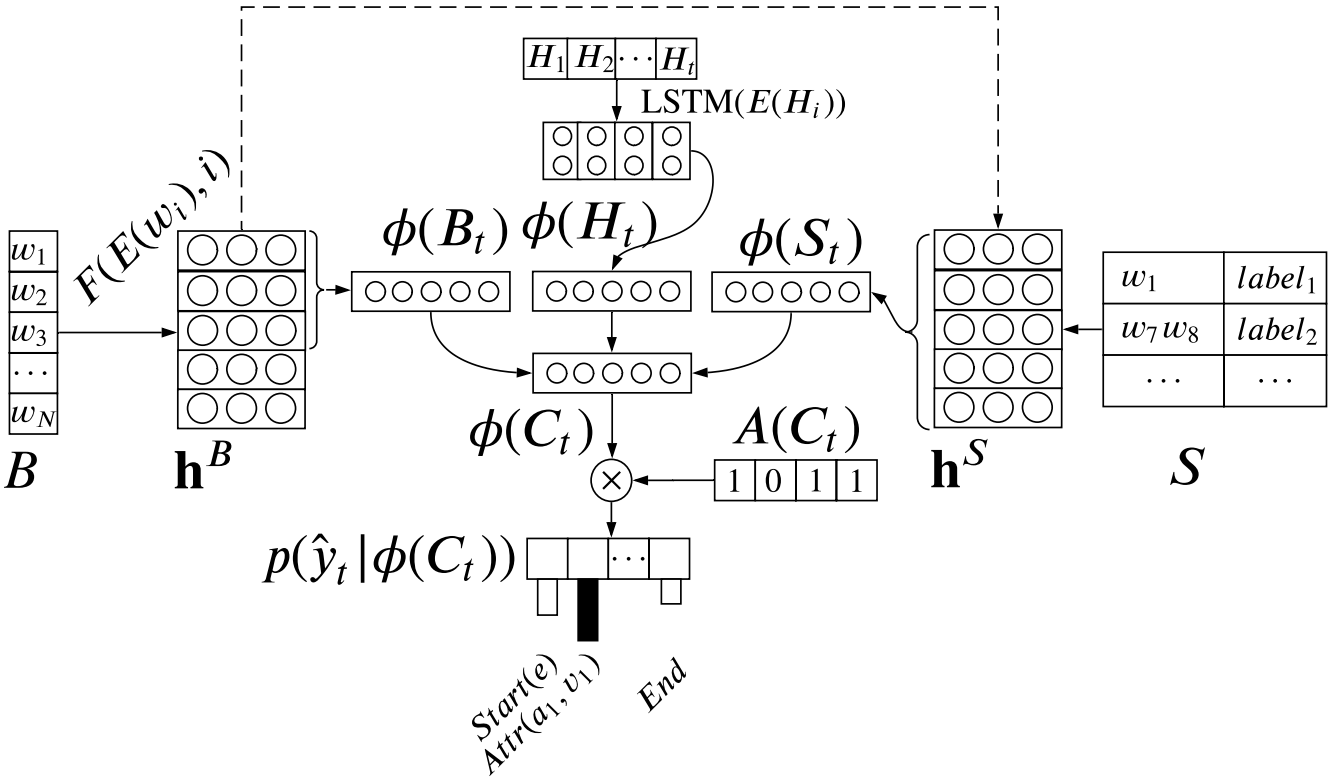


Рисунок 2.5 — Архитектура нейросетевой модели на основе системы переходов

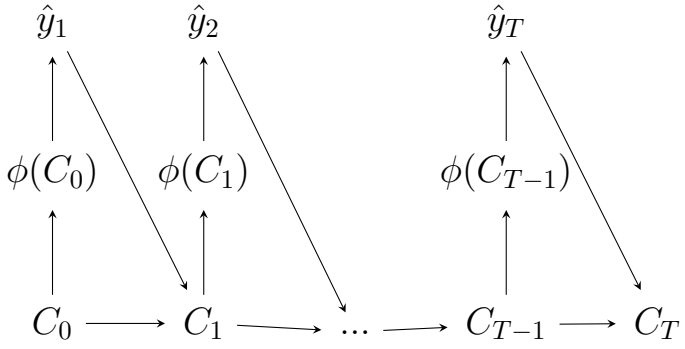


Рисунок 2.6 — Процесс получения предсказания

распределения вероятности совершения переходов согласно алгоритму 3:

$$\hat{\mathbf{p}} = (p_\theta(\hat{y}_1|C_0), p_\theta(\hat{y}_2|C_1^{\hat{y}_1}), ..., p_\theta(\hat{y}_{\bar{T}}|C_{\bar{T}-1}^{\hat{y}_{\bar{T}-1}})).$$

Тогда функцией ошибки для одного примера является сумма перекрестной энтропии между истинным переходом и распределением, предсказанным моделью, для каждого шага предсказания:

$$\mathcal{L}(\mathbf{y}, \hat{\mathbf{p}}) = \sum_{t-1}^{T} \mathcal{L}_t(y_t, \hat{\mathbf{p}}_t) = \sum_{t=1}^{T} -\log p_\theta(\hat{y}_t|C_{t=1})_{y_t}. \quad (2.8)$$

Алгоритм 3: Алгоритм предсказания последовательности переходов `Forward`

Вход: Текст (w_1,w_2,...,w_N)

Выход: Последовательность предсказанных переходов $(\hat{y}_1, \hat{y}_2, ..., \hat{y}_T)$

$B \coloneqq (w_1, w_2, ..., w_N)$, $S \coloneqq \varnothing$, $L \coloneqq \varnothing$, $H \coloneqq \varnothing$;

Рассчитать $\mathbf{h}_i^B$ согласно (2.3);

$t \coloneqq 0$;

repeat

  $\phi(C_t) = [\phi(B_t); \phi(S_t); \phi(H_t)]$;

  $p(\hat{y}_{t+1}|C_t) \coloneqq \text{softmax}(\mathbf{W}\phi(C_t) + \mathbf{b})$;

  $\hat{y}_{t+1} \coloneqq \arg\max_{\hat{y}_t \in A(C_t)} p(\hat{y}_{t+1}|C_t)$;

  Изменить C в соответствии с правилами в таблице 2.1;

  $t \coloneqq t + 1$;

until $\hat{y}_t \neq End$;

Полная ошибка по обучающему набору выглядит следующим образом:

$$\mathcal{L}_D = \sum_{\mathbf{y}, \hat{\mathbf{p}} \in D} \mathcal{L}(\mathbf{y}, \hat{\mathbf{p}}) \tag{2.9}$$

Обучение модели происходит путем минимизации выражения 2.9 с помощью комбинации процедур обратного распространения ошибки [108] и градиентной оптимизации. Процедура обратного распространения ошибки необходима для определения градиента функции ошибки относительно множества параметров нейронной сети:

$$\nabla_\theta \mathcal{L} = \frac{\partial \mathcal{L}}{\partial \theta} = \frac{\partial \mathcal{L}_1}{\partial \theta} + ... = \frac{\partial \mathcal{L}_1}{\partial \hat{\mathbf{p}}_1} \cdot \frac{\partial \hat{\mathbf{p}}_1}{\partial \theta} + ...$$

После рассчета $\nabla_\theta \mathcal{L}$ производится шаг оптимизации весов. Обычно используется метод градиентного спуска, который смещает параметры модели в сторону анти-градиента с некоторым малым шагом α:

$$\theta_{iter+1} = \theta_{iter} - \alpha \nabla_\theta \mathcal{L}.$$

Как правило, вместо вычисления $\nabla_\theta \mathcal{L}$ на всем наборе D используют приближенную оценку $\tilde{\nabla}_\theta \mathcal{L}$, рассчитанную на случайной малой части примеров – мини-батче (англ. minibacth). В таком случае говорят о стохастическом градиентном спуске. В научной литературе предложено большое количество методов,

развивающих идеи стохастического градиентного спуска. Метод адаптивной градиентной оптимизации Adam [109] получил широкое распространение в практике благодаря высокой скорости сходимости и низкой чувствительности к выбору начальных параметров. Он будет использоваться в последующих экспериментах.

Важным вопросом при обучении любой авторегрессионой модели является способ, которым информация о решении с предыдущего шага передается на текущий. Возможны два базовых подхода. Пусть C_t^y обозначает, что переход в конфигурацию был совершен при выполнении действия y. Тогда в первом подходе на шаг $t+1$ передается предсказание, совершенное нейронной сетью на шаге t, т. е.:

$$\hat{y}_t = \underset{y_t \in A(C_{t-1})}{\arg\max}\, p_\theta(y_t | C_{t-1}),$$

$$\hat{\mathbf{p}} = (p_\theta(y_1|C_0), p_\theta(y_2|C_1^{\hat{y}_1}), ..., p_\theta(y_T|C_{T-1}^{\hat{y}_{T-1}})).$$

Схематичное изображение распространения градиента по развернутой структуре сети приведено на рисунке 2.7. На практике данный способ применяется редко. В самом начале обучения предсказания модели носят случайный характер, что приводит к тому, что параметры модели изменяются в соответствии с ошибочной последовательностью предсказаний и соответствующих скрытых состояний.

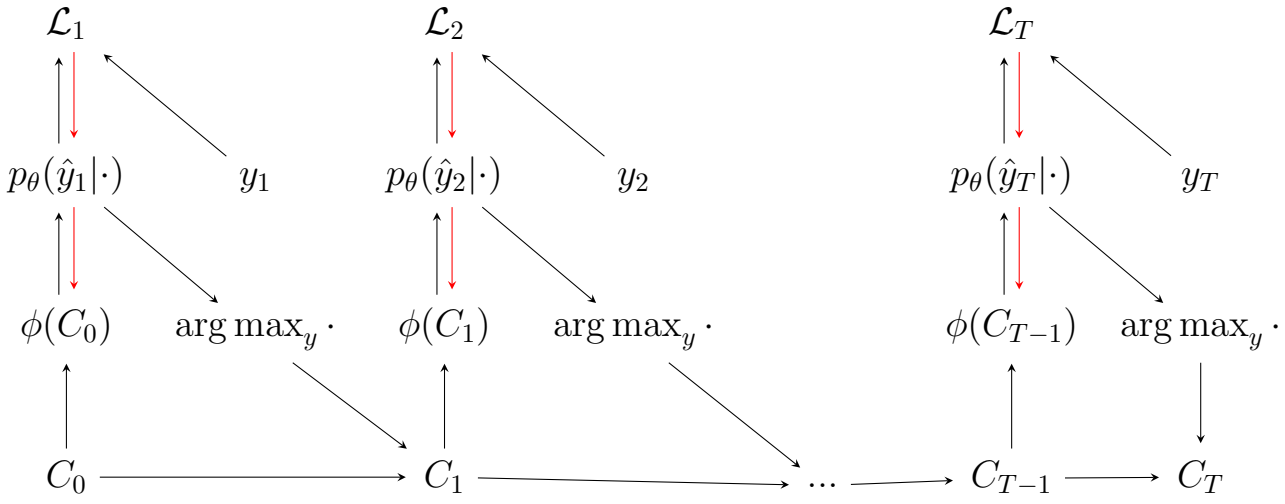


Рисунок 2.7 — Обучение модели без форсирования учителя

Второй подход получил название «форсирование учителя» (англ. teacher forcing) [110]. Он заключается в том, что на шаг $t+1$ вместо предсказания $\hat{y}_t$

подается y_t из истинной последовательности меток $\mathbf{y}$:

$$\hat{\mathbf{p}} = (p_\theta(y_1|C_0), p_\theta(y_2|C_1^{y_1}), ..., p_\theta(y_T|C_{T-1}^{y_{T-1}})).$$

Это позволяет оптимизировать критерий условного максимального правдоподобия напрямую, что стабилизирует динамику обучения по сравнению с первым подходом [110]. Однако при такой процедуре обучения модель никогда не попадает в «ошибочные» состояния и не будет иметь возможности научиться компенсировать вредоносный эффект отдельных неверно предсказанных переходов на все предсказание в целом [111]. В научной литературе предложено множество методов, позволяющих соблюсти баланс между сходимостью обучения и возможностью исследовать пространство ошибочных состояний: специальные процедуры семплировния [111], динамические оракулы для построения последовательностей переходов [112; 113].

В данной работе во всех экспериментах используется форсирование учителя. Хотя такой выбор потенциально ограничивает точность модели, он прост в реализации и достаточен для обеспечения точности, превышающей другие рассмотренные модели. Исследование альтернативных схем обучения оставлено автором для последующих работ.

Процедура обучения модели представлена в алгоритме 4. Схематичное изображение распространения градиента по развернутой структуре сети для одного обучающего примера приведено на рисунке 2.8. Черными стрелками обозначен прямой ход вычислений, красными – направления обратного распространения ошибки.

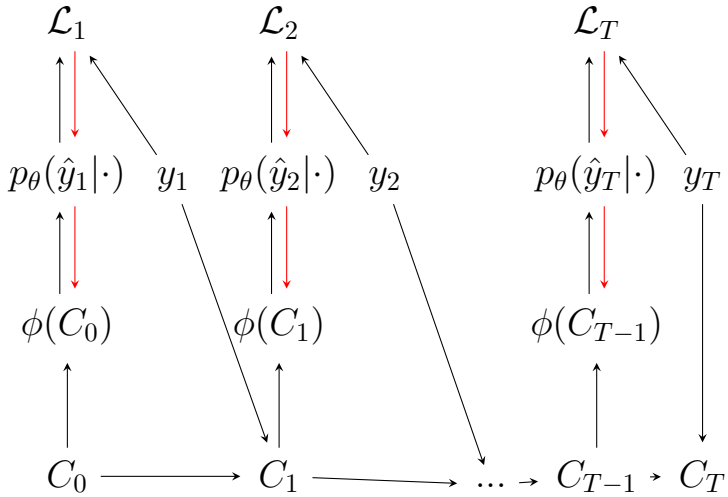


Рисунок 2.8 — Обучение модели с использованием форсирования учителя

Алгоритм 4: Процедура обучения модели

Вход: Обучающая выборка $D = \{\mathbf{w}_i, (\mathbf{y}_i)\}$

Выход: Множество параметров $\hat{\theta}$

$t := t + 1$;

Задать начальное приближение θ_t случайным образом ;

repeat

$\mathbf{w}_i, \mathbf{y}_i \sim D$;

Получить последовательность $\hat{\mathbf{p}}$ согласно алгоритму 3;

$l = \sum_{t-1}^{T} \mathcal{L}_t(y_t, \hat{\mathbf{p}}_t)$;

$\nabla_\theta \mathcal{L}(\mathbf{y}, \hat{\mathbf{p}}) = \text{BackPropagate}(l, \theta)$;

$\theta_{t+1} = \theta_t - \alpha \nabla_\theta \mathcal{L}(\mathbf{y}, \hat{\mathbf{p}})$;

$t := t + 1$;

until *критерий останова выполнен*;

2.1.3 Выбор и обоснование метрик оценки качества модели

При обучении модели, настройке её гиперпараметров и сравнении её с альтернативами необходимо оценивать качество извлечения составных объектов. Следуя работам [11; 63], предлагается использовать меру F_1 для оценки качества извлечения отдельных составляющих объектов: фрагментов, отношений между ними, атрибутов. Мера F_1 рассчитывается следующим образом:

$$F_1 = \left(\frac{R^{-1} + P^{-1}}{2} \right)^{-1} = 2 \frac{R \cdot P}{P + R} \tag{2.10}$$

Далее необходимо определить конкретный способ расчета P и R в зависимости от особенностей решаемой задачи. Например, при распознавании именованных сущностей принято определять полное соответствие фрагментов истинной и обнаруженной сущности. Однако авторы [114] отмечают, что однозначно определять границы фрагментов в задаче определения мнений затруднительно даже для экспертов, поэтому оценка с помощью полного соответствия фрагментов будет неоправданно занижать качество модели. Поэтому ими предлагается «мягкий» подход к оценке, который позволяет учитывать частичные совпадения фрагментов.

Пусть sp и $\hat{sp}$ – истинный и предсказанный моделью фрагменты, $|sp|$ – количество слов во фрагменте, $sp \cap \hat{sp}$ – фрагмент, состоящий из общих для

sp и $\hat{sp}$ слов. Тогда степень покрытия фрагмента $\hat{sp}$ фрагментом sp будет определяться следующим образом:

$$c(sp, \hat{sp}) = \frac{|s \cap \hat{s}|}{|\hat{s}|} \tag{2.11}$$

На основе этого степень покрытия межу двумя множествами фрагментов определяется следующим образом:

$$C(SP,\hat{SP}) = \sum_{sp_i \in SP} \sum_{\hat{sp}_j \in \hat{SP}} c(sp_i, \hat{sp}_j)$$

Тогда P и R будут определяться следующим образом:

$$P^{span}(SP,\hat{SP}) = \frac{C(SP, \hat{SP})}{|\hat{SP}|}, R^{span}(SP,\hat{SP}) = \frac{C(\hat{SP}, SP)}{|SP|},$$

В данной работе при оценке качества извлечения фрагментов мера F_1 будет рассчитываться для каждого типа фрагментов отдельно, выражение 2.11 будет равняться нулю для пересекающихся фрагментов разных типов. Общее качество извлечения фрагментов будем рассчитывать как среднее значение F_1 для каждого типа фрагментов, т.е. посредством макро-усреднения.

Рассчет F_1^{rel} при определении качества извлечения отношений между фрагментами будем производить следующим способом. Пусть RS и $\hat{RS}$ – множества истинных и предсказанных отношений, $left(r)$ и $right(r)$– левостоящий и правостоящий фрагменты в отношении r, $u(sp_1, sp_2) = type(sp_1) = type(sp_2 \wedge |sp_1 \cap sp_2| \geqslant 1$ – индикаторная функция соответствия двух фрагментов. Тогда истинно-положительные (TP), ложно-отрицательные (FN) и ложно-положительные срабатывания будем определять следующим образом:

$$TP^{rel} = |\{(r, \hat{r}) : u(left(r), left(\hat{r})) \wedge u(right(r), right(\hat{r})), r \in R, \hat{r} \in \hat{R}\}|$$
$$FP^{rel} = |\hat{R}| - TP^{rel}$$
$$FN^{rel} = |R| - TP^{rel}$$

Соответственно, P^{rel} и R^{rel} для определения F_1^{rel} рассчитаем следующим образом:

$$P^{rel} = \frac{TP^{rel}}{TP^{rel} + FP^{rel}}$$
$$R^{rel} = \frac{TP^{rel}}{TP^{rel} + FN^{rel}}$$

Рассчет F_1 для значения атрибутов осуществляется схожим образом, однако в качестве множеств R и $\hat{R}$ выступают множества истинных и предсказанных атрибутов. Интегральная оценка качества определения атрибутов также будет определятся посредством макро-усреднения F_1 для каждого атрибута.

Применение описанных модели и алгоритма для решения конкретных прикладных задач требует задания множества допустимых типов фрагментов Lbl, множества атрибутов AV и функции ограничений на допустимые переходы $A(C_t)$ в соответствии со структурой извлекаемых объектов.

2.2 Нейросетевая модель для извлечения и анализа пользовательских мнений из текстов отзывов о потребительских свойствах товаров

2.2.1 Постановка задачи извлечения и анализа пользовательских мнений о потребительских свойствах товаров

Выявление отношения пользователей к приобретаемым товарам (продуктам) и услугам является важной частью работы маркетингового отдела любого бизнеса. Опираясь на полученную информацию, компании могу управлять ассортиментом продукции, учитывать предпочтения отдельных социальных групп при формировании рекламных сообщений, диагностировать возникающие при эксплуатации проблемы и вовремя реагировать на них, использовать негативную обратную связь для формирования видения будущих продуктовых линеек. Доступным источником большого количества исходных данных для подобного анализа являются многочисленные сайты, где потребители могут обсуждать приобретаемые товары, высказывая своё мнение о различных аспектах товара.

В данной работе предлагается извлекать из текстов отзывов отдельные оценочные высказывания покупателей, которые содержали бы в себе основной посыл сообщения, в форме, полезной при решении возникающих на этапе сопровождения продукта задач. Автор исходит из предположения, что основная ценная информация в отзыве представлена в виде эмоциональ-

но окрашенных высказываний о свойствах товаров, представимых в виде пар «аспект–описание». Также имеет смысл определять тональную окраску извлеченных высказываний. Извлеченные высказывания далее по тексту будут называться мнениями. Такая форма обработки обеспечивает извлечение мнений в различных срезах, таких как:

– *тональность*: положительная, отрицательная;

– *уровень детальности анализа*: отдельная характеристика продукта, группа характеристик (например, все про доставку), продукт в целом, категория продуктов, бренд/производитель;

– *объект интереса*: собственные продукты, продукты конкурентов;

– *момент анализа*: в данный момент, динамика изменений настроения пользователей.

Кроме того, извлеченные мнения можно использовать в рекламных целях в качестве социального доказательства (англ. social proof) – материала, направленного на формирование доверия к продукту путем демонстрации преимуществ, обнаруженных потребителями, уже опробовавшими продукт в действии. Данный тип материалов может демонстрироваться на сайте продукта, применяться при составлении рекламных e-mail рассылок.

Анализ смыслового содержания мнений о потребительских свойствах продуктов предлагается провести на базе текстов отзывов, размещенных на сайте интернет-магазина «Ali Express». Магазин предлагает широкий ассортимент товаров из различных категорий и имеет большую пользовательскую базу по всему миру, что позволяет получать большие объемы отзывов о товарах.

На «Ali Express» широко представлены производители относительно недорогих товаров из Китая, не имеющих собственных известных брендов с репутацией, продающие свои продукты напрямую потребителям. Так как данный магазин в большой степени ориентирован на зарубежные рынки, очень важную составляющую в нем играют два взаимосвязанных фактора: логистический и фактор доверия. Логистический фактор определяется скоростью доставки и её качеством, при этом продавец и магазин имеют контроль над процессом только на первых этапах. Некачественное исполнение обязательств третьей стороной (почтовой или курьерской службой, таможней) негативным образом сказывается на отношении покупателей и является серьезным риском. Фактор доверия определяется репутацией продавца и качеством предоставляемых им продуктов. С учётом особенностей «Ali Express», на котором представлено огромное

количество продавцов, предлагающих схожие продукты, доверие и цена являются определяющими факторами, влияющими на вероятность приобретения товара. Магазин предоставляет дополнительные услуги призванные убедить потенциального покупателя в надежности продавца и снять риски с покупателя: гарантирует возврат средств в случае непоставки товара, обеспечивает разрешение споров, поддерживает систему социального рейтинга товаров и продавцов.

Проведенное автором исследование текстов отзывов о товарах показало, что чаще всего в своих отзывах пользователи говорят о трех источниках проблем: логистическом факторе (скорость доставки, качество упаковки, состояние товара), качестве товаров, опыте общения с продавцом. Отчасти наблюдаемое поведение можно объяснить тем, что потребители с осторожностью относятся к покупкам в данном магазине и чаще выбирают дешевые товары без высоких требований к качеству, беспокоясь в большей степени о сроках и качестве доставки. Описание качества, как правило, довольно немногословное и ограничивается общей оценкой товара. Это позволяет сделать вывод о том, что в первую очередь анализ будет полезен маркетологам для улучшения качества доставки и поиска путей повышения доверия у пользователей. Можно сказать, что в будущем ситуация начнет меняться, так как «Ali Express» активно улучшает логистический аспект своего бизнеса. По мере того как проблем с доставкой станет меньше, пользователи начнут больше внимания уделять непосредственно качеству товара.

Для объединения множества типов мнений пользователей в группы с целью выявления наиболее важных направлений развития продукта и сопровождающих сервисов, в которые требуется вложить дополнительные ресурсы, предложен оригинальный классификатор, основанный на элементах универсальной модели деятельности: субъект деятельности, объект на который направлена деятельность, средства используемые в процессе деятельности, взаимосвязи между элементами деятельности.

Применительно к особенностям решаемой задачи субъектом деятельности является продавец, объектом деятельности – подлежащий продаже продукт. К основным средствам можно отнести сайт интернет-магазина, логистические службы, отдел маркетинга. Исходя из предложенных элементов декомпозиции, предлагается ассоциировать с каждым её элементом типы оценочных высказываний, которые пользователи могут использовать для выражения своего мнения при написании текстов отзывов. Выбор типов мнений осуществлялся в соответ-

ствии с критерием информативности/значимости высказывания для решения основных задач, возникающих на этапах эксплуатации и сопровождения продуктов, представленных в главе 1: технической поддержки, модификации продукта и модификации комплекса маркетинговых мероприятий. Полученный иерархический классификатор типов мнений пользователей приведен на рисунке 2.9. Опишем элементы классификатора более детально.

Потребительские свойства товара: мнения о товаре в целом и его эксплуатационных характеристиках: качестве исполнения, выполнении заявленных функций, внешнем вид товара.

Потребительские свойства товаров конкурентов: мнения этого типа в целом аналогичны классу «Потребительские свойства товара», однако в них речь идет о характеристиках товаров конкурентов, упоминаемых потребителями в отзывах.

Качество обслуживания: мнение потребителей о том, как продавец учитывает пожелания, если это применимо к конкретным товарам. В эту же категорию можно отнести упоминания о готовности продавца делать специальные предложения: скидки, бесплатную доставку, подарки.

Способность разрешать конфликтные ситуации: оценка потребителем того, как продавец ведет себя в случае конфликта интересов: при обнаружении брака, возникновении проблем при доставке товара и т. д.

Гарантийное обслуживание: оценка справедливости гарантийных сроков и условий обслуживания, качество и оперативность ремонта товаров.

Оплата: форма и качество процедуры оплаты товаров и/или услуг.

Доставка: скорость отправки товара, качество упаковки товара, целостность товара после прибытия. В эту группу включаются только те аспекты доставки, на которые может повилять продавец.

На основе проведённого анализа текстов отзывов о продуктах пользователей интернет-магазина «Ali Express» для организации разметки данных и проведения экспериментального исследования были выбраны наиболее часто встречающиеся типы мнений и объединены в следующие классы:

1) товар (потребительские качества товара);

2) продавец (общее качество обслуживания, способность решать конфликтные ситуации);

3) доставка (скорость и качество доставки).

Приведем примеры высказываний из трех выделенных классов мнений:

Рисунок 2.9 — Классификатор типов мнений пользователей

Товар: «Ноут хороший, винду переустановил легко, все работает», «Омрачило покупку засветит по верхним углам монитора», «Качество очень порадовало. Размер подошёл», «Все по размеру, прошиты хорошо!», «Без запаха, нитки нигде не торчат».

Продавец: «Продавец очень внимательный и ответственный, смело заказывайте», «Продавца рекомендую», «Спасибо большое продавцу», «Спасибо продавцу! В подарок плёнка на экран», «Продавец довольно общительный. Отвечает на все вопросы».

Доставка: «Отправил через три дня после оплаты заказа», «Трек отслеживался», «Упакован хорошо», «Упакована в вакуумный пакет, сверху упаковочная пленка и все это в почтовом пакете», «Упаковка на высшем уровне», «Все хорошо упаковано в дутый пакет, стекло и доп. чехол тоже в отличном состоянии», «Спасибо продавцу отправил в тот же день».

С учетом вышеупомянутого задача извлечения и анализа мнений пользователей о товаре может быть представлена в следующем виде. Пусть задано множество текстов отзывов покупателей T, содержащих мнения из множества O о наборе продуктов P. Мнением будем называть четверку (Аспект, Описание, Тональность, Тип). Аспектом будем называть последова-

тельность слов, обозначающую важную характеристику или упоминание продукта. Описанием будем называть последовательность слов, содержащую высказанное пользователем мнение о некотором аспекте. Тональность зададим категориальной шкалой из трех уровней: положительная(+), нейтральная(0), негативная(−). Множество типов мнений зададим в соответствии с введенным иерархическим классификатором, однако в целях формирования обучающего набора данных и практической апробации будем использовать укрупненные классы: {Товар, Продавец, Доставка}.

Имея исходные множества T, P, O и заданную структуру классификатора мнений, необходимо извлечь из новых текстов отзывов $\hat{T}$ множество мнений пользователей $\hat{O}$ о новом наборе товаров $\hat{P}$. Для решения этой задачи необходимо адаптировать описанную ранее нейросетевую модель (2.1—2.7) путем определения состава множеств Lbl, A, $V(a)$ согласно смысловому наполнению введенного выше понятия мнения [19; 22; 23]:

$$
\begin{aligned}
&Lbl = \{\text{Аспект}, \text{Описание}\},\\
&A = \{\text{Тональность}, \text{Цель}\},\\
&V(\text{Тональность}) = \{+,0,-\},\\
&V(\text{Цель}) = \{\text{Товар}, \text{Продавец}, \text{Доставка}\}.
\end{aligned}
$$

В функцию $A(C_t)$ необходимо добавить новое условие, разрешающее строить связь только в тех случаях, когда она объединяет аспект и описание. Исходя из данной формализации, множество переходов определено следующим образом:

$$
\begin{aligned}
Y = \{&Shift, Start(\text{Аспект}), Start(\text{Описание}),\\
&Add(\text{Аспект}), Add(\text{Описание}), Link(n_1,n_2), End,\\
&Attr(\text{Тональность},+), Attr(\text{Тональность},0),\\
&Attr(\text{Тональность},-), Attr(\text{Цель},\text{Товар}),\\
&Attr(\text{Цель},\text{Продавец}), Attr(\text{Цель},\text{Доставка})\}
\end{aligned}
$$

.

Пример мнения с заданной структурой приведен на рисунке 2.10. Итоговая архитектура нейронной сети для извлечения мнений из текстов с учетом заданных Lbl, A, $V(a)$ и $A(C_t)$, полученная из общей модели, приведена на рисунке 2.11.

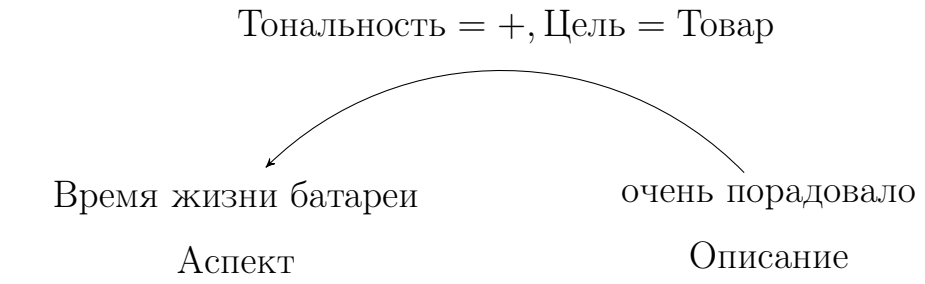


Рисунок 2.10 — Пример формальной структуры мнения пользователя

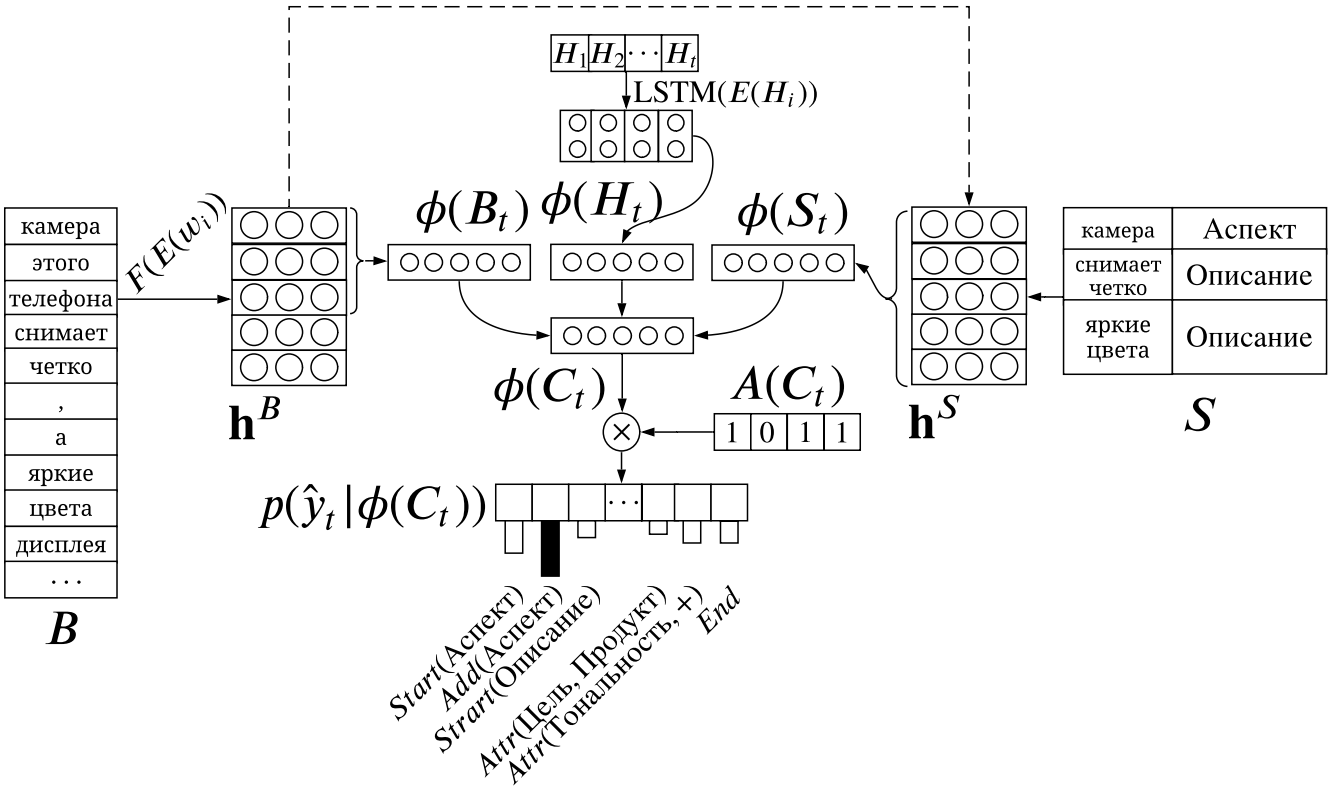


Рисунок 2.11 — Архитектура нейронной сети для извлечения мнений

2.2.2 Подготовка набора данных для обучения модели

Экспериментальное исследование модели для извлечения мнений проводилось на материале отзывов покупателей из интернет-магазина «AliExpress» на русском языке. Набор состоит из текстов о трех самых многочисленных категориях товаров:

1) бытовая техника;
2) дом и авто;
3) одежда.

Предложенные категории содержат товары, обладающие достаточно различным набором аспектов и описаний, что позволяет точнее оценить качество

результатов в разных предментных областях. Кроме того, разделение на категории позволит получить пессимистичные оценки точности модели в случае, когда модель тестируется на текстах отзывов категории, которая не использовалась при обучении.

Разметка данных осуществлялась в специально разработанном для этого веб-приложении, позволяющем нескольким асессорам параллельно осуществлять разметку текстов. Тексты размечались в несколько этапов. Целью первого этапа было обучение команды из 4-х асессоров. В течении 2-х дней каждый асессор разметил 50 текстов отзывов, после чего было оценено качество проведенной работы и уточнены правила проведения разметки текстов. На втором этапе каждый асессор разметил один и тот же набор из 50 отзывов, после чего были рассчитаны попарные коэффициенты каппа Коэна [115] для фрагментов сущностей, позволяющие оценить согласованность разметки у разных асессоров. Полученные значения приведены в таблице 2.3.

Таблица 2.3 — Попарные коэффициенты согласованности асессоров при разметке отзывов «Ali Express»

	Асессор 1	Асессор 2	Асессор 3	Асессор 4
Асессор 1	–	0,62	0,59	0,58
Асессор 2	0,62	–	0,65	0,46
Асессор 3	0,59	0,65	–	0,48
Асессор 4	0,58	0,46	0,48	–

По полученным результатам был сделан вывод о том, что выполненная асессором 4 разметка сильнее отличается от разметки, выполненной тремя другими членами команды. Поэтому в дальнейшей разметке принимали участие асессоры 1, 2 и 3. Затем последовал финальный этап разметки, длившийся 3 недели. После этого было проведено несколько этапов очистки собранного набора данных, в ходе которых исправлялись ошибки в разметке, удалялись дубликаты предложений и добавлялись новые отзывы, размеченные согласно обновленным рекомендациям. Количественные характеристики полученного набора данных приведены в таблице 2.4. Примеры размеченных предложений из текстов отзывов пользователей из магазина «AliExpress» представлены на рисунке 2.12.

Важным обстоятельством в рамках данного исследования является степень похожести текстов различных категорий с точки зрения встречающихся

Таблица 2.4 — Количественные характеристики набора данных «Ali Express»

Характеристика	Бытовая техника	Дом и авто	Одежда
Количество отзывов	1068	1042	1068
Предложений на отзыв	4,88	4,82	4,86
Количество мнений	6481	6590	7419
Слов в аспекте	1,39	1,32	1,32
Слов в оцен. высказывании	2,43	2,28	2,18

Рисунок 2.12 — Примеры размеченных предложений для отзывов «AliExpress»

форм аспектов и описаний. Высокая степень похожести текстов свидетельствует о том, что потребители различных товаров в своих отзывах упоминают одни и те же атрибуты товаров и высказывают свое отношение к ним схожим образом. Хотя даже в таком случае извлеченная информация о продукте может быть полезной, низкое разнообразие форм может сделать задачу извлечения фрагментов тривиальной и не требующей применения сложных моделей. Степень пересечения категорий c_1 и c_2 рассчитывалась следующим образом:

$$o(c_1, c_2) = \frac{|S(c_1) \cap S(c_2)|}{|S(c_1)|}, \tag{2.12}$$

где $S(c)$ – множество всех упоминаний поверхностных форм фрагментов в текстах категории c. Дополнительно оценить характер пересечения можно, рассчитав доли высокочастотных и низкочастотных форм в $o(c_1, c_2)$:

$$o_{>}(c_1, c_2) = \frac{|\{s : s \in S(c_1) \cap S(c_2) \wedge n(s) > 3)\}|}{|S(c_1) \cap S(c_2)|}, \tag{2.13}$$

$$o_{<}(c_1, c_2) = \frac{|\{s : s \in S(c_1) \cap S(c_2) \wedge n(s) \leqslant 3)\}|}{|S(c_1) \cap S(c_2)|}. \tag{2.14}$$

В таблицах 2.5 и 2.6 приведены рассчитанные значения пересечения форм для аспектов и описаний. Степень пересечения форм аспектов выше, чем у описаний, что объясняется их меньшей средней длиной и, следовательно, более высокой вероятностью совпадения формы. Для обоих типов фрагментов основную долю среди совпадающих упоминаний составляют высокочастотные формы. Таким образом, при количественной оценке качества извлечения и обобщающей способности модели, следует обратить внимание на низкочастотные формы отдельно.

Следует отметить, что при использовании векторных представлений слов в качестве признаков модели эффективная степень пересечения может быть выше в силу того, что семантически похожие слова будут иметь похожие векторные представления несмотря на разную поверхностную форму.

Таблица 2.5 — Взаимное пересечение форм аспектов между категориями

	Бытов. техника	Дом и авто	Одежда
Бытов. техника	–	0,43/0,86	0,36/0,87
Дом и авто	0,43/0,87	–	0,47/0,86
Одежда	0,34/0,88	0,43/0,87	–

Таблица 2.6 — Взаимное пересечение форм описаний между категориями

	Бытов. техника	Дом и авто	Одежда
Бытов. техника	–	0,31/0,75	0,29/0,76
Дом и авто	0,31/0,77	–	0,35/0,76
Одежда	0,26/0,78	0,32/0,79	–

Частоты встречаемости атрибутов приведены на рисунке 2.13. Из приведенных диаграмм видно, что значения атрибутов распределены в текстах неравномерно: наиболее представленными являются положительная тональность и «товар» в качестве цели высказывания. Если оценить степень неравномерности количественно как отношение частот наиболее и наименее распространенных классов, то для атрибута «тональность» она составляет 3,7, 3,9 и 4,1 в соответствующих категориях, для атрибута «цель» — 3,9, 4,2 и 4,9. Наличие серьезного дисбаланса в распределении значений атрибутов делает оправданным использование макро-усреднения F-меры при оценке качества их определения.

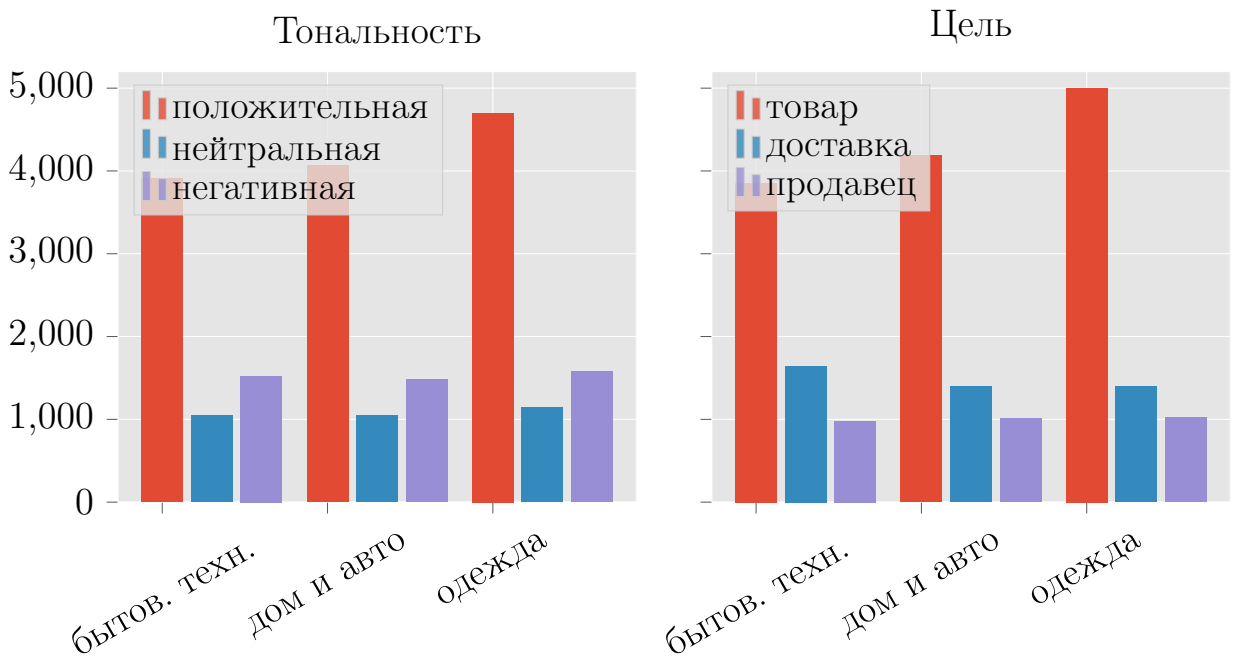


Рисунок 2.13 — Частота встречаемости значений атрибутов в текстах отзывов

2.2.3 Экспериментальные исследования модели и анализ результатов

Экспериментальное исследование нейросетевой модели для извлечения и анализа пользовательских мнений проводилось в сравнении с несколькими альтернативами.

1. *Базовая модель* запоминает все фрагменты, встретившиеся ей в обучающей выборке. Во время предсказания фрагменты формируются из подстрок максимальной длины, которые совпадают с запомненными на этапе обучения фрагментами. Отношениями связываются пары аспектов и описаний, находящиеся ближе всего друг к другу. Так как данная модель не обладает возможностями к обобщению, она может использоваться для оценки фактора мемоизации поверхностных форм фрагментов составных объектов при оценке качества других рассматриваемых моделей.

2. *Многокомпонентная гибридная модель на основе свёрточных и рекуррентных нейронных сетей* (Hybrid-NN) из работы [11]. Каждый компонент отвечает за предсказание отдельных частей составных объектов: Span-CNN-RNN извлекает фрагменты, Link-RNN – отношения между ними, Sentiment-RNN – тональность описаний. Для адаптации модели к рассматриваемой задаче был добавлен компонент Target-RNN, по своей структуре

аналогичный Sentiment-RNN, но определяющий цель высказывания по аспекту.

3. *Модель на основе двунаправленной рекуррентной сети LSTM и условного случайного поля* (LSTM-CRF) для извлечения фрагментов, по структуре повторяющая модель для извлечения именованных сущностей из [91]. Для предсказания отношений и атрибутов использовались компоненты Link-RNN, Sentiment-RNN и Traget-RNN из модели Hybrid-NN.

Компоненты моделей Hybrid-NN и LSTM-CRF обучаются независимо друг от друга, как это показана на рисунке 2.14. При этом компоненты, которые ожидают на входе результат предсказания фрагментов (Link-RNN, Sentiment-RNN и Type-RNN). Так как во время вывода истинные фрагменты не известны, все компоненты, совершающие предсказания на их основе, во время обучения получают на вход не предсказания, а истинные фрагменты из обучающего множества, как это показано на рисунке 2.15. Таким образом, компоненты для определения структуры и атрибутов объектов обучаются в предположении о безошибочной работе компонента для извлечения фрагментов.

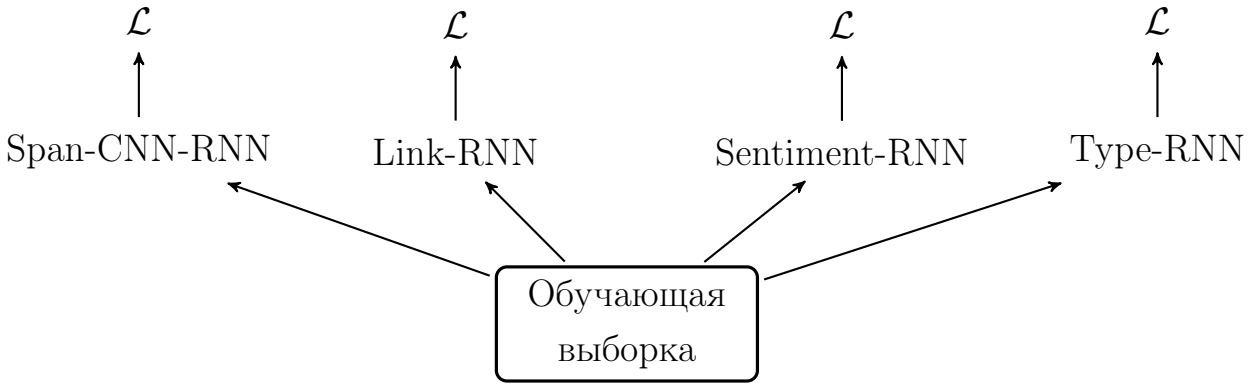


Рисунок 2.14 — Схема обучения отдельных компонентов гибридной модели

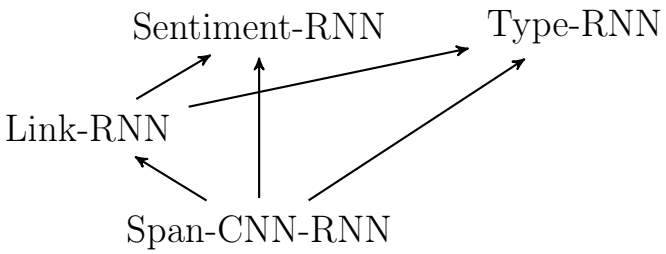


Рисунок 2.15 — Схема вывода гибридной модели

Если во время тестирования компонент для определения фрагментов ошибается в своем предсказании, эта ошибка распространяется далее по модели, негативно влияя на работу компонентов более высокого уровня. Данное явление

хорошо изучено в научной литературе, посвященной обработке естественного языка [13—15]. При использовании моделей на основе нейронных сетей одним из возможных способов решения данной проблемы является передача между компонентами не результатов предсказания, а векторных представлений результата. Например, в [67] при обучении модели синтаксического анализа в качестве входа используются представления, полученные из компонента для частеречного анализа. Обе сети обучаются одновременно, реализуя таким образом принцип многозадачного обучения.

В ходе экспериментов рассмотренные альтернативы будут сравниваться с двумя вариантами нейросетевой модели на основе системы переходов, отличающихся способом извлечения признаков элементов последовательности Trans-CNN со сверточной сетью 1.2 и Trans-LSTM с Bi-LSTM (1.3).

При обучении Trans-CNN использовались следующие параметры:

– окно свертки – 3 слова;
– сверточные фильтры – 150;
– сверточных слоев – 3;
– нелинейность между сверточными слоями – Maxout.

Параметры для обучения Trans-LSTM:

– рекуррентных двунаправленных слоёв – 2 ;
– размер скрытого слоя LSTM – 200.

Параметры, общие для обоих вариантов:

– размер $LSTM_H$ – 30;
– размер $E(H_t)$ – 30;
– размер $E(type(i))$ – 30;
– n_B – 3;
– n_S – 7;
– нейронов в скрытом слое классификатора – 150.

Векторные представления слов строились предобученной моделью fastText[1] для русского и английского языков. Данная версия модели обучена на комбинированном корпусе текстов Wikipedia и Common Crawl[2] на соответствующих языках. Параметры fastText во время обучения полагались фиксированными и к ним не применялась процедура обратного распространения ошибки.

[1]https://fasttext.cc/docs/en/crawl-vectors.html
[2]https://commoncrawl.org/

Оптимизация параметров модели осуществляется методом Adam со скоростью обучения 10^{-3}. Для предотвращения переобучения используются следующие техники регуляризации:

– прореживание [116] (dropout) скрытого слоя классификатора – 10%;

– прореживание сверточных слоев с вероятностью 10%;

– вариационное прореживаение [117] (variational dropout) с вероятностью 20% в каждом рекуррентном слое для Trans-LSTM.

– регуляризация L_2 нормы весов с коэффициентом $\lambda_{L_2} = 1{,}2 \times 10^{-6}$.

Вычисление оценок качества извлечения производилось с помощью процедуры k-fold кросс-валидации, при которой выборка разбивается на k непересекающихся блоков, затем производится k итераций обучения модели на объединении $k-1$ блоков и расчет оценки на оставшемся блоке, который не использовался при обучении. Каждый блок участвует в качестве тестового один раз. Набор из k полученных оценок затем усредняется.

В данной работе каждый блок формировался из отзывов одной категории товаров. Такой режим вычисления оценок позволяет уменьшить влияние фактора запоминания моделью специфичной для конкретной категории лексики. Во время обучения модель может запоминать типы и наличие связей между определенными словами и выражениями, поэтому разбиение выборки на блоки случайным образом может не выявить этого эффекта и тестирование не покажет потенциальных проблем в обобщающей способности модели. Схема использования выборки для трех категорий представлена на рисунке 2.16, где цветом отмечен блок, используемый для тестирования.

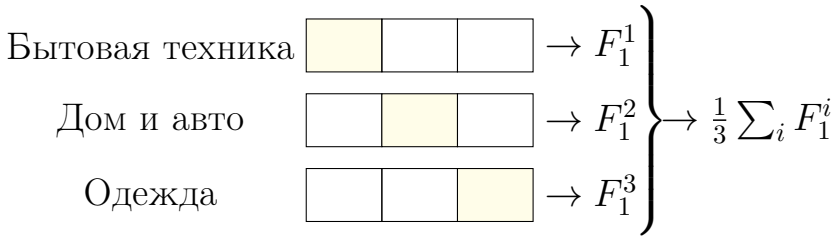


Рисунок 2.16 — Обучение и тестирование моделей с помощью процедуры кросс-валидации

Усредненные по трем категориям товаров из «Ali Express» значения F_1 при извлечении всех типов сущностей составного объекта для всех трех рассмотренных моделей приведены в таблице 2.7. Жирным выделена лучшая достигнутая точность, курсивом – следующая за ней. Базовая модель, несмот-

Таблица 2.7 — Оценка качества извлечения частей составных объектов на наборе данных «Ali Express» (F_1)

Модель	Аспект	Описание	Отнош.	Тонал.	Цель
Базовая	0,569	0,611	0,318	–	–
Hybrid-NN	0,753	0,763	0,661	0,447	0,549
LSTM-CRF	*0,771*	*0,782*	*0,713*	*0,484*	*0,591*
Trans-CNN	0,770	0,787	0,699	0,532	0,659
Trans-LSTM	**0,788**	**0,802**	**0,723**	**0,578**	**0,684**
Сред. Trans-LSTM	0,795		0,723	0,631	

ря на простоту, смогла показать относительно высокий F_1 при извлечении фрагментов: 0,569 для аспектов и 0,611 для описаний. Это является следствием низкого разнообразия фрагментов в рассмотренном наборе данных. Однако, простая эвристика не способна обеспечить высокое качество определения отношений между фрагментами по сравнению с другими альтернативами. Интересным является превосходство модели LSTM-CRF над Hybrid-NN при условии того, что модели отличались только используемым компонентом для извлечения фрагментов. Прирост F_1 на 2,4% для аспектов и 2,5% для описаний отразился в 7,9% роста F_1 при обнаружении отношений.

Наилучшие результаты показала модель Trans-LSTM. Улучшение относительно следующей наилучшей альтернативой составило: при извлечении аспектов и описаний – 1,84% и 2,57%, отношений – 1,41%, атрибутов тональности и цели – 19,43% и 15,75%. Итоговый F_1 при извлечении фрагментов составил 0,795, отношений – 0,723, атрибутов – 0,631. Высокие показатели роста качества извлечения атрибутов составных объектов свидетельствуют о том, что предлагаемая модель на основе системы переходов позволяет модели лучше интегрировать информацию о составных частях объекта при предсказании. При этом можно отметить, что точность определения атрибутов остаётся довольно низкой. Это обусловлено низкой точностью определения нейтральной и негативной тональности для всех категорий товаров, что сказывается на финальной оценке, полученной макро-усреднением. Возможной причиной этому являются сильная несбалансированность распределений значений атрибутов в исходном наборе данных и сильная зависимость тональности от конкретной категории и особенностей самого товара. Качество определения цели мнения значительно выше по сравнению с тональностью несмотря на аналогичную ситуацию с сильным дисбалансом возможных значений этого атрибута в выборке. Полные

данные экспериментов для моделей Hybrid-NN, LSTM-CRF и Trans-LSTM приведены в таблицах 2.9, 2.8, 2.10.

Таблица 2.8 — Результаты Hybrid-NN на наборе данных «Ali Express»

Сущность	Бытов. тех.			Дом и авто			Одежда		
	P	R	F_1	P	R	F_1	P	R	F_1
Аспект	0,708	0,744	0,726	0,724	0,762	0,742	0,806	0,774	0,790
Описание	0,807	0,707	0,754	0,791	0,757	0,774	0,740	0,782	0,761
Связь	0,643	0,640	0,642	0,637	0,693	0,664	0,656	0,699	0,677
Тональность									
Положительная	0,536	0,583	0,559	0,636	0,620	0,628	0,617	0,607	0,612
Нейтральная	0,408	0,380	0,394	0,397	0,368	0,382	0,368	0,326	0,346
Негативная	0,384	0,345	0,364	0,481	0,371	0,419	0,353	0,298	0,323
Тип высказывания									
Товар	0,531	0,529	0,530	0,642	0,547	0,591	0,648	0,593	0,619
Доставка	0,512	0,519	0,515	0,500	0,503	0,502	0,580	0,577	0,578
Продавец	0,526	0,536	0,531	0,462	0,527	0,492	0,587	0,571	0,579

Таблица 2.9 — Результаты LSTM-CRF на наборе данных «Ali Express»

Сущность	Бытов. тех.			Дом и авто			Одежда		
	P	R	F_1	P	R	F_1	P	R	F_1
Аспект	0,764	0,760	0,742	0,803	0,761	0,772	0,817	0,819	0,798
Описание	0,817	0,781	0,788	0,838	0,801	0,799	0,771	0,806	0,758
Отношение	0,701	0,693	0,697	0,741	0,710	0,725	0,711	0,724	0,717
Тональность									
Положительная	0,569	0,632	0,599	0,634	0,692	0,662	0,638	0,644	0,641
Нейтральная	0,441	0,388	0,413	0,434	0,415	0,424	0,393	0,368	0,380
Негативная	0,435	0,399	0,416	0,480	0,477	0,479	0,354	0,331	0,342
Тип высказывания									
Товар	0,580	0,592	0,586	0,638	0,633	0,636	0,666	0,655	0,661
Доставка	0,552	0,532	0,542	0,567	0,608	0,587	0,609	0,590	0,599
Продавец	0,551	0,590	0,570	0,471	0,605	0,529	0,610	0,613	0,611

Для оценки влияния отдельных компонентов конфигурации на точность было проведено абляционное исследование модели Trans-LSTM. В каждом эксперименте из модели исключался один из следующих элементов конфигурации: S, H, B. Соответствующим образом изменялся набор признаков в выражении 2.2. Так как исключать список необработанных слов B не имеет смысла, был проведен эксперимент с исключением функции получения контекстуализированного представления $F(E(w_i))$. Результаты экспериментов, приведённых

Таблица 2.10 — Результаты Trans-LSTM на наборе данных «Ali Express»

Сущность	Бытов. тех.			Дом и авто			Одежда		
	P	R	F_1	P	R	F_1	P	R	F_1
Аспект	0,755	0,760	0,757	0,778	0,781	0,779	0,835	0,819	0,827
Описание	0,823	0,769	0,795	0,838	0,796	0,817	0,775	0,814	0,794
Связь	0,706	0,701	0,703	0,719	0,738	0,729	0,720	0,756	0,737
Тональность									
Положительная	0,666	0,680	0,673	0,709	0,710	0,709	0,747	0,701	0,724
Нейтральная	0,493	0,550	0,520	0,517	0,550	0,533	0,485	0,550	0,515
Негативная	0,507	0,470	0,488	0,559	0,487	0,520	0,551	0,492	0,520
Тип высказывания									
Товар	0,640	0,635	0,638	0,685	0,660	0,672	0,745	0,679	0,710
Доставка	0,653	0,680	0,666	0,675	0,665	0,670	0,689	0,733	0,710
Продавец	0,665	0,671	0,668	0,722	0,723	0,723	0,702	0,703	0,703

в таблице 2.11, показывают, что наибольшее влияние на качество извлечения составных объектов оказывает история совершенных предсказаний H.

Таблица 2.11 — Результаты абляционных экспериментов на наборе данных «Ali Express» для Trans-LSTM

Изменение	Аспект	Описание	Отношн.	Тонал.	Цель	Сред.
без $\phi(S_t)$	−3,0%	−5,1%	−4,6%	−7,7%	−8,4%	−5,6%
без $\phi(H_t)$	−14,2%	−10,2%	−22,4%	−40,6%	−26,0%	−22,6%
без $F(E(w_i))$	−6,8%	−5,5%	−9,4%	−21,7%	−10,0%	−10,5

Важным свойством моделей является возможность извлекать составные объекты при работе с длинными текстами. На графике (рисунок 2.17) продемонстрировано сравнение точности извлечения фрагментов, отношений и атрибутов моделями Hybrid-NN, LSTM-CRF и Trans-LSTM в зависимости от длины предложения. Для всех моделей характерно резкое снижение точности при увеличении длины предложения, при этом хуже всего показывает себя Hybrid-NN. Значимой разницы в поведении Trans-LSTM и Hybrid-NN при извлечении фрагментов и отношений не наблюдается, однако преложенная модель значительно лучше определяет атрибуты составных объектов на всех рассмотренных длинах предложений.

Способность моделей строить отношения между удаленными сущностями показана на графике (рисунок 2.18) как зависимость точности извлечения отношений от количества слов, разделяющих связанные ими фрагменты. Все

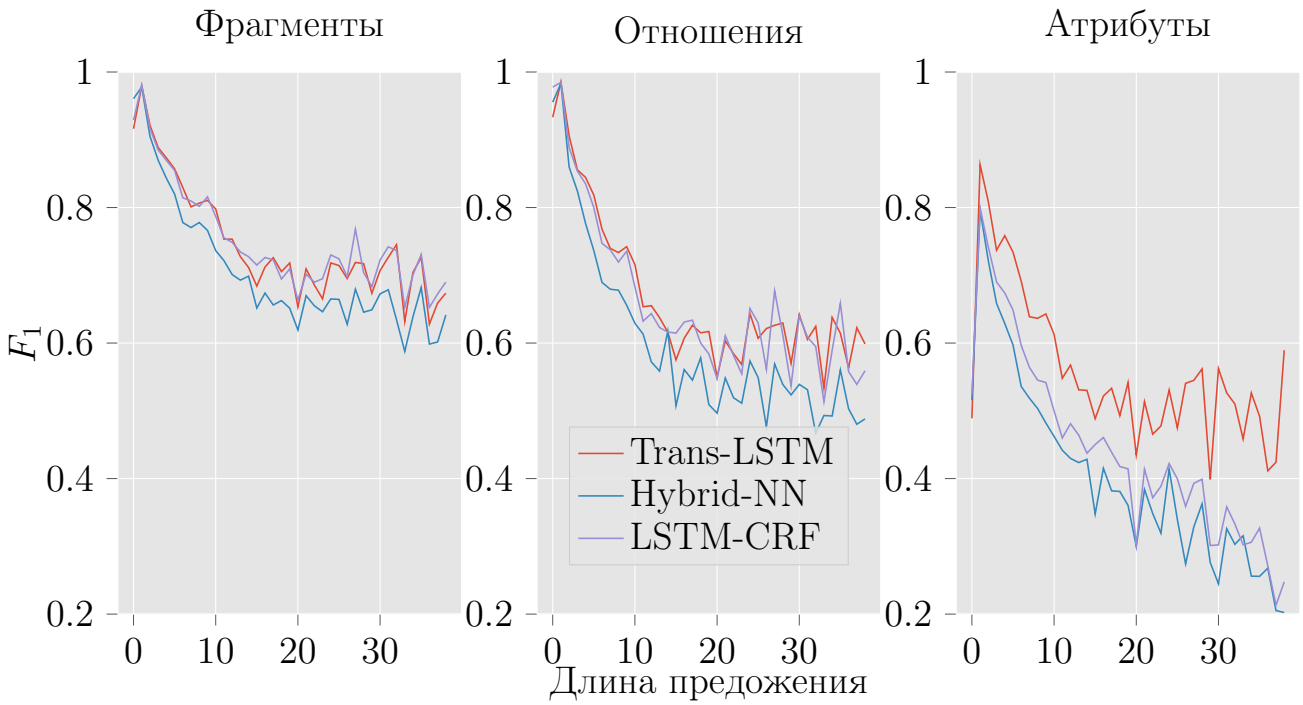


Рисунок 2.17 — Зависимость качества извлечения фрагментов, отношений и атрибутов от длины предложения в отзывах пользователей «Ali Express»

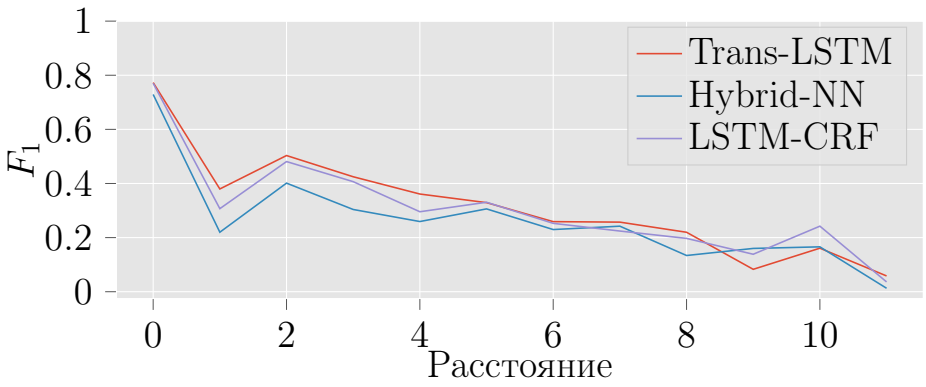


Рисунок 2.18 — Зависимость качества извлечения отношений от расстояния между фрагментами

рассмотренные модели демонстрируют общую тенденцию к сильному ухудшению качества извлечения отношений при увеличении расстояния между фрагментами. Hybrid-NN показывает отставание от альтернатив практически на всех рассмотренных расстояниях, Trans-LSTM имеет незначительное преимущество над LSTM-CRF на длинах 1 и 2. Из продемонстрированных результатов можно сделать вывод о том, что все рассмотренные модели в одинаковой степени полагаются на близость фрагментов несмотря на то, что только в Trans-LSTM содержится явное предположение о взаимном расположении фрагментов в виде перехода $Link(n_1,n_2)$. Это служит подтверждением того, что несмотря

на совершаемые при извлечении фрагментов ошибки, модель способна лучше учитывать частичные совпадения и предсказывает большее количество связей правильно.

Указанная в разделе 2.2.2 проблема высокой степени пересечения поверхностных форм фрагментов между текстами отзывов покупателей может привести к тому, что во время обучения модель запомнит конкретные формы из обучающей выборки и покажет высокие значения F_1 при извлечении фрагментов. Оценить степень влияния данной проблемы можно, отдельно рассчитав качество извлечения двух групп фрагментов: присутствующих как в обучающем, так и в проверочном наборе; присутствующих только в проверочном наборе. Результаты оценки приведены в таблице 2.12. Полученные результаты

Таблица 2.12 — Качество извлечения фрагментов в зависимости от упоминания в обучающем наборе на данных «Ali Express»

Модель	Аспект		Описание	
	О и П	**П**	**О и П**	**П**
Hybrid-NN	0.791	0.562	0.777	0.629
LSTM-CRF	0.821	0.619	0.806	0.689
Trans-LSTM	0.822	0.623	0.810	0.691

говорят о том, что качество извлечения встречающихся только в проверочном наборе текстов всегда ниже, чем встречающихся в обучающем и проверочном. При этом модель Hybrid-NN показывает меньшую обобщающую способность по сравнению с альтернативами: точность извлечения аспектов и описаний из второй группы ниже на 0.061 и 0.062 по сравнению с Trans-LSTM.

Примеры предсказаний, полученных моделью на текстах из валидационной выборки (**П**), а также эталонная разметка (**И**) приведены ниже. На примере предложения 1 можно увидеть, что хотя некоторые предсказания не содержатся в эталоне, они являются осмысленными с точки зрения человека.

Пример 1

И $[\text{Носки}]_{A1}$ $[\text{как на картинке}]^{+:\text{тов}}_{O1}$, $[\text{хорошего качества}]^{+:\text{тов}}_{O1}$, $[\text{приятные на ощупь}]^{+:\text{тов}}_{O1}$

П $[\text{Носки}]_{A1}$ $[\text{как на картинке}]^{+:\text{тов}}_{O1}$, $[\text{хорошего}]^{+:\text{тов}}_{O2}$ $[\text{качества}]_{A2}$, $[\text{приятные}]^{+:\text{тов}}_{O3}$ $[\text{на ощупь}]_{A3}$

Пример 2

И [Палатка]$_{A1}$ [хорошего качества]$^{+:\text{тов}}_{O1}$, [есть небольшой]$^{0:\text{тов}}_{O2}$ [запах]$_{A2}$, но это не беда, [посылка]$_{A3}$ [шла до Уфы месяц]$^{0:\text{дос}}_{O3}$

П [Палатка]$_{A1}$ [хорошего качества]$^{+:\text{тов}}_{O1}$, [есть небольшой]$^{-:\text{тов}}_{O2}$ [запах]$_{A2}$, но это не беда, [посылка]$_{A3}$ [шла до Уфы месяц]$^{-:\text{дос}}_{O3}$

Пример 3

И [Чехлы]$_{A1}$ [ооооочень крутые]$^{+:\text{тов}}_{O1}$, [подходят на любые стулья]$^{+:\text{тов}}_{O1}$, на наши [сели вообще идеально]$^{+:\text{тов}}_{O1}$, [очень рекомендуем]$^{(+:\text{тов}),(+:\text{пр})}_{O(2,3)}$ [товар]$_{A2}$ и [продавца]$_{A3}$!

П [Чехлы]$_{A1}$ [ооооочень крутые]$^{+:\text{тов}}_{O1}$, [подходят на любые стулья]$^{+:\text{тов}}_{O1}$, на наши сели вообще идеально,[очень рекомендуем]$^{(+:\text{тов}),(+:\text{пр})}_{O2}$ [товар]$_{A2}$ и [продавца]$_{A2}$!

Пример 4

И Жаль только что [по России долго]$^{-:\text{дос}}_{O1}$ [шёл]$_{A1}$ 20 дней и [долго]$^{-:\text{дос}}_{O2}$ [растомаживался]$_{A2}$.

П Жаль только что по России [долго]$^{-:\text{дос}}_{O1}$ [шёл]$_{A1}$ [20 дней]$^{0:\text{дос}}_{O1}$ и [долго растомаживался]$^{-:\text{дос}}_{O1}$.

Пример 5

И Продавец прикрепил [неверный]$^{-:\text{прд}}_{O1}$ [номер для отслеживания]$_{A1}$!

П [Продавец]$_{A1}$ [прикрепил неверный номер для отслеживания]$^{-:\text{прд}}_{O1}$!

Пример 6

И Но [некоторые формы]$_{A1}$ [отсутствуют]$^{-:\text{тов}}_{O1}$ или [заменены на другие]$^{-:\text{тов}}_{O1}$ (не соответствие картинке) [качество]$_{A2}$ [на первый взгляд очень хорошее]$^{+:\text{тов}}_{O2}$, не знаю как со временем, сотрётся или нет

П Но некоторые [формы]$_{A1}$ [отсутствуют или заменены на другие]$^{-:\text{тов}}_{O1}$ ([не соответствие картинке]$^{-:\text{тов}}_{A1}$) [качество]$_{A2}$ на первый взгляд [очень хорошее]$^{+:\text{тов}}_{O2}$, [не знаю как со временем, сотрётся или нет]$^{-:\text{тов}}_{O2}$.

Результаты проведенных экспериментов позволяют сделать вывод о том, что предложенная модель показывает лучшие по сравнению с альтернативами результаты при извлечении фрагментов, отношений и атрибутов. Предложенная автором структура модели позволяет обеспечить более высокую точность определения атрибутов для всех рассмотренных длин предложений, с чем не справляются другие рассмотренные модели. Качественный анализ полученных результатов говорит о возможности практического применения предложенной модели для извлечения и анализа пользовательских мнений о потребительских свойствах товаров.

2.3 Нейросетевая модель для обработки запросов пользователей на этапе эксплуатации программного продукта

2.3.1 Постановка задачи обработки запросов пользователей на этапе эксплуатации и сопровождения программного продукта

Конкурентоспособность рыночного программного продукта во многом зависит от качества и своевременности реагирования ИТ-компании на запросы конечных пользователей по проблемам, связанным с некачественной работой программных продуктов (ПП), ошибками в технической документации, недостаточной квалификацией конечных пользователей и т. д. Оперативное решение этих проблем пользователей осуществляется на этапе эксплуатации и сопровождения программного продукта [118]. По данным [119] на эти этапы приходится 67% времени жизненного цикла ПП.

Для обработки запросов пользователей, как правило представленных в неструктурированной текстовой форме (электронные письма, сообщения на форумах и чатах поддержки), и реагирования на них в ИТ-компаниях создаются службы технической поддержки пользователей. Следует отметить, что современные системы автоматизации работы службы поддержки (helpdesk-системами), такие как HappyFox, Service Desk Plus, FreshDesk, Zendesk, имеют достаточно развитый функционал, обеспечивающий прием и хранение заявок пользователей, связь с другими заявками, мониторинг исполнения заявок, хранение истории переписки с пользователем. Однако основная часть работы по семантическому анализу текста обращения пользователя и назначение специалистов по-прежнему лежит на человеке. При росте количества пользователей остро встает проблема масштабируемости поддержки и увеличения финансовых затрат на содержание непрофильных структур.

Кроме обработки «явных» обращений пользователей, связанных с функционалом ПП, большой интерес для компаний могут представлять «неявные» обращения, высказываемые пользователями в каналах, которые редко являются объектами внимания службы поддержки: форумы, блоги и социальные сети за пределами созданных разработчиками ПП групп и сообществ. «Неявные» обращения, выражаемые в виде текстов мнений или отзывов, могут быть на-

столько же полезны для разработчика ПП, как и явные, и вместе с тем намного более многочисленны. Так как ручной анализ большого объема текстовой информации затруднителен, данный пласт информации либо обрабатывается не систематично, либо не обрабатывается вовсе.

Эти проблемы могут быть решены путем автоматизированной обработки текстов обращений пользователей с целью определения высказываний о проблемах пользователей, возникших при эксплуатации ПП. В научной литературе представлен ряд работ, посвященных автоматической обработке текстов пользовательских запросов с целью выявления знаний, полезных при разработке и сопровождении программных продуктов. Так, в [120] для выявления жалоб на программные ошибки из сообщений в багтрекерах проектов на основе открытого исходного кода рассматривается применение решающих деревьев, наивного байесовского классификатора и логистической регрессии. Исследование [121] посвящено анализу различных аспектов текстов отзывов о мобильных приложениях, публикуемых в магазине «Apple AppStore». Отмечается, что хотя часть отзывов не представляет интереса для разработчиков приложений, другие содержат сообщения об ошибках, пользовательский опыт и запросы на введение новых функций. Проблема анализа текстов отзывов из магазина приложений Google на предмет наличия запросов на расширение функциональности рассматривается в [122]. Авторы используют набор лингвистических правил для классификации предложений по критерию «содержит/не содержит запрос», затем при помощи тематической модели LDA определяют основные темы текстов. В [123] авторы предлагают корпус отзывов о мобильных приложениях на немецком языке из магазина приложений «Google Play», который содержит упоминания признаков приложений – аспекты, и оценочные высказывания пользователей о них – описания.

Таким образом, можно выделить два основных недостатка существующих решений для анализа обратной связи (в виде текстов) от пользователей ПП. Во-первых, в рамках одной модели предусмотрено извлечение только одного типа запроса. Многоаспектный анализ потребует интеграции нескольких различных моделей в одно решение, что может быть нетривиальной задачей. Во-вторых, большинство существующих моделей работают на уровне предложений и не позволяют непосредственно определять фразы, в которых высказана основная суть обращения пользователя.

Для устранения перечисленных выше недостатков предлагается извлекать из текстов запросов пользователей информативные фразы (ИФ), содержащих конкретные пожелания и требования пользователей. Структуру ИФ предлагается задать в виде пары «объект – описание», где под объектом понимается упоминание в тексте самого ПП, его функций, элементов графического интерфейса, а под описанием – некоторая фраза, в которой пользователь оценивает объект, рассказывает о сложившейся ситуации.

По аналогии с задачей извлечения и анализа пользовательских мнений о потребительских свойствах товаров предлагается объединить ИФ в группы для концентрации усилий разработчиков на наиболее часто встречающихся типах запросов. В соответствии с универсальной моделью деятельности в процессе эксплуатации ПП субъектом деятельности является пользователь ПП, объектом — программный продукт, средствами — аппаратно-программные средства, обеспечивающие функционирование ПП. На основе данной модели был разработан иерархический классификатор типов ИФ, представленный на рисунке 2.19. Опишем элементы классификатора.

Компетенции пользователя: вопросы пользователей о функциональных возможностях ПП, его графическом интерфейсе и сопутствующей документации.

Оценка функциональных возможностей ПП: положительные и негативные мнения пользователей о функциональных возможностях ПП.

Функционал ПП: фразы, в которых пользователи сообщают о некорректной работе функций ПП и его графического интерфейса, фактических ошибках в документации.

Развитие ПП: запросы пользователей на развитие функциональности ПП и доработку документации.

Сбой в работе программно-аппаратных средств: жалобы пользователей на сбои в работе внешних по отношении к ПП программных и аппаратных средств.

Извлеченная в соответствии с предложенным классификатором информация из запросов пользователей имеет несколько вариантов применения.

Распределение запросов пользователей по специалистам службы поддержки. После извлечения ИФ можно группировать запросы по упомянутым объектам и автоматически назначать специалиста с необходимыми компетенциями.

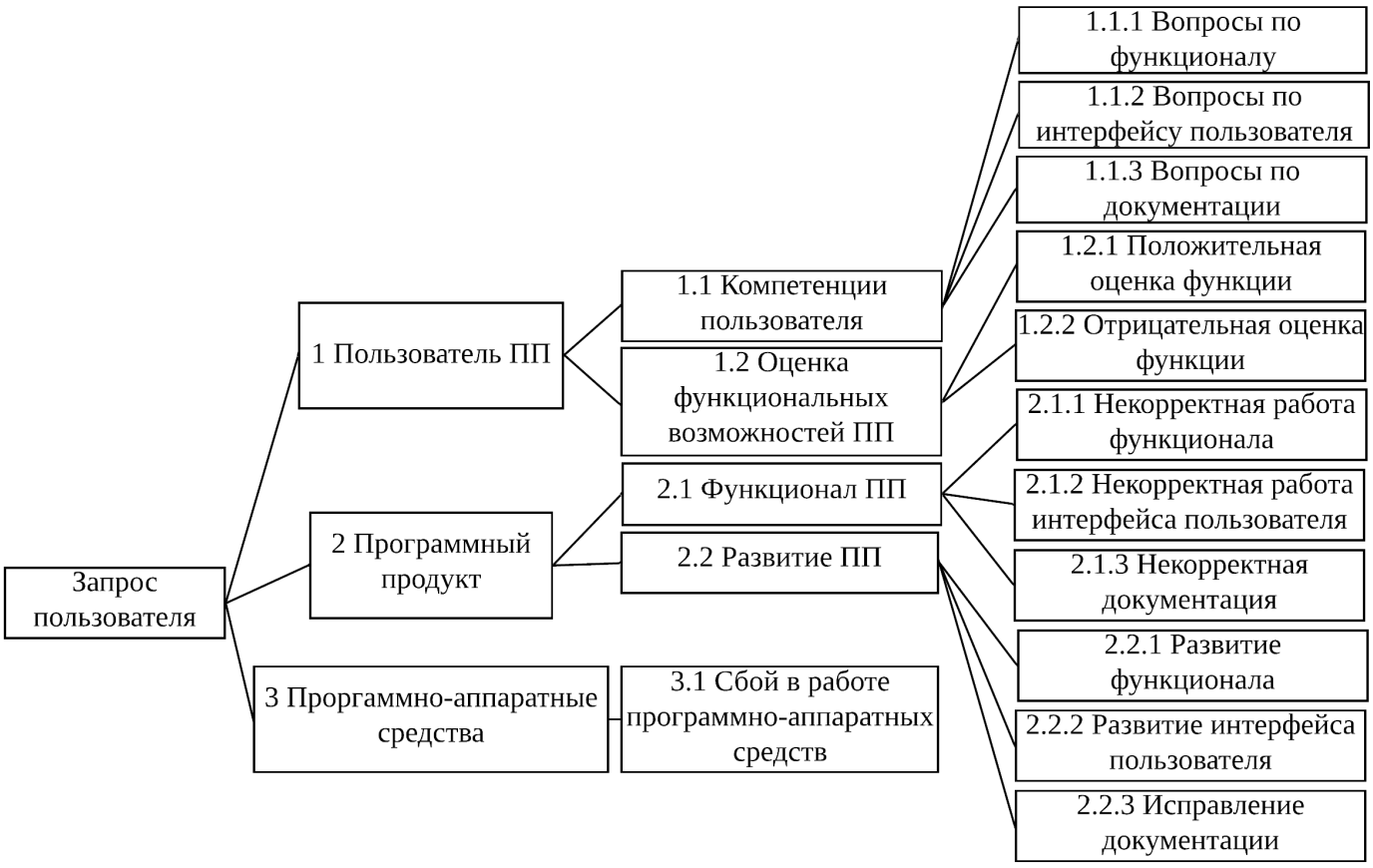


Рисунок 2.19 — Классификатор типов информативных фраз

Получение более объективных представлений о самых важных требованиях пользователей. Небольшие команды разработки могут испытывать сложности с выбором приоритетных направлений в развитии ПП. Имея инструмент, позволяющий анализировать запросы большого числа людей в сети, можно получить достаточно полную картину претензий и предложений пользователей, и направить ресурсы на наиболее реализацию наиболее важных функциональных возможностей и исправление массовых критических ошибок.

Непрерывное отслеживание продуктов конкурентов. Сегодня многие массовые ПП распространяются через специализированные магазины: Google Play Store, App Store, Microsoft Store и т. д. Тексты отзывов пользователей в них находятся в свободном доступе, что позволяет проводить не только анализ собственных продуктов, но и предложений конкурентов.

Для экспериментальной апробации на основе классификатора было выведено 4 общих укрупненных класса ИФ:

1) программная ошибка (ошибки программного продукта);
2) запрос на новый функционал («развитие программного продукта»);

3) положительная оценка функции;

4) отрицательная оценка функции.

Приведём примеры ИФ на примере текстов из магазина приложений «Google Play Market».

Программная ошибка: «плеер не отображается на экране блокировки», «плейлист не обновляется при свейпе вниз».

Запрос на новый функционал: «добавить кнопку проигрывания в перемешку», «иметь возможность редактировать теги».

Положительная оценка функции: «много возможностей для манипуляций над звуком», «отменить прием – быстро».

Отрицательная оценка функции: «не хранится архив записей», «дублирует функционал приложения госуслуги».

С учетом вышеизложенного, задача обработки запросов пользователей на этапе эксплуатации и сопровождения программного продукта может быть представлена в следующем виде. Пусть задано множество текстов запросов пользователей T, содержащих информативные фразы из множества O касательно программных продуктов P. Информативной фразой мы будем называть тройку (Функция, Описание, Тип). Функцией будем называть как функциональные возможности, элементы графического элемента пользователя, так и упоминания самого ПП. Описанием будем называть последовательность слов, содержащую высказанное пользователем мнение о функции, её состоянии или свои пожелания, описание сложившейся ситуации. Множество типов ИФ зададим в соответствии с предложенным иерархическим классификатором, однако для практической апробации будем использовать укрупненные классы: положительная оценка функции (далее +), отрицательная оценка функции (далее −), ошибка (соответствует типам «Сбой в работе программно-аппаратных средств», «Некорректная работа функционала», «Некорректная работа интерфейса пользователя»), запрос (соответствует типу «Запрос на новый функционал»).

Имея исходные множества T, P, O и заданную структуру классификатора ИФ, необходимо извлечь из новых текстов запросов $\hat{T}$ информативные фразы $\hat{O}$ о новом наборе программных продуктов $\hat{P}$. Для решения задачи необходимо адаптировать нейросетевую модель модель (2.1—2.7) путем определения состава

множеств Lbl, A, $V(a)$ согласно смысловому наполнению ИФ [20; 24]:

$$Llb = \{\text{Функция}, +, -, \text{Ошибка}, \text{Запрос}\},$$
$$A = \varnothing.$$

Множество переходов Y в этом случае будет состоять из следующих элементов:

$$\begin{aligned} Y = \{&Shift, Start(\text{Функция}), Start(+), Start(-),\\ &Start(\text{Ошибка}), Start(\text{Запрос}), Add(\text{Функция}),\\ &Add(+), Add(-), Add(\text{Ошибка}), Add(\text{Запрос}),\\ &Link(n_1,n_2), End\}. \end{aligned}$$

Необходимо дополнить условие допустимости перехода $Link(n_1,n_2)$ ограничением, по которому отношениями могут связываться только фрагмент типа Функция и фрагмент любого другого типа:

$$\exists S_{n_1} \wedge \exists S_{n_2} \wedge (n_1,n_2) \notin L \wedge (type(S_{n_1}) = \text{Feature} \wedge type(S_{n_2}) \neq Object)$$

Пример ИФ с заданной структурой приведен на рисунке 2.20. Итоговая архитектура нейронной сети для извлечения ИФ из текстов с учетом заданных $Label$, AV и $A(C_t)$, полученная из общей модели, приведена на рисунке 2.21.

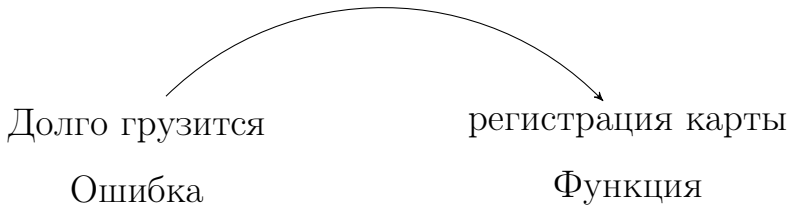


Рисунок 2.20 — Пример формальной структуры мнения пользователя

2.3.2 Подготовка набора данных для обучения модели

Для оценки качества предложенной модели, на основе материалов магазина приложений «Google Play» был подготовлен корпус обращений пользователей на русском языке. Сбор обращений производился из следующих девяти категорий мобильных приложений:

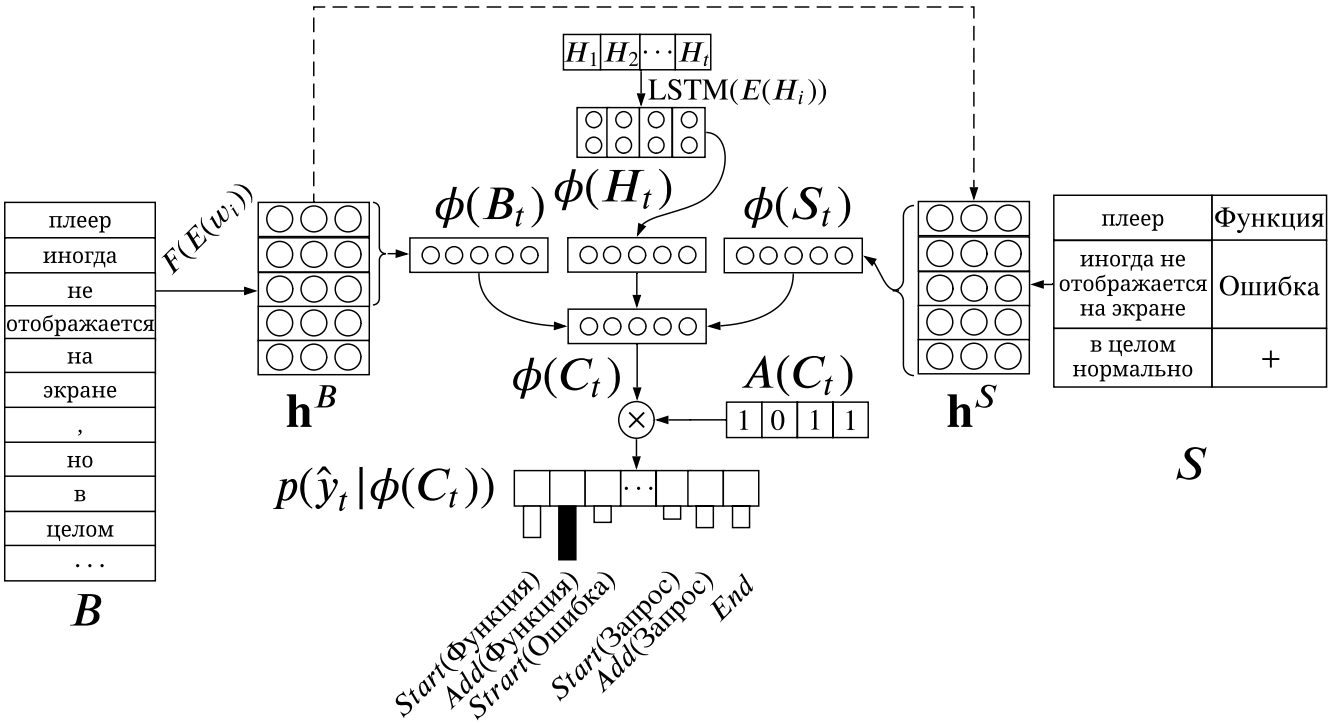


Рисунок 2.21 — Архитектура нейронной сети для обработки запросов пользователей

1) автомобили и транспорт;
2) карты и навигация;
3) медицина;
4) музыка и аудио;
5) персонализация;
6) финансы;
7) шоппинг;
8) образование;
9) видеоплееры и редакторы.

В каждой категории было выбрано пять приложений и для каждого приложения случайным образом выбрано 20 обращений. Каждое обращение, в свою очередь, разделялось на отдельные предложения. Разметка запросов пользователей осуществлялась автором работы на протяжении 3 недель, после чего результат разметки был отдан стороннему человеку для аудита и поиска ошибок. Затем на основе полученной обратной связи в разметку были внесены исправления. Количественные характеристики полученного набора данных приведены в таблице 2.13.

Примеры размеченных предложений представлены на рисунке 2.22.

Таблица 2.13 — Количественные характеристики набора данных «Google Play Store»

Характеристика	Google Play Store
Количество запросов пользователей	900
Предложений на отзыв	4,97
Количество объектов	2273
Количество положительных оценок функции	999
Количество отрицательных оценок функции	851
Количество запросов на новый функционал	200
Количество программных ошибок	677
Слов в аспекте	1,45
Слов в оцен. высказывании	2,73

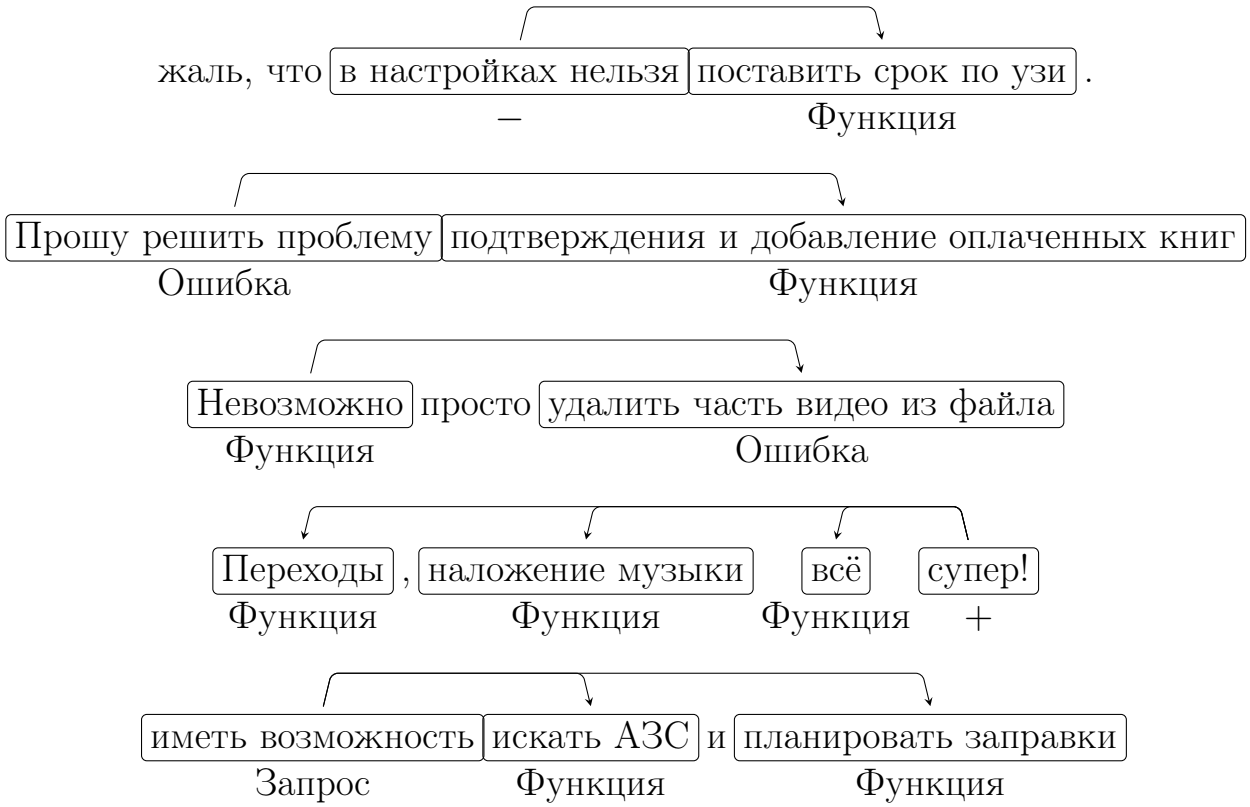


Рисунок 2.22 — Примеры размеченных предложений для обращений «Google Play Market»

По аналогии с набором данных для решения задачи извлечения и анализа пользовательских мнений, оценим степень пересечения поверхностных форм фрагментов в текстах обращений пользователей из различных категорий приложений по формулам (2.12—2.14). Результаты приведены в таблице 2.14. Степень пересечения форм для обращений пользователей ПО в среднем ниже по сравнению с отзывами на товары и составляет в среднем 17,4% для функций и 11,3% для описаний. В таком случае модели в меньшей степени могут опираться на

запоминание поверхностных форм фрагментов, и результаты процедуры категориальной кросс-валидации в большей степени продемонстрируют способность моделей к обобщению.

Таблица 2.14 — Взаимное пересечение форм фрагментов между текстами различных категорий приложений

	А	О	Ф	Н	Мд	Му	П	Ш	В
	Аспект								
А	–	0.27/0.9	0.30/0.9	0.22/0.9	0.27/0.9	0.24/0.9	0.13/0.8	0.27/0.9	0.26/0.9
О	0.20/0.9	–	0.22/0.8	0.15/0.9	0.20/0.8	0.18/0.8	0.09/0.8	0.20/0.8	0.22/0.7
Ф	0.19/0.8	0.19/0.8	–	0.14/0.9	0.18/0.8	0.15/0.9	0.10/0.7	0.18/0.9	0.18/0.8
Н	0.19/0.8	0.18/0.9	0.20/0.8	–	0.20/0.8	0.17/0.9	0.10/0.8	0.18/0.9	0.21/0.8
Мд	0.23/0.8	0.24/0.8	0.24/0.7	0.20/0.8	–	0.19/0.9	0.12/0.8	0.23/0.8	0.22/0.7
Му	0.14/0.9	0.14/0.8	0.14/0.9	0.12/0.9	0.14/0.9	–	0.09/0.7	0.14/1.0	0.18/0.8
П	0.09/0.8	0.09/0.8	0.11/0.7	0.08/0.9	0.09/0.8	0.11/0.7	–	0.08/0.8	0.10/0.7
Ш	0.21/0.9	0.22/0.8	0.22/0.9	0.17/0.9	0.21/0.8	0.17/1.0	0.09/0.8	–	0.19/0.8
В	0.17/0.9	0.20/0.8	0.19/0.8	0.16/0.8	0.17/0.8	0.20/0.8	0.10/0.7	0.16/0.8	–
	Описание								
А	–	0.14/0.4	0.15/0.5	0.09/0.5	0.12/0.5	0.12/0.5	0.08/0.5	0.16/0.5	0.14/0.5
О	0.10/0.4	–	0.15/0.6	0.09/0.5	0.13/0.6	0.12/0.6	0.09/0.4	0.15/0.5	0.12/0.5
Ф	0.09/0.6	0.13/0.6	–	0.09/0.6	0.12/0.6	0.14/0.6	0.10/0.4	0.14/0.6	0.15/0.5
Н	0.07/0.5	0.10/0.2	0.11/0.4	–	0.09/0.3	0.09/0.3	0.07/0.5	0.09/0.3	0.11/0.4
Мд	0.10/0.6	0.15/0.5	0.17/0.6	0.10/0.5	–	0.14/0.6	0.09/0.3	0.15/0.6	0.13/0.5
Му	0.08/0.5	0.11/0.6	0.15/0.6	0.07/0.4	0.11/0.6	–	0.07/0.6	0.12/0.6	0.13/0.7
П	0.06/0.4	0.10/0.2	0.13/0.3	0.07/0.6	0.08/0.2	0.08/0.3	–	0.09/0.3	0.09/0.3
Ш	0.13/0.5	0.17/0.4	0.18/0.5	0.09/0.3	0.15/0.5	0.16/0.4	0.09/0.3	–	0.14/0.3
В	0.09/0.6	0.10/0.5	0.15/0.5	0.08/0.5	0.10/0.6	0.13/0.6	0.07/0.4	0.10/0.6	–

Распределение типов описаний во всём наборе текстов обращений пользователей представлено на рисунке 2.23. Соотношение наиболее и наименее распространенных типов равняется 5.

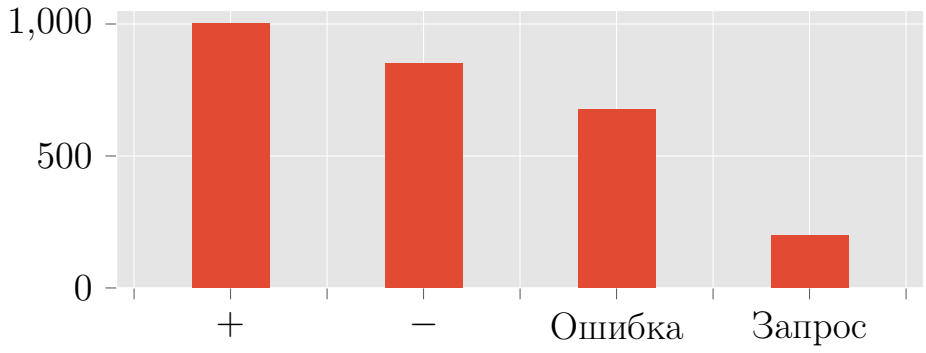


Рисунок 2.23 — Частота встречаемости типов описаний в текстах обращений пользователей «Google Play»

2.3.3 Экспериментальное исследование модели и анализ результатов

В ходе экспериментов использовался тот же набор моделей, что и при решении задачи извлечения и анализа пользовательских мнений в параграфе 2.2.3.

1. Базовая модель, запоминающая обучающую выборку.

2. Многокомпонентная модель на основе свёрточных и рекуррентных нейронных сетей Hybrid-NN.

3. Модель на основе нейронной сети Bi-LSTM и CRF (LSTM-CRF).

4. Два варианта предложенной нейросетевой модели на основе системы переходов: Trans-CNN на основе свёрточных нейронных сетей 1.1 и Trans-LSTM на основе рекуррентных нейронных сетей LSTM 1.3.

Так как в данной задаче объекты лишены атрибутов, компоненты для их предсказания не использовались. Векторные представления слов аналогичным образом получались с помощью модели fastText, обученной на корпусе Common Crawl. Оптимизация параметров модели осуществляется методом Adam со скоростью обучения 10^{-3}. Для предотвращения переобучения используются следующие техники регуляризации:

– прореживание [116] (dropout) с вероятностью 10% после каждого сверточного слоя в Trans-CNN;

– вариационное прореживаение [117] (variational dropout) с вероятностью 10% в каждом рекуррентном слое для Trans-LSTM;

– регуляризация L_2 нормы весов с коэффициентом $\lambda_{L_2} = 1{,}2 \times 10^{-6}$.

Обучение моделей и вычисление оценок качества извлечения производилось с помощью процедуры k-fold кросс-валидации, рассмотренной в 2.2.3. В таблице 2.15 приведены показатели F_1 для извлечения функции, всех типов ИФ (усредненный) и связей между ними, которые усреднены по всем 9 категориям приложений. Жирным выделен лучший полученный результат, курсивом – следующий за ним. В данном случае базовая модель демонстрирует значительно более низкие результаты как при извлечении фрагментов, так и отношений между ними, что объясняется меньшим средним уровнем взаимного пересечения поверхностных форм фрагментов в текстах обращений для разных категорий приложений. Наилучшие результаты имеет модель Trans-LSTM, рост F_1 по срав-

Таблица 2.15 — Оценка качества извлечения частей составных объектов на наборе данных «Google Play»

Модель	Функция	Описание	Отношение
Базовая	0,192	0,157	0,275
Hybrid-NN	0,464	0,468	0,396
LSTM-CRF	*0,650*	*0,552*	*0,455*
Trans-CNN	0,601	0,551	0,654
Trans-LSTM	**0,675**	**0,592**	**0,693**
Сред. Trans-LSTM	0,633		0,693

нению с LSTM-CRF составил: 3,9% для функций, 7,2% для описаний и 52% для отношений. Неагрегированные результаты оценки качества извлечения частей составных объектов каждой моделью для каждой категории приложений приведены в таблицах 2.16–2.18.

Таблица 2.16 — Результаты Hybrid-NN на наборе данных «Google Play» (F_1)

Категория	Функция	Полож.	Отриц.	Ошибка	Запрос функц.	Отношение
Авто	0,685	0,583	0,291	0,475	0,261	0,424
Персонал.	0,413	0,590	0,244	0,644	0,148	0,385
Музыка	0,269	0,311	0,661	0,581	0,249	0,339
Навигация	0,474	0,494	0,307	0,653	0,331	0,370
Медицина	0,459	0,652	0,412	0,640	0,389	0,408
Финансы	0,373	0,344	0,665	0,650	0,402	0,420
Шоппинг	0,470	0,713	0,701	0,387	0,337	0,430
Образов.	0,705	0,436	0,700	0,339	0,299	0,398
Видео	0,328	0,651	0,643	0,308	0,356	0,388
Среднее	0,464	0,530	0,514	0,520	0,308	0,396

Таблица 2.17 — Результаты LSTM-CRF на наборе данных «Google Play» (F_1)

Категория	Функция	Полож.	Отриц.	Ошибка	Запрос функц.	Отношение
Авто	0,735	0,663	0,448	0,635	0,563	0,493
Персонал.	0,645	0,404	0,523	0,623	0,585	0,464
Музыка	0,686	0,389	0,466	0,651	0,381	0,410
Навигация	0,685	0,446	0,377	0,412	0,553	0,435
Медицина	0,655	0,501	0,490	0,672	0,417	0,465
Финансы	0,500	0,717	0,660	0,698	0,538	0,457
Шоппинг	0,741	0,759	0,520	0,721	0,536	0,468
Образов.	0,526	0,457	0,475	0,714	0,708	0,444
Видео	0,674	0,474	0,466	0,698	0,527	0,458
Среднее	0,650	0,534	0,492	0,647	0,534	0,455

Таблица 2.18 — Результаты Trans-LSTM на наборе данных «Google Play» (F_1)

Категория	Функция	Полож.	Отриц.	Ошибка	Запрос функц.	Отношение
Авто	0,724	0,692	0,478	0,625	0,598	0,731
Персонал.	0,721	0,411	0,536	0,684	0,671	0,670
Музыка	0,740	0,718	0,509	0,437	0,346	0,663
Навигация	0,435	0,624	0,527	0,702	0,402	0,668
Медицина	0,577	0,521	0,691	0,714	0,491	0,697
Финансы	0,727	0,762	0,476	0,730	0,545	0,708
Шоппинг	0,737	0,767	0,545	0,781	0,538	0,732
Образов.	0,733	0,553	0,767	0,473	0,671	0,721
Видео	0,683	0,541	0,417	0,712	0,643	0,646
Среднее	0,675	0,621	0,550	0,651	0,545	0,693

Результаты исследования влияния компонентов предложенной модели на качество анализа приведены в таблице 2.19. Наибольшее влияние на качество

Таблица 2.19 — Результаты абляционных экспериментов на данных «Google Play» для Trans-LSTM

Модель	Функция	Описание	Отношение	Среднее
без $\phi(S_t)$	$-8{,}1\%$	$-0{,}5\%$	$-0{,}2\%$	$-2{,}25\%$
без $F(E(w_i))$	$-6{,}4\%$	$-15{,}4\%$	$-16{,}1\%$	$-13{,}48\%$
без $\phi(H_t)$	$-15{,}3\%$	$-14{,}3\%$	$-22{,}0\%$	$-18{,}4\%$

извлечения оказывает исключение признаков истории предсказаний $\phi(H_t)$.

Зависимость качества извлечения частей составных объектов от длины предложения показана на рисунке 2.24. Преложенная модель Trans-LSTM показывает незначительное преимущество при извлечении фрагментов на коротких предложениях и консистентное превосходство над альтернативными моделям для всех рассмотренных длин при определении отношений между ними.

С другой стороны, все рассмотренные модели показывают низкую точность определения отношений между фрагментами при увеличении расстояния между ними, что продемонстрировано на рисунке 2.25. Trans-LSTM имеет преимущество на расстояниях до 5 слов, однако далее уступает обеим представленным моделям.

Результаты исследования влияния фактора запоминания поверхностных форм фрагментов приведено в таблице 2.20. Из них следует, что способность Trans-LSTM к обобщению при извлечении фрагментов, встречающихся только в

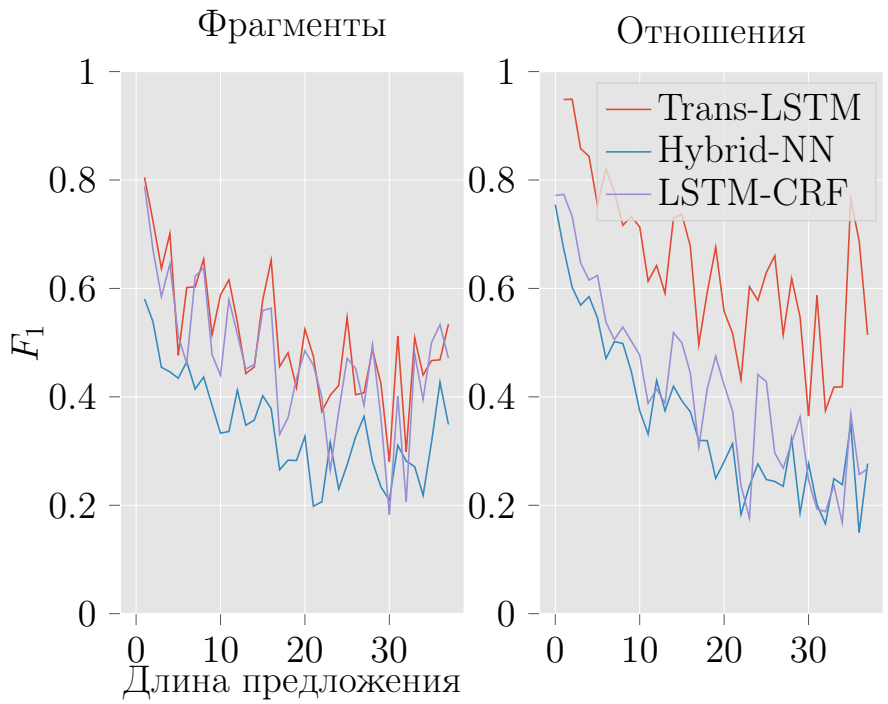


Рисунок 2.24 — Зависимость качества извлечения фрагментов и отношений от длины предложения в обращениях пользователей «Google Play»

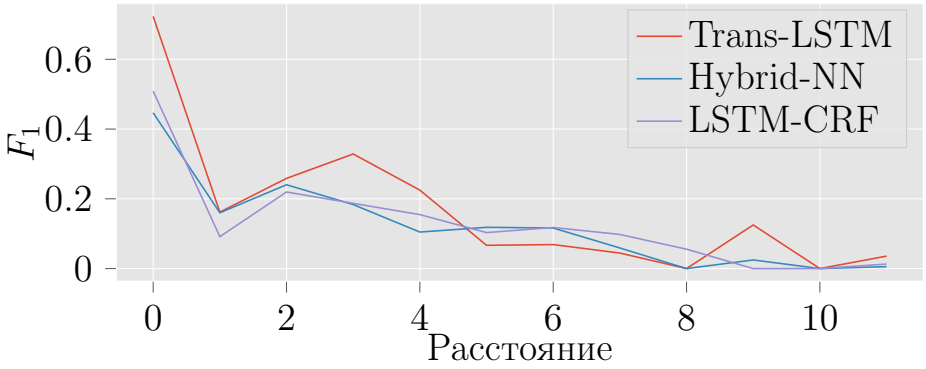


Рисунок 2.25 — Зависимость качества извлечения отношений от расстояния между фрагментами

проверочном множестве текстов выше, чем у альтернатив: преимущество перед LSTM-CRF составляет 7,7% для функций и 7,8% для описаний.

Сравнение предсказаний, совершенных моделью Trans-LSTM (**П**) с эталонной разметкой (**И**), приведено ниже.

Пример 1

И Почему после обновления приложения мне [пришлось заново]$_{O1}$ [привязывать карту]$_{A1}$ МИР и пришлось мне это делать во время заправки!!!

П Почему после обновления приложения мне [пришлось заново]$_{O1}$ [привязывать карту МИР]$_{A1}$ и пришлось мне это делать во время заправки!!!

Пример 2

Таблица 2.20 — Качество извлечения фрагментов в зависимости от упоминания в обучающем наборе на данных «Google Play» (F_1)

Модель	Функция		Описание	
	О и П	**П**	**О и П**	**П**
Hybrid-NN	0,723	0,593	0,392	0,277
LSTM-CRF	0,770	0,605	0,534	0,425
Trans-LSTM	0,780	0,652	0,604	0,458

И В принципе, [приложение]$_{A1}$ [не плохое]$_{O1}$.

П В принципе, [приложение]$_{A1}$ [не плохое]$_{O1}$.

Пример 3

И [Сделайте]$_{O(1,2)}$ [поиск по номеру автомобиля]$_{A1}$, как у Яндекса, или [qr код на лобовом стекле или двери]$_{A2}$ - [иногда очень сложно]$_{O3}$ [найти машину]$_{A3}$ пальцем на карте.

П [Сделайте]$_{O(1,2)}$ [поиск по номеру автомобиля]$_{A1}$, как у Яндекса, или [qr код]$_{A2}$ на лобовом стекле или [двери]$_{A3}$ - [иногда очень сложно найти машину пальцем на карте]$_{O3}$.

Пример 4

И А еще [хорошо бы]$_{O(1,2,3)}$ [напоминалки]$_{A1}$, или [план обследований]$_{A2}$, на какой неделе, на каком сроке [что проходить нужно по плану]$_{A3}$!!

П А еще [хорошо бы]$_{O1}$ [напоминалки]$_{A1}$, или план обследований, на какой неделе, на каком сроке что проходить нужно по плану!!

Пример 5

И [Напрягает необходимость периодически]$_{O1}$ [начинать работу «с начала»]$_{A1}$ после того, как пропадают прежде настроенные экраны рабочего стола – ведь, это ещё один-два часа рутинных по сути операций.

П [Напрягает необходимость периодически]$_{O1}$ [начинать работу «с начала» после того]$_{A1}$, как пропадают прежде настроенные экраны рабочего стола – ведь, это ещё один-два часа рутинных по сути операций.

Пример 6

И Когда даже загрузил плейлист на устройство , [метаданные]$_{A1}$ [локально не сохраняются]$_{O1}$.

П Когда даже загрузил плейлист на устройство , [метаданные]$_{A1}$ [локально не сохраняются]$_{O1}$.

Пример 7

И [Не работает]$_{O1}$ [система чеков и квитанций]$_{A1}$.

П [Не работает]$_{O(1,2)}$ [система чеков]$_{A1}$ и [квитанций]$_{A2}$.

Результаты проведенных экспериментов позволяют говорить о том, что вариант предложенной модели с использованием Bi-LSTM показывает лучшие результаты при решении задачи обработки запросов пользователей ПО по сравнению с рассмотренными альтернативами. Однако предлагаемая модель не имеет преимуществ при определении отношений между удаленными друг от друга фрагментами. Улучшение этого аспекта является перспективным направлением для дальнейших исследований.

Качественный анализ полученных результатов говорит о возможности практического применения предложенной модели для обработки запросов пользователей при эксплуатации и сопровождении программных продуктов.

Выводы ко второй главе

1. Предложенная общая нейросетевая модель на основе системы переходов для извлечения составных объектов и их атрибутов позволяет адаптировать её для извлечения объектов в различных предметных областях путем конкретизации состава множеств типов фрагментов и атрибутов.

2. Экспериментальное исследование нейросетевой модели для извлечения и анализа пользовательских мнений, проведённое на размеченном наборе данных, составленном из русскоязычных отзывов магазина «Ali Express», показало, что предложенная модель превзошла рассмотренные альтернативы по качеству извлечения фрагментов, отношений и атрибутов. Итоговые значения F_1 для модели Trans-LSTM, полученные в ходе кросс-валидации на текстах отзывов о трех категориях товаров, следующие: 0,795 – при извлечении фрагментов, 0,723 – при определении отношений, 0,631 – при определении значений атрибутов. Предложенная модель имеет более высокое по сравнению с альтернативами качество определения значений атрибутов при увеличении длины предложения, что говорит о способности лучше учитывать контекстную информацию при совершении предсказаний. Использованная методика оценки моделей и полученные результаты позволяют сделать вывод о практической пригодности модели для извлечения мнений о товарах из категорий, не включенных в обучающую выборку, без дополнительного обучения.

3. Экспериментальное исследование нейросетевой модели для обработки запросов пользователей на этапе эксплуатации и сопровождения программного продукта, проведенное на размеченном наборе русскоязычных обращений из магазина «Google Play», показало, что предложенная модель превосходит иные рассмотренные альтернативы. Метрика F_1 при извлечении фрагментов равна 0,633, при определении отношений – 0,693. Предложенная модель имеет более высокие по сравнению с альтернативами показатели качества определения отношений в длинных предложениях. Кроме того, она обеспечивает более высокий уровень обобщения при извлечении фрагментов, отсутствующих в обучающей выборке. Использованная методика оценки моделей и полученные результаты позволяют говорить о том, что модель имеет высокую обобщающую способность и может применяться для обработки запросов любых типов приложений.

Глава 3. Практическая апробация и внедрение моделей, алгоритмов и программного обеспечения

3.1 Анализ существующих программных продуктов для обработки естественного языка

В связи с объективным ростом потребности в решении задач обработки текстов на рынке программных продуктов появляются многочисленные программные средства для обработки естественного языка, призванные как упростить решение типичных задач для предварительной обработки текстовых данных, так и предоставить средства для решения конкретных проблем конечных пользователей. Рассмотрим некоторые широко представленные на рынке ПО решения.

Intel NLP Architect

Библиотека для обучения моделей машинного обучения, позволяющая решать широкий спектр задач обработки естественного языка. В частности, реализованы модели для решения следующих задач: токенизация; распознавание именованных сущностей; разбор синтаксических зависимостей; определение намерений и заполнение слотов; анализ тональности; моделирование языка с помощью статистических лингвистических моделей; извлечение отношений; аспектно-ориентированный анализ тональности. Модель аспектно-ориентированного анализа тональности основывается на использовании словарей аспектной и оценочной лексики, которые расширяются за счет использования синтаксических шаблонов и алгоритма двойного распространения, после чего определяется тональность извлеченных аспектов.

На рисунке 3.1 приведен результат работы алгоритма определения тональности аспектов на материале отзывов о мобильных телефонах.

AllenNLP

Открытая библиотека для обработки естественного языка на основе фреймворка PyTorch. Предлагает множество обученных моделей для решения типичных задач обработки естественного языка, включая анализ тональности. Кроме того, библиотека содержит большое число компонентов для построения нейросетевых архитектур (свёрточные и рекуррентные нейронные сети, модели

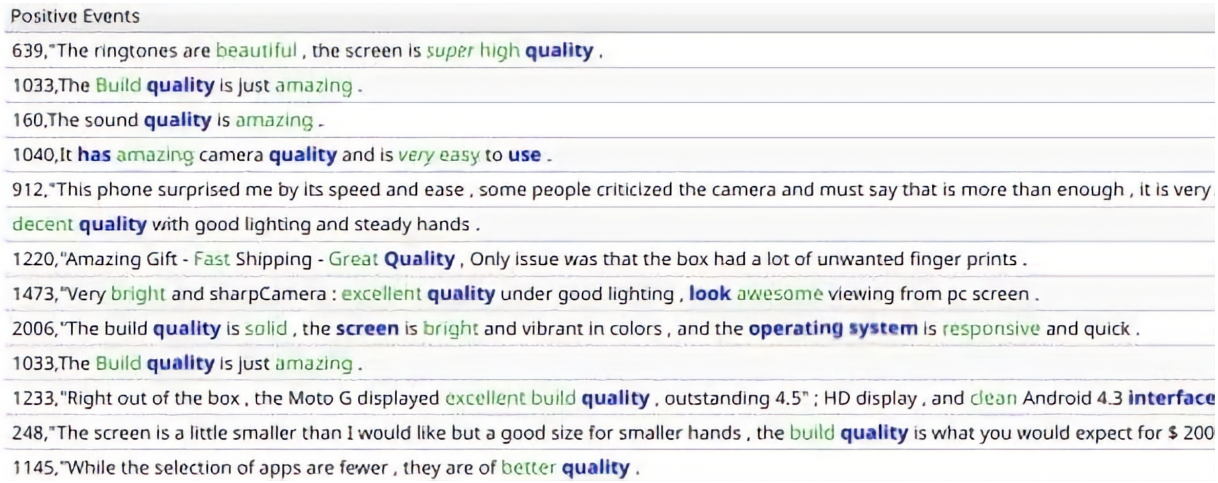


Рисунок 3.1 — Аспектная классификация с помощью NLP Architect

на основе внимания и т. д.) и подготовки исходных данных для их обучения. Реализована на языке Python и доступна под лицензией Apache.

spaCy

Библиотека для обработки естественного языка. Отличается от конкурентов наличием специальных средств для построения многоступенчатых систем анализа текста с возможностью связывать результаты анализа с исходными данными без усилий со стороны программиста. Предлагает собственную реализацию нейросетевых моделей для синтаксического анализа, классификации и извлечения именованных сущностей с прицелом на высокую производительность. Реализована на языках программирования Python/Cython для достижения компромисса между удобством разработки и скоростью работы библиотеки.

Microsoft Language Understanding Intelligent Service (LUIS)

SaaS (Software-as-a-Service) для решения задач анализа естественного языка от компании Microsoft. В частности, предлагаются сервисы для решения следующих задач:

1) Bing Spell Check – проверка и предложение вариантов исправления орфографических и пунктуационных ошибок;

2) peech Services – конвертация текста в речь и речи в текст;

3) Text Analytics – определение тональности, извлечение именованных сущностей и ключевых фраз, определение языка;

4) Azure Bot Service – распознание намерений запросов и заполнение слотов.

Все решения разворачиваются в облачном сервисе Microsoft Azure и доступны через JSON API.

MonkeyLearn

Решение для разработки и обучения алгоритмов обработки естественного языка на основе машинного обучения. Поддерживается обучение моделей классификации и извлечения фрагментов из текстов. Пользователям предлагается графический интерфейс, в котором они могут в режиме онлайн обучать модель решению прикладной задачи на выборке неразмеченных текстов. Решение использует методы активного обучения, которые сокращают объем разметки, необходимой для достижения определенного качества модели. Предлагается широкий спектр интеграций в другие продукты «в один клик». В качестве примера использования решения авторы предлагают систему для аспектно-ориентированного анализа тональности на основе классификатора, как это показано на рисунке 3.2.

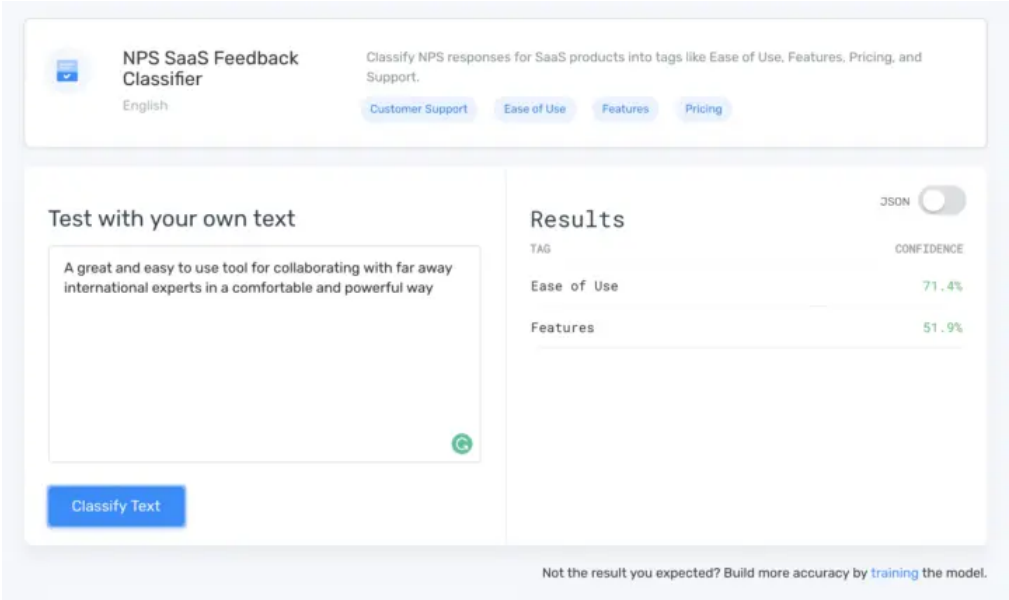


Рисунок 3.2 — Аспектная классификация с помощью MonkeyLearn

Aylien

Предоставляет набор API для решения следующих задач обработки текста: классификация текста по таксономии; определение языка; определение тональности на уровне документа, сущности и аспекта; извлечение именованных сущностей и концептов; суммаризация текста. Предложенная реализация аспектно-ориентированного анализа тональности предполагает наличие фиксированного перечня аспектов в рамках определенной предметной области и не извлекает упоминания аспектных терминов и оценочных фраз. Пример анализа

текста на английском языке приведен на рисунке 3.3. Анализ тональности на

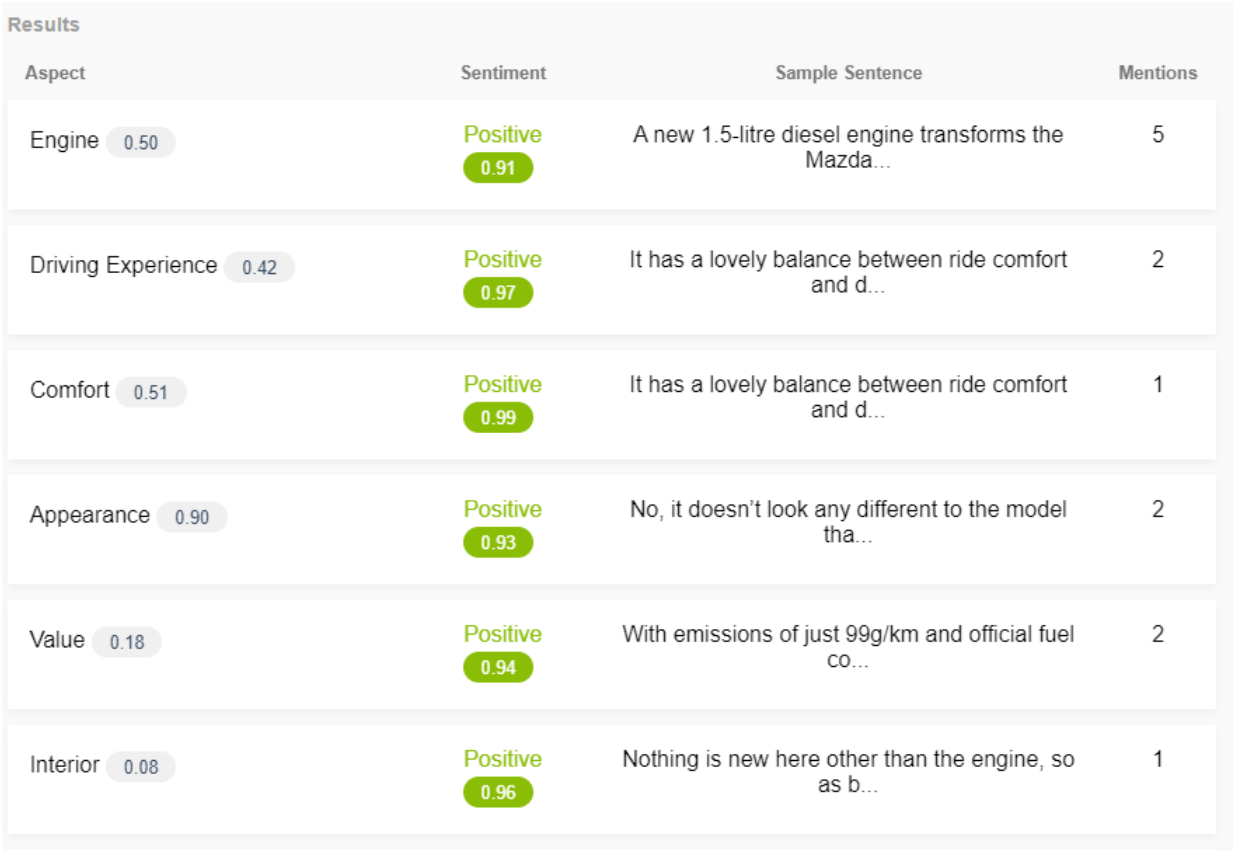


Рисунок 3.3 — Аспектно-ориентированный анализ тональности в Aylien

уровне сущностей действует в два этапа. Сначала из текста с помощью алгоритма извлечения именованных сущностей извлекаются упоминания сущностей, после чего по контексту каждой найденной сущности определяется его тональность в тексте. Пример анализа приведен на рисунке 3.4.

Проведенный анализ показал, что существующие решения на базе открытого исходного кода представлены в основном многофункциональным библиотеками, реализующими модели для обработки естественного языка и компоненты для их обучения и подготовки данных. Хотя многие библиотеки имеют в составе предварительно обученные модели, автору удалось найти всего одну реализацию аспектно-ориентированного анализа тональности, которая работает только с текстами на английском языке. Коммерческие программные решения предоставляются в виде SaaS и предоставляют широкий набор средств для решения прикладных задач обработки языка. Существующие реализации аспектно-ориентированного анализа тональности основаны на моделях классификации текста, что ограничивает их применимость узкими предметными областями с заранее известным количеством аспектов. Кроме того, данная группа решений предполагает размещение и обучение моделей в облаке поставщика

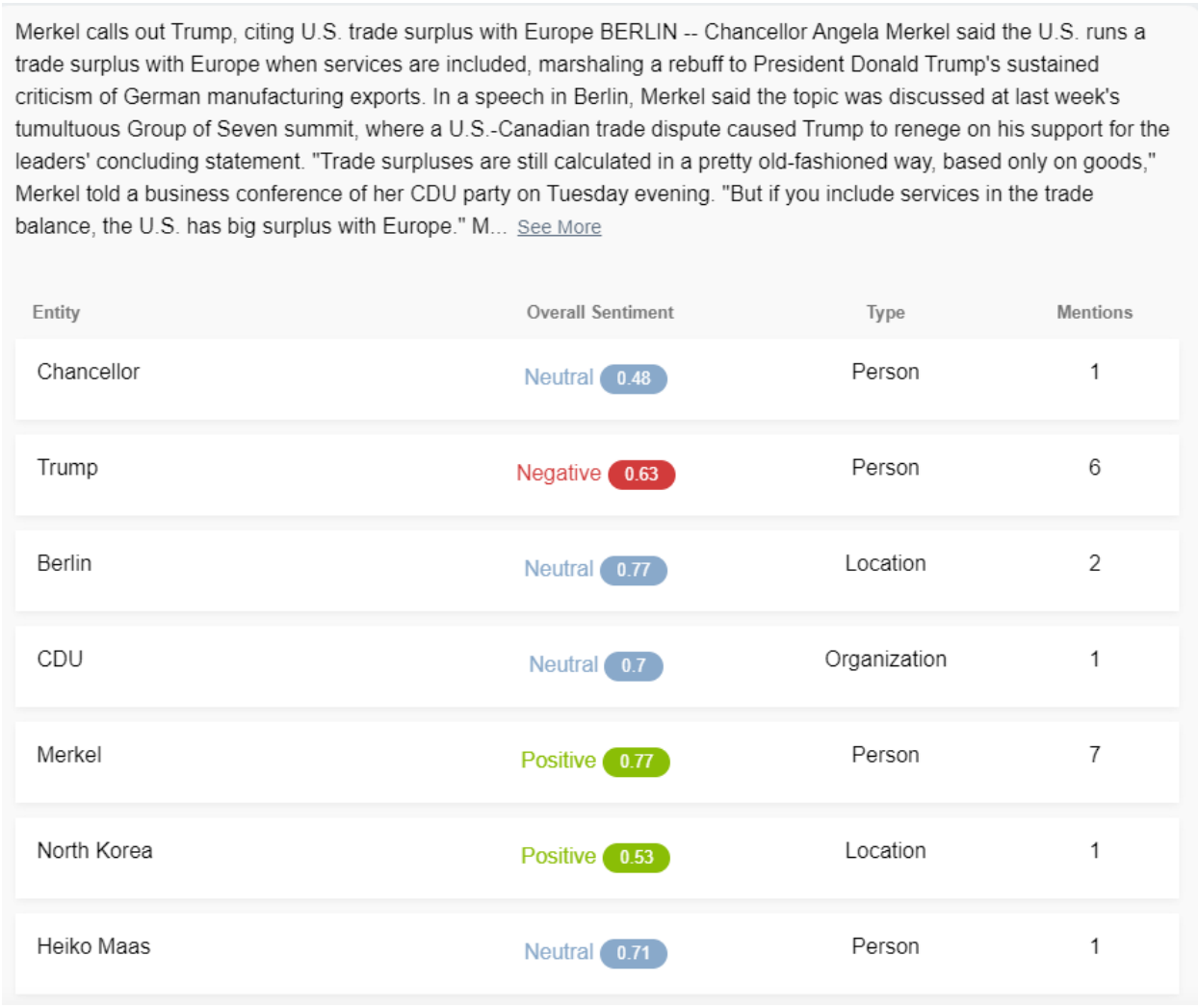


Рисунок 3.4 — Анализ тональности на уровне сущностей в Alyen

решения, что может быть препятствием при работе с данными с повышенными требованиями к безопасности.

В связи с вышеизложенным, было решено реализовать предложенные модели для извлечения структурированной информации о продуктах из текстов пользователей в виде модулей оригинальных программных продуктов.

3.2 Апробация нейросетевой модели на основе системы переходов в задаче извлечения и анализа тональности пользовательских мнений о потребительских свойствах товаров

Предложенная в рамках диссертационного исследования нейросетевая модель для извлечения и анализа тональности мнений пользователей из тек-

стов отзывов была положена в основу программной системы (ПС) «Quiddi Semantics», реализующей следующие функции:

1) сбор и сохранение информации со страниц товаров на сайтах интернет-магазинов, включая элементы описания товара, изображения, отзывы покупателей;

2) извлечение из текстов отзывов мнений, определение их тональности и категоризация;

3) пост-обработка мнений, включающая в себя группировку аспектов по похожести и определение дубликатов для уменьшения избыточности извлекаемой информации;

4) извлечение из текстов отзывов высказываний, содержащих примеры использования товаров для решения возникающих у пользователей проблем;

5) сохранение результатов анализа в базу данных для последующего использования.

ПС состоит из трех подсистем. Так как ПС состоит из подсистем, представленных веб-сервисами с простым внутренним устройством, ПС имеет смысл представить в нотации архитектурных схем Amazon Web Services (AWS)[1]. Итоговая архитектура ПС представлена на рисунке 3.5.

Распределение задач между процессами производится посредством сервиса очереди сообщений, который обеспечивает отказоустойчивое хранение и доставку заданий до процессов. Собранная со страниц магазинов информация о товарах и результаты анализа отзывов сохраняются в реляционную базу данных.

Подсистемы сбора информации и анализа отзывов состоят из набора параллельно работающих процессов, что позволяет повысить скорость работы и надежность системы в целом. Задача подсистемы сбора информации заключается в обходе веб-страниц с описанием продуктов из интернет-магазина, разбор их содержимого, извлечение необходимой информации и сохранение её в базу данных для дальнейшей обработки. Подсистема состоит из набора параллельно работающих краулеров, каждый из которых получает задание на разбор страницы продукта из общей очереди сообщений для задач разбора страниц, разбирает HTML код и извлекает из неё данные, специфичные для каждого магазина: название товара; цена товара; категория товара и его положение в иерархическом каталоге; описание товара от производителя или продавца;

[1]https://aws.amazon.com/ru/architecture/

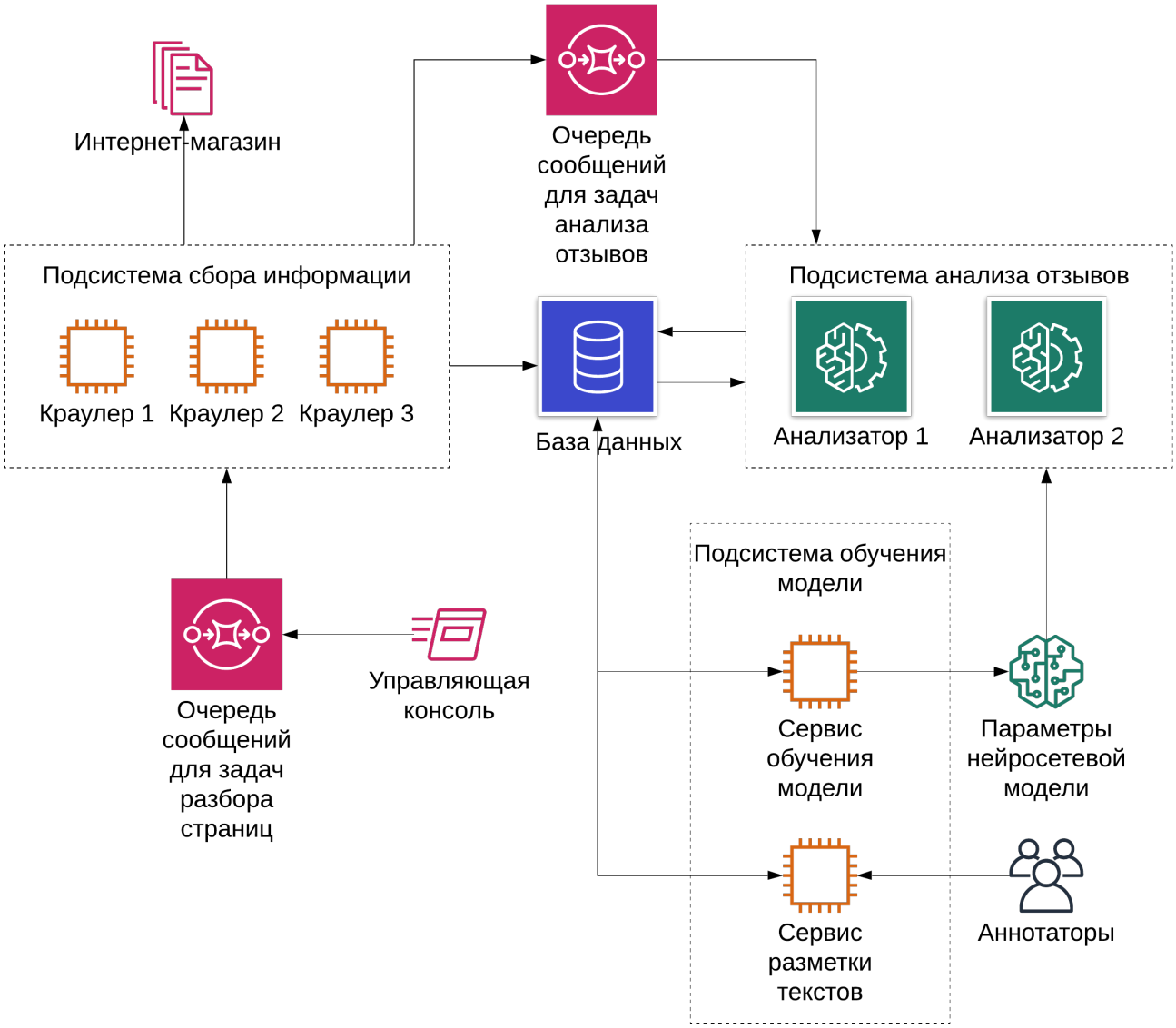


Рисунок 3.5 — Архитектура ПС «Quiddi Semantics».

характеристики товара в форме «ключ-значение», если они имеются; отзывы пользователей, включая текст отзыва и рейтинг товара от покупателя. Для добавления новых задач в очередь используется управляющая консоль.

Так как интернет-магазины различаются структурой веб-страниц, для сбора и извлечения информации из нового источника требуется реализовать соответствующие алгоритмы разбора. Собранные данные о товаре помещаются в общую для всех источников базу данных, а в очередь сообщений для задач анализа отзывов отправляется соответствующее сообщение. Данный сервис реализован на языке программирования PHP[2] с использованием фреймворка

[2]https://www.php.net/

Laravel[3]. В качестве сервиса очереди сообщений используется Amazon Simple Queue Service[4].

Перед использованием нейросетевой модели необходимо собрать корпус текстов отзывов пользователей, разметить его, и на основе размеченного корпуса провести обучение модели. Этот функционал реализует подсистема обучения модели. Модуль разметки текстов предоставляет функционал по хранению, разметке, просмотру и выгрузке текстов отзывов пользователей. В частности, на его базе разработан инструмент, позволяющий осуществлять разметку корпуса нескольким аннотаторам одновременно с удаленных рабочих мест. В нем пользователи могут выделять фрагменты текста и связывать их друг с другом в мнения, а также задавать классы мнений и их тональности. Размеченные тексты сохраняются в базе данных и доступны для выгрузки в формате JSON (JavaScript Object Notation). Кроме того, в данном модуле реализованы правила проверки корректности разметки с выдачей отчета о найденных ошибках.

Модуль обучения модели получает размеченные данные из БД, разделяет их на обучающую и тестовую выборку и запускает процесс оптимизации модели. Также модуль поддерживает возможность подбора оптимального значения гипермараметров модели с помощью процедуры случайного поиска. По окончании обучения, параметры для модели с самой высокой точностью и её гиперпараметры сохраняются для дальнейшего использования в подсистеме анализа отзывов.

Подсистема анализа отзывов предоставляет реализацию предложенной в данной работе нейросетевой модели на основе системы переходов в виде веб-сервиса, использующего параметры модели, предварительно обученные на размеченном корпусе текстов отзывов пользователей. Тексты для анализа в подсистему приходят из очереди сообщений для задач анализа отзывов и распределяются между несколькими доступными процессами анализа. На вход процесс получает набор отзывов для одного продукта в формате JSON в виде пар «идентификатор отзыва – текст отзыва».

Таким образом, результатом работы алгоритма являются сгруппированные и несгруппированные мнения. Важным обстоятельством является то, что при извлечении сохраняется информация о том, из какого отзыва мнение извлечено и на каких позициях располагаются фрагменты мнения. Это позволяет

[3]https://laravel.com/

[4]https://aws.amazon.com/ru/sqs/

выяснить источник любого мнения при его использовании в качестве материала при решении прикладных задач.

Сервис реализован на языке программирования Python 3. Нейросетевая модель на основе системы переходов реализована с использованием фреймворка машинного обучения PyTorch. Предварительная обработка текста осуществляется с помощью фреймворка для обработки естественного языка spaCy. Программная система зарегистрирована в «Реестре программ для ЭВМ Федеральной службы по интеллектуальной собственности» (свидетельство 2019612276 от 14.02.2019), правообладателем является Общество с ограниченной ответственностью «ТомскСофт»[27].

Рассмотренное программное обеспечение были использовано компанией ООО «ТомскСофт» при разработке агрегаторов товаров «Quiddi.ru» (на русском языке) и «Top Rank Products» (на английском языке). Агрегаторы предоставляет информацию о товарах из магазина «AliExpress», включающую основные характеристики товара, динамику изменения его цены, оценки и отзывы пользователей. Монетизация агрегатора осуществляется путем аффилиации с «AliExperss», по условиям которой аффилиат получает процент с каждой покупки, совершенной в магазине после перехода в него по специальной ссылке. При такой модели монетизации основной задачей аффилиата является предоставление некоторой дополнительной информации о товарах, которая побудит пользователей прийти на сайт магазина и купить товар через предоставленную ссылку. Данная информация генерируется автоматически на основе данных, извлекаемых разработанной нейросетевой моделью извлечения и анализа тональности мнений пользователей из текстов. Извлеченные и сгруппированные мнения используются в следующих информационных блоках:

– *что говорят покупатели* – самые популярные положительные и отрицательные мнения типа «Товар», извлеченные из отзывов;

– *соответствие товара ожиданиям покупателя* – оценка качества самого товара, рассчитанная как доля положительных мнений типа «Товар»;

– *качество доставки* – оценка качества упаковки и скорости доставки товара, рассчитанная как доля положительных мнений типа «Доставка»;

– *качество обслуживания* – оценка качества общения с продавцом, рассчитанная как доля положительных мнений типа «Продавец»;

– *рейтинг товара* – интегральная оценка, полученная на основе оценок качества товара, доставки, и обслуживания;

– *цитаты из отзывов* – наиболее информативные для потенциальных покупателей предложения, выбранные с помощью классификатора на основе машинного обучения.

На рисунке 3.6 представлен скриншот части страницы с описанием товара, содержащий дополнительную информацию, полученную на основе анализа текста отзывов.

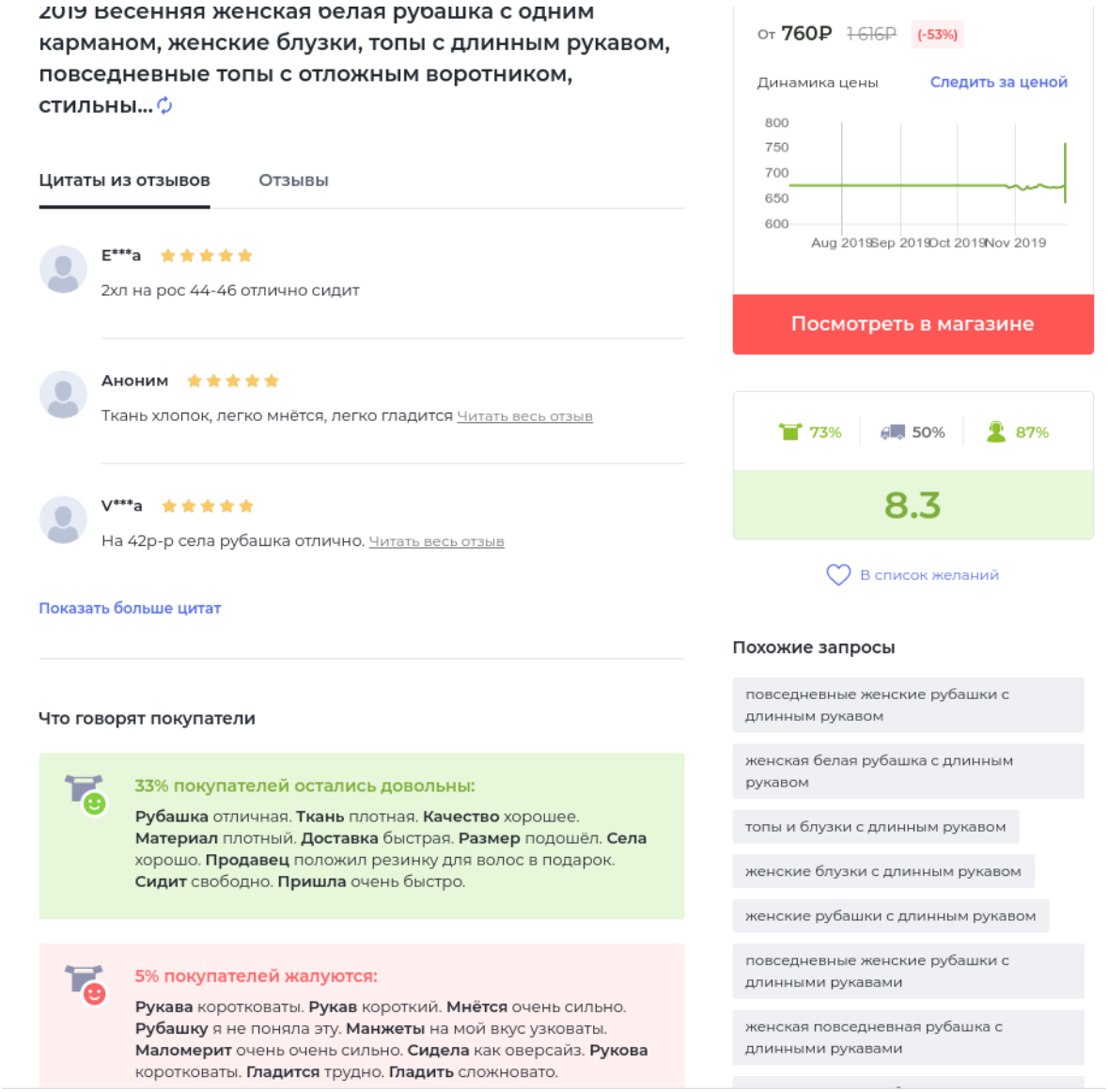


Рисунок 3.6 — Пример страницы с автоматически сгенерированным описанием товара.

На момент написания диссертации программная система была использована для анализа отзывов о 5 миллионах продуктов магазина «AliExpress». Внедрение нейросетевой модели в составе программной системы «Quiddi Semantics» позволило компании «ТомскСофт» открыть новое направление

бизнеса по оказанию информационных услуг потенциальным покупателям магазина «AliExpress». На момент написания диссертации проводится рекламная кампания по продвижению агрегатора в социальных сетях «Одноклассники» и «ВКонтакте», а также на платформах «Pinterest»,«Яндекс.Дзен», «Пульс Mail.ru».

3.3 Апробация нейросетевой модели на основе системы переходов в задаче обработки запросов пользователей на этапе эксплуатации и сопровождения программного продукта

Нейросетевая модель на основе системы переходов для обработки запросов пользователей на этапе эксплуатации программного продукта положена в основу прототипа программной системы «Quiddi Support Analyst», реализующей следующие функции:

1) сбор обращений пользователей, приходящий на электронные почтовые адреса сотрудников службы поддержки;

2) извлечение из текстов обращений запросов, жалоб и предложений пользователей и их классификация;

3) пост-обработка извлеченной информации, включающая в себя группировку объектов по похожести и определение дубликатов для уменьшения избыточности извлекаемой информации;

4) генерация отчета, содержащего сведения о степени удовлетворенности пользователей отдельными функциями программного обеспечения, о возникающих ошибках, о наиболее запрашиваемых функциях.

ПС реализована в виде микросервисной архитектуры и состоит из нескольких веб-сервисов. Архитектура ПС представлена на рисунке 3.7 в нотации архитектурных схем AWS.

Сервис сбора обращений регулярно связывается с почтовым сервером службы поддержки, выгружает поступившие сообщения с помощью протокола IMAP и сохраняет в базе данных текст и временной штамп для последующей обработки. Сервисы разметки и обучения модели по своим функциям совпадают аналогичным сервисам, описанным в разделе 3.1.

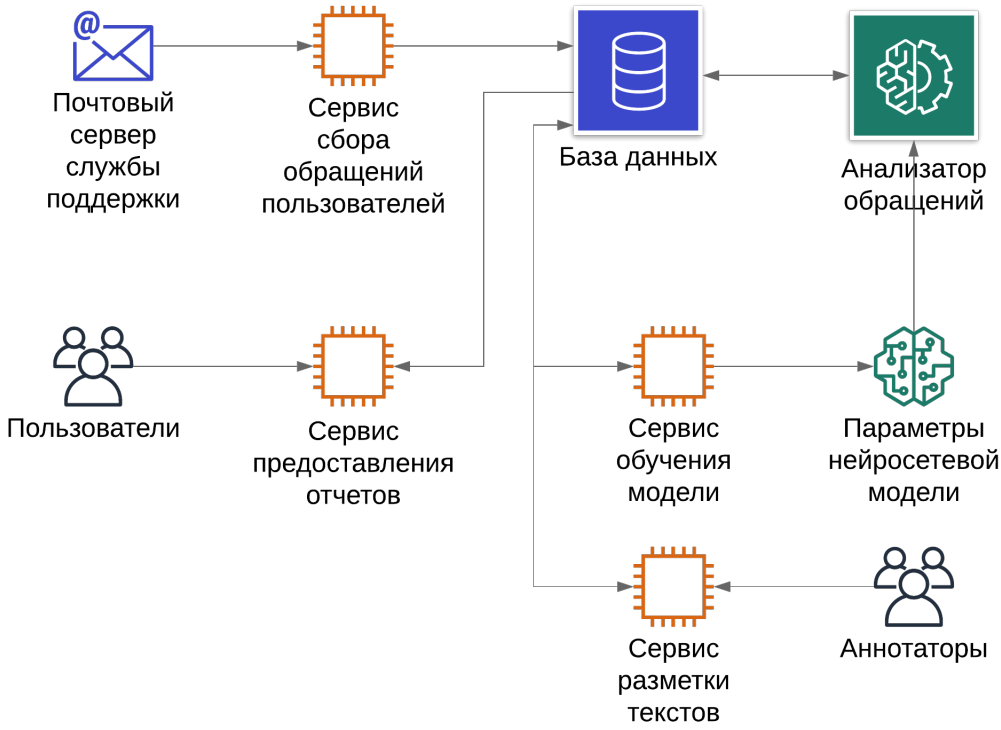


Рисунок 3.7 — Архитектура ПС «Quiddi Support Analyst».

Анализатор сообщений реализует нейросетевую модель на основе системы переходов, при помощи которой извлекаются положительные и отрицательные мнения о функциях программ, запросы на добавление новых или доработку старых функций, жалобы на проблемы при эксплуатации программы. Результат обработки сохраняется в БД.

Сервис предоставления отчетов предназначен для генерации и предоставления пользователям ПС отчетов о результатах анализа текстов обращений в службу поддержки. Поддерживается три типа отчетов, которые на момент написания диссертации представлены структурами в формате JSON.

Отчет «Мнения пользователей о функционале программного обеспечения» содержит часто упоминаемые пользователями функции и соотношение положительных/отрицательных мнений. Возможна сортировка мнений по следующим критериям: название объекта мнения в лексикографическом порядке, степень преобладания тональности (положительной или негативной), общее количество обращений с упоминанием объекта мнения.

Отчет «Динамика изменения мнения пользователей о функционале программного обеспечения» отображает изменение соотношения тональности мнений об определенном объекте мнения во времени в виде временного ряда. Кроме того, выдается базовая статистическая информация: количество новых

мнений об объекте за определенный период, направление тренда. Также определяются точки резкого изменения тренда в тональности.

Отчет «Ошибки в функционировании ПО» предоставляет список наиболее часто встречающихся жалоб на сбои в работе ПО. Поддерживается группировка извлеченных упоминаний ошибок по функциям ПО и по семантической схожести описания ошибки.

Объединение в семантические группы производится следующим образом. Сначала фразы с жалобами преобразуются в распределенные векторные представления с помощью модели LASER [124]. Затем размерность полученных векторов понижается алгоритмом UMAP [125] до 50. Уменьшение размерности помогает улучшить качество кластеризации, так как в пространствах высокой размерности алгоритмы кластеризации испытывают сложности из-за «проклятья размерности» [126]. После уменьшения размерности, полученные вектора делятся на группы алгоритмом HDBSCAN [126; 127]. Особенностью данного алгоритма является возможность выявлять примеры, не относящиеся ни к одному из кластеров. Реализованная таким образом семантическая группировка может быть полезна при поиске ошибок, имеющих схожее текстовое описание и, возможно, одну причину.

Отчет «Запросы пользователей на новый функционал» отображает самые популярные запросы пользователей на расширение старых и добавление новых функций в ПО. Запросы объединены в семантические группы по описанному выше алгоритму с целью выделить наиболее важные группы требований и учесть эту информацию при планировании разработки.

На данный момент ПС «Quiddi Support Analyst» находится в стадии рабочего прототипа, на который получено свидетельство о регистрации программы для ЭВМ №2020614799 от 24.04.2020[28].

В дальнейшем планируется доработка проекта, включающая реализацию веб-интерфейса для просмотра отчетов, реализация сбора информации из других источников: «Google Play», «App Store».

Выводы к третьей главе

1) Анализ существующего свободного и коммерческого ПО показал, что рассмотренные альтернативы хотя и предоставляют широкий набор средств для решения типичных задач анализа естественного языка, не имеют развитых инструментов для извлечения структурированной информации о продуктах и услугах из текстов пользователей.

2) На основе результатов проведенного было принято решение о реализации предложенных моделей извлечения структурированной информации из текстов пользователей в виде оригинальных программных продуктов: «Quiddi Semantics» для сбора и анализа отзывов покупателей о товарах, представленных в интернет-магазине «Ali Express»; «Quiddi Support Analyst» для анализа обращений пользователей в службу технической поддержки программного продукта.

3) Результаты практической апробации предложенных нейросетевых моделей, реализованных в виде компонентов программных систем «Quiddi Semantics» и «Quiddi Support Analyst» позволяют сделать вывод о их работоспособности и практической пригодности при решении задач по извлечению мнений и запросов пользователей из текстов, что подтверждается актом внедрения и использовании программных систем в компании ООО «ТомскСофт».

Заключение

В результате выполнения диссертационной работы были получены следующие теоретические и практические результаты.

1. Проведен анализ современного состояния исследования в области анализа пользовательских текстов на естественном языке, в ходе которого выявлено преимущество методов, направленных на извлечение информации о продуктах в структурированной форме; обозначена проблема низкой точности описанных в научной литературе методов, обусловленная использованием многокомпонентных моделей. Сделан вывод о необходимости использования методов, позволяющих предсказывать структуру объекта в рамках единой модели в задачах, связанных с извлечением структурированной информации о продуктах из текстов.

2. Обозначены особенности употребления языка в текстах пользователей о продуктах, которые оправдывают применение современных нейросетевых методов. Рассмотрены существующие методы структурного предсказания и их преимущества перед альтернативами. В частности, выделен класс методов, использующий системы переходов, который позволяет использовать стандартные методы обучения и вывода нейронных сетей, обеспечивая при этом возможность выражать сложные структурные признаки путем использования конфигураций – структур данных, хранящих промежуточное представление предсказываемого объекта.

3. Предложена оригинальная нейросетевая модель на основе системы переходов для извлечения составных объектов и их атрибутов из текстов на естественном языке, позволяющая одновременно извлекать фрагменты объектов и определять взаимосвязи между ними с возможностью адаптации к конкретной предметной области через задание множеств, определяющих смысловое наполнение фрагментов составных объектов и их атрибутов.

4. Предложенная модель легла в основу оригинальной нейросетевой модели для извлечения и анализа мнений из текстов пользовательских отзывов о продуктах, отличающаяся от известных моделей использованием подхода на основе системы переходов. Для экспериментального исследования модели подготовлен набор данных, состоящий из текстов отзывов интернет-магазина «Ali Express» на русском языке. Исследование модели показало более вы-

сокое качество извлечения мнений предложенной моделью по сравнению с рассмотренными альтернативами. Качественный анализ полученных результатов говорит о возможности практического применения данной модели для извлечения и анализа мнений пользователей о потребительских свойствах товаров.

5. Предложенная модель легла в основу оригинальной нейросетевой модели для анализа запросов пользователей на этапе эксплуатации и сопровождения программного продукта, отличающаяся от известных моделей использованием подхода на основе системы переходов. Для проведения экспериментального исследования модели предложен набор запросов из магазина приложений «Google Play Market» на русском языке. Результаты экспериментального исследования позволяют говорить о более высоком качестве анализа запросов предложенной моделью по сравнению с альтернативами.

6. Предложенные нейросетевые модели были положены в основу разработанных программных систем «Quiddi Semantics»(свидетельство о регистрации программы для ЭВМ №2019612276 от 14.02.2019) и «Quiddi Support Analyst»(свидетельство о регистрации программы для ЭВМ №2020614799 от 24.04.2020). Программные системы внедрены и используются в компании ООО «ТомскСофт».

7. Результаты диссертационного исследования использованы в ФГБОУ ВО «ТУСУР» при выполнении государственного задания Министерства науки и высшего образования РФ, проект FEWM-2020-0036 «Методологическое и инструментальное обеспечение принятия решений в задачах управления социально-экономическими системами и процессами в гетерогенной информационной среде»; в учебном процессе кафедры автоматизации обработки информации (АОИ) при чтении курса лекций и проведении практических занятий по дисциплинам «Интеллектуальные вычислительные системы», «Анализ больших данных» при подготовке магистров по направлению 09.04.04 — «Программная инженерия».

В заключение автор выражает благодарность и большую признательность научному руководителю Ехлаков Ю. П. за поддержку, помощь, обсуждение результатов и научное руководство. Также автор благодарит Безходранова И. В. за возможность совмещения исследовательской и прикладной деятельности во время работы в ООО «ТомскСофт», Трошина М. В. за продуктивное обсуждение практических аспектов работы и разработку сервиса для разметки

текстов, авторов шаблона *Russian-Phd-LaTeX-Dissertation-Template* за помощь в оформлении диссертации. Особая благодарность родителям, близким друзьям и коллегам за моральную поддержку в сложный период написания диссертационной работы.

Список сокращений и условных обозначений

CRF	conditional random field, условное случайное поле
CNN	convolutional neural network, свёрточная нейронная сеть
RNN	recurrent neural network, рекуррентная нейронная сеть
LSTM	long short-term memory, долгая краткосрочная память
ReLU	rectified linear unit, линейный выпрямитель
SVM	support vector machine, метод опорных векторов

Список литературы

1. *Литневская, Е. И.* О некоторых графико-орфографических особенностях письменных жанров разговорной речи / Е. И. Литневская // Вестник Московского университета. Серия 9 Филология. — 2009. — № 6. — С. 65—76.

2. *Литневская, Е. И.* О некоторых графико-орфографических особенностях письменных жанров разговорной речи / Е. И. Литневская // Научный вестник Московского государственного технического университета гражданской авиации. — 2008. — № 137. — С. 101—105.

3. *Pang, B.* Seeing stars: exploiting class relationships for sentiment categorization with respect to rating scales / B. Pang, L. Lee // Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics. — Stroudsburg : ACL, 2005. — P. 115—124.

4. *Pang, B.* Thumbs up? Sentiment Classification using Machine Learning Techniques / B. Pang, L. Lee, S. Vaithyanathan // Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing. — Stroudsburg : ACL, 2002. — P. 79—86.

5. *Turney, P. D.* Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews / P. D. Turney // Proceedings of the 40th Annual Meeting on Association for Computational Linguistics. — Stroudsburg : ACL, 2002. — P. 417—424.

6. *Лукашевич, Н. В.* Извлечение и использование оценочных слов в задаче классификации отзывов на три класса / Н. В. Лукашевич, И. И. Четверкин // Вычислительные методы и программирование: новые вычислительные технологии. — 2011. — № 2. — С. 73—81.

7. *Лукашевич, Н. В.* Построение модели для извлечения оценочной лексики в различных предметных областях / Н. В. Лукашевич, И. И. Четверкин // Моделирование и анализ информационных систем. — 2013. — № 2. — С. 70—79.

8. *Русначенко, Н. Л.* Методы интеграции лексиконов в машинное обучение для систем анализа тональности / Н. Л. Русначенко, Н. В. Лукашевич // Искусственный интеллкт и принятие решений. — 2017. — № 2. — С. 78—89.

9. *Тутубалина, Е. В.* Извлечение проблемных высказываний, связанных с неисправностями и нарушением функциональности продуктов, на основе отзывов пользователей / Е. В. Тутубалина // Вестник Казанского государственного технического университета им. А.Н. Туполева. — 2015. — Т. 71, № 3. — С. 139—146.

10. *Тутубалина, Е. В.* Совместная вероятностная тематическая модель для идентификации проблемных высказываний, связанных нарушением функциональности продуктов / Е. В. Тутубалина // Труды Института системного программирования РАН. — 2015. — Т. 27, № 4. — С. 111—128.

11. *Jebbara, S.* Aspect-Based Relational Sentiment Analysis Using a Stacked Neural Network Architecture / S. Jebbara, P. Cimiano // Proceedings of the 22nd European Conference on Artificial Intelligence. Vol. 285. — Amsterdam : IOS Press, 2016. — P. 1123—1131. — URL: https://doi.org/10.3233/978-1-61499-672-9-1123.

12. Opinion Mining on the Web by Extracting Subject-Aspect-Evaluation Relations / N. Kobayashi [et al.] // Proceedings of AAAI Spring Symposium: Computational Approaches to Analyzing Weblogs. — Menlo Park : AAAI Press, 2006. — P. 86—91.

13. *McDonald, R.* Characterizing the Errors of Data-Driven Dependency Parsing Models / R. McDonald, J. Nivre // Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL). — Stroudsburg : ACL, 2007. — P. 122—131.

14. Effects of Parsing Errors on Pre-Reordering Performance for Chinese-to-Japanese SMT / D. Han [et al.] // Proceedings of the 27th Pacific Asia Conference on Language, Information, and Computation (PACLIC 27). — Taipei : NCU, 2013. — P. 267—276.

15. *Lopresti, D.* Optical Character Recognition Errors and Their Effects on Natural Language Processing / D. Lopresti // Proceedings of the Second Workshop on Analytics for Noisy Unstructured Text Data. — N. Y. : ACM, 2008. — P. 9—16.

16. Transition-Based Dependency Parsing with Stack Long Short-Term Memory / C. Dyer [et al.] // Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers). — Stroudsburg : ACL, 2015. — P. 334—343.

17. *Nivre, J.* An Efficient Algorithm for Projective Dependency Parsing / J. Nivre // Proceedings of the Eighth International Conference on Parsing Technologies. — 2003. — P. 149—160.

18. Deep Contextualized Word Embeddings in Transition-Based and Graph-Based Dependency Parsing - A Tale of Two Parsers Revisited / A. Kulmizev [et al.] // Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing. — Stroudsburg : ACL, 2019. — P. 2755—2768.

19. *Ехлаков, Ю. П.* Модель извлечения пользовательских мнений о потребительских свойствах товара на основе рекуррентной нейронной сети / Ю. П. Ехлаков, Е. И. Грибков // Бизнес-информатика. — 2018. — Т. 46, № 4. — С. 7—16.

20. *Грибков, Е. И.* Нейросетевая модель для обработки запросов пользователей на этапе эксплуатации и сопровождения программного продукта / Е. И. Грибков, Ю. П. Ехлаков // Бизнес-информатика. — 2020. — Т. 14, № 1. — С. 7—18.

21. *Грибков, Е. И.* Нейросетевая модель на основе системы переходов для извлечения составных объектов и их атрибутов из текстов на естественном языке / Е. И. Грибков, Ю. П. Ехлаков // Доклады ТУСУР. — 2020. — Т. 23, № 1. — С. 47—52.

22. *Грибков, Е. И.* Нейросетевая модель на основе системы переходов для извлечения и анализа тональности пользовательских мнений / Е. И. Грибков, Ю. П. Ехлаков // Искусственный интеллект и принятие решений. — 2020. — № 1. — С. 99—110.

23. *Грибков, Е. И.* Набор данных и модель глубокого обучения для анализа текстов отзывов пользователей / Е. И. Грибков, Ю. П. Ехлаков // Наука. Технологии. Инновации. Сборник научных трудов. В 9-ти частях. — Новосибирск : НГТУ, 2018. — С. 180—184.

24. *Грибков, Е. И.* Модель обработки запросов пользователей на этапе эксплуатации и сопровождения программного продукта / Е. И. Грибков, Ю. П. Ехлаков // Электронные средства и системы управления. Материалы докладов Международной научно-практической конференции. — Томск : В-Спектр, 2019. — С. 141—143.

25. *Грибков, Е. И.* Модель извлечения структурированных объектов и их атрибутов из текстов на естественном языке / Е. И. Грибков // Сборник избранных статей научной сессии ТУСУРа (Томск, 22–24 мая 2019 г.): в 2 ч. — Томск : В-Спектр, 2019. — С. 54—56.

26. *Грибков, Е. И.* Модель на основе системы переходов для извлечения составных объектов из текстов / Е. И. Грибков, Ю. П. Ехлаков // Сборник избранных статей научной сессии ТУСУР, Томск, 13–30 мая 2020. — Томск : В-Спектр, 2020. — С. 52—55.

27. «Quiddi Semantics» / Е. И. Грибков [и др.] // Свидетельство о государственной регистрации программы для ЭВМ №2019612276 от 14.02.2019.

28. «Quiddi Support Analyst» / Е. И. Грибков [и др.] // Свидетельство о государственной регистрации программы для ЭВМ №2020614799 от 24.04.2020.

29. *Котлер, Ф.* Основы маркетинга. Краткий курс: Пер. с англ. / Ф. Котлер. — М. : Вильямс, 2007. — 656 с.

30. Identifying Restaurant Features via Sentiment Analysis on Yelp Reviews [Электронный ресурс] / B. Yu [et al.]. — 2017. — URL: https://arxiv.org/abs/1709.08698.

31. *Maynard, D.* Automatic detection of political opinions in tweets / D. Maynard, A. Funk // Proceedings of the 8th International Conference on the Semantic Web. — 2011. — P. 88—99.

32. *Hu, M.* Mining and Summarizing Customer Reviews / M. Hu, B. Liu // Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. — N.Y. : ACM, 2004. — P. 168—177.

33. *Kim, S.* Determining the sentiment of opinions / S. Kim, E. Hovy // Proceedings of the 20th Interntional conference on Computational Linguistics. — Geneva : COLING, 2004. — P. 1367—1373.

34. *Mohammad, S.* Generating High-Coverage Semantic Orientation Lexicons From Overtly Marked Words and a Thesaurus / S. Mohammad, C. Dunne, B. Dorr // Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing. — Stroudsburg : ACL, 2009. — P. 599—608.

35. *Hatzivassiloglou, V.* Predicting the Semantic Orientation of Adjectives / V. Hatzivassiloglou, K. R. McKeown // Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics and 8th Conference of the European Chapter of the Association for Computational Linguistics. — Stroudsburg : ACL, 1997. — P. 174—181.

36. *Di Caro, L.* Sentiment analysis via dependency parsing / L. Di Caro, M. Grella // Computer Standards and Interfaces. — 2013. — Vol. 35, no. 5. — P. 442—453.

37. *Thomas, B.* Synthesized feature space for multiclass emotion classification / B. Thomas, K. A. Dhanya, P. Vinod // 2014 First International Conference on Networks Soft Computing. — Piscataway : IEEE, 2014. — P. 188—192.

38. *Moraes, R.* Document-level sentiment classification: An empirical comparison between SVM and ANN / R. Moraes, J. F. Valiati, W. P. G. Neto // Expert Systems with Applications. — 2013. — Vol. 40, no. 2. — P. 621—633.

39. *Kim, Y.* Convolutional Neural Networks for Sentence Classification / Y. Kim // Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing. — Stroudsburg : ACL, 2014. — P. 1746—1751.

40. *Yu, H.* Towards Answering Opinion Questions: Separating Facts from Opinions and Identifying the Polarity of Opinion Sentences / H. Yu, V. Hatzivassiloglou // Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing. — Stroudsburg : ACL, 2003. — P. 129—136.

41. Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank / R. Socher [et al.] // Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing. — Stroudsburg : ACL, 2013. — P. 1631—1642.

42. *Kotzias D. and Denil, M.* Deep Multi-Instance Transfer Learning [Электронный ресурс] / M. Kotzias D. and Denil. — 2017. — URL: https://arxiv.org/abs/1411.3128.

43. *Schouten, K.* Survey on Aspect-Level Sentiment Analysis / K. Schouten, F. Frasincar // IEEE Transactions on Knowledge and Data Engineering. — 2016. — Vol. 28, no. 3. — P. 813—830.

44. *Hu, M.* Mining Opinion Features in Customer Reviews / M. Hu, B. Liu // Proceedings of the 19th National Conference on Artifical Intelligence. — Menlo Park : AAAI Press, 2004. — P. 755—760.

45. *Hai, Z.* Implicit Feature Identification via Co-occurrence Association Rule Mining / Z. Hai, K. Chang, J. Kim // Computational Linguistics and Intelligent Text Processing / ed. by A. F. Gelbukh. — Heidelberg : Springer, 2011. — P. 393—404.

46. Red Opal: Product-Feature Scoring from Reviews / C. Scaffidi [et al.] // Proceedings of the 8th ACM Conference on Electronic Commerce. — N. Y. : ACM, 2007. — P. 182—191.

47. Generalizing Syntactic Structures for Product Attribute Candidate Extraction / Y. Zhao [et al.] // Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics. — Stroudsburg : ACL, 2010. — P. 377—380.

48. Expanding Domain Sentiment Lexicon through Double Propagation / G. Qiu [et al.] // Proceedings of the 21st International Jont Conference on Artifical Intelligence. — San Francisco : Morgan Kaufmann, 2009. — P. 1199—1204.

49. Extracting and Ranking Product Features in Opinion Documents / L. Zhang [et al.] // Proceedings of the 23rd International Conference on Computational Linguistics: Posters. — Stroudsburg : ACL, 2010. — P. 1462—1470.

50. Opinion Word Expansion and Target Extraction through Double Propagation / G. Qiu [et al.] // Computational Linguistic. — 2011. — Vol. 37, no. 1. — P. 9—27.

51. *Jakob, N.* Extracting Opinion Targets in a Single and Cross-Domain Setting with Conditional Random Fields / N. Jakob, I. Gurevych // Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing. — Stroudsburg : ACL, 2010. — P. 1035—1045.

52. *Liu, P.* Fine-grained Opinion Mining with Recurrent Neural Networks and Word Embeddings / P. Liu, S. Joty, H. Meng // Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing. — Stroudsburg : ACL, 2015. — P. 1433—1443.

53. *Blei, D. M.* Latent Dirichlet allocation / D. M. Blei, A. Ng, M. Jordan // Journal of Machine Learning Research. — 2003. — Vol. 3. — P. 993—1022.

54. Multi-Aspect Sentiment Analysis with Topic Models / B. Lu [et al.] // Proceedings of the 2011 IEEE 11th International Conference on Data Mining Workshops. — Piscataway : IEEE, 2011. — P. 81—88.

55. *Zhan, T.-J.* Semantic Dependent Word Pairs Generative Model for Fine-Grained Product Feature Mining / T.-J. Zhan, C.-H. Li // Proceedings of the Advances in Knowledge Discovery and Data Mining. — Berlin : Springer, 2011. — P. 460—475.

56. *Moghaddam, S.* The FLDA Model for Aspect-Based Opinion Mining: Addressing the Cold Start Problem / S. Moghaddam, M. Ester // Proceedings of the 22nd International Conference on World Wide Web. — N. Y. : ACM, 2013. — P. 909—918.

57. *Moghaddam, S.* Opinion Digger: An Unsupervised Opinion Miner from Unstructured Product Reviews / S. Moghaddam, M. Ester // Proceedings of the 19th ACM International Conference on Information and Knowledge Management. — N. Y : ACM, 2010. — P. 1825—1828.

58. Multi-Aspect Opinion Polling from Textual Reviews / J. Zhu [et al.] // Proceedings of the 18th ACM Conference on Information and Knowledge Management. — N. Y. : Association for Computing Machinery, 2009. — P. 1799—1802.

59. Aspect Ranking: Identifying Important Product Aspects from Online Consumer Reviews / J. Yu, M. Zha Z.-J. and Wang, T.-S. Chua // Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies. — Stroudsburg : ACL, 2011. — P. 1496—1505.

60. Structure-Aware Review Mining and Summarization / F. Li [et al.] // Proceedings of the 23rd International Conference on Computational Linguistics. — Beijing : Tsinghua University Press, 2010. — P. 653—661.

61. *Jin, W.* OpinionMiner: A Novel Machine Learning System for Web Opinion Mining and Extraction / W. Jin, H. H. Ho, R. K. Srihari // Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. — N. Y. : ACM, 2009. — P. 1195—1204.

62. *Laddha, A.* Extracting Aspect Specific Opinion Expressions / A. Laddha, A. Mukherjee // Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing. — Stroudsburg : ACL, 2016. — P. 627—637.

63. *Klinger, R.* The USAGE review corpus for fine grained multi lingual opinion analysis / R. Klinger, P. Cimiano // Proceedings of the 9th International Conference on Language Resources and Evaluation. — Amsterdam : ELRA, 2014. — P. 2211—2218.

64. *Zhang, X.* Character-level Convolutional Networks for Text Classification / X. Zhang, J. J. Zhao, Y. LeCun // Proceedings of an Advances in Neural Information Processing Systems 28. — Red Hook : Curran Associates, 2015. — P. 649—657.

65. Hierarchical Attention Networks for Document Classification / Z. Yang [et al.] // Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. — Stroudsburg : ACL, 2016. — P. 1480—1489.

66. *Dozat, T.* Deep Biaffine Attention for Neural Dependency Parsing / T. Dozat, C. D. Manning // Proceedings of the 5th International Conference on Learning Representations. — OpenReview.net, 2017. — URL: https://openreview.net/forum?id=Hk95PK9le.

67. *Zhang, Y.* Stack-propagation: Improved Representation Learning for Syntax / Y. Zhang, D. Weiss // Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). — Stroudsburg : ACL, 2016. — P. 1557—1566.

68. On the Properties of Neural Machine Translation: Encoder-Decoder Approaches / K. Cho [et al.] // Proceedings of SSST@EMNLP 2014, 8th Workshop on Syntax, Semantics and Structure in Statistical Translation. — Stroudsburg : ACL, 2014. — P. 103—111.

69. *Bahdanau, D.* Neural Machine Translation by Jointly Learning to Align and Translate / D. Bahdanau, K. Cho, Y. Bengio // Proceedings of the 3rd International Conference on Learning Representations. — 2015. — URL: http://arxiv.org/abs/1409.0473.

70. Attention is All You Need / A. Vaswani [et al.] // Proceedings of the 31st International Conference on Neural Information Processing Systems. — Red Hook, NY, USA : Curran Associates Inc., 2017. — P. 6000—6010.

71. A Neural Probabilistic Language Model / Y. Bengio [et al.] // Journal of Machine Learning Research. — 2003. — Vol. 3. — P. 1137—1155.

72. *Schwenk, H.* Continuous space language models / H. Schwenk // Computer Speech and Language. — 2007. — Vol. 21, no. 3. — P. 492—518.

73. Distributed Representations of Words and Phrases and their Compositionality / T. Mikolov [et al.] // Proceedings of an Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems. — Red Hook : Curran Associates, 2013. — P. 3111—3119.

74. *Pennington, J.* Glove: Global Vectors for Word Representation / J. Pennington, R. Socher, C. D. Manning // Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing. — Stroudsburg : ACL, 2014. — P. 1532—1543.

75. *Bansal, M.* Tailoring Continuous Word Representations for Dependency Parsing / M. Bansal, K. Gimpel, K. Livescu // Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, Volume 2: Short Papers. — Stroudsburg : ACL, 2014. — P. 809—815.

76. *Chen, D.* A Fast and Accurate Dependency Parser using Neural Networks / D. Chen, C. D. Manning // Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing. — Stroudsburg : ACL, 2014. — P. 740—750.

77. *Kalchbrenner, N.* A Convolutional Neural Network for Modelling Sentences / N. Kalchbrenner, E. Grefenstette, P. Blunsom // Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). — Stroudsburg : ACL, 2014. — P. 655—665.

78. *Gu, X.* Cascaded Convolutional Neural Networks for Aspect-Based Opinion Summary / X. Gu, Y. Gu, H. Wu // Neural Processing Letters. — 2017. — Vol. 46, no. 2. — P. 581—594.

79. Convolutional Sequence to Sequence Learning / J. Gehring [et al.] // Proceedings of the 34th International Conference on Machine Learning. Vol. 70. — PMLR, 2017. — P. 1243—1252.

80. *Krizhevsky, A.* ImageNet Classification with Deep Convolutional Neural Networks / A. Krizhevsky, I. Sutskever, G. E. Hinton // Proceedings of 26th Annual Conference on Neural Information Processing Systems. — Red Hook : Curran Associates, 2012. — P. 1106—1114.

81. Maxout Networks / I. J. Goodfellow [et al.] // Proceedings of the 30th International Conference on Machine Learning. Vol. 28. — JMLR.org, 2013. — P. 1319—1327.

82. Deep Residual Learning for Image Recognition / K. He [et al.] // Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition. — Piscataway : IEEE, 2016. — P. 770—778.

83. *Bjerva, J.* Semantic Tagging with Deep Residual Networks / J. Bjerva, B. Plank, J. Bos // Proceedings of the 26th International Conference on Computational Linguistics. — Stroudsburg : ACL, 2016. — P. 3531—3541.

84. *Östling, R.* Morphological reinflection with convolutional neural networks / R. Östling // Proceedings of the 14th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology. — Stroudsburg : ACL, 2016. — P. 23—26.

85. *Graves, A.* Hybrid speech recognition with Deep Bidirectional LSTM / A. Graves, N. Jaitly, A. Mohamed // Proceedings of the 2013 IEEE Workshop on Automatic Speech Recognition and Understanding. — Piscataway : IEEE, 2013. — P. 273—278.

86. *Pascanu, R.* On the difficulty of training recurrent neural networks / R. Pascanu, T. Mikolov, Y. Bengio // Proceedings of the 30th International Conference on Machine Learning. Vol. 28. — JMLR.org, 2013. — P. 1310—1318.

87. *Hochreiter, S.* Long Short-Term Memory / S. Hochreiter, J. Schmidhuber // Neural Computation. — 1997. — Vol. 9, no. 8. — P. 1735—1780.

88. Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling / J. Chung [et al.] // CoRR. — 2014. — Vol. abs/1412.3555. — arXiv: 1412.3555. — URL: http://arxiv.org/abs/1412.3555.

89. *Sutskever, I.* Sequence to Sequence Learning with Neural Networks / I. Sutskever, O. Vinyals, Q. V. Le // Proceedings of the Conference on Neural Information Processing Systems 2014. — Red Hook : Curran Associates, 2014. — P. 3104—3112.

90. Show, Attend and Tell: Neural Image Caption Generation with Visual Attention / K. Xu [et al.] // Proceedings of the 32nd International Conference on Machine Learning. Vol. 37. — JMLR.org, 2015. — P. 2048—2057. — URL: http://proceedings.mlr.press/v37/xuc15.html.

91. Neural Architectures for Named Entity Recognition / G. Lample [et al.] // Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. — Stroudsburg : ACL, 2016. — P. 260—270.

92. Recurrent Neural Network Grammars / C. Dyer [et al.] // Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. — Stroudsburg : ACL, 2016. — P. 199—209. — URL: https://www.aclweb.org/anthology/N16-1024.

93. *İrsoy, O.* Opinion Mining with Deep Recurrent Neural Networks / O. İrsoy, C. Cardie // Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing. — Stroudsburg : ACL, 2014. — P. 720—728.

94. Target-Dependent Sentiment Classification with Long Short Term Memory / D. Tang [et al.] // CoRR. — 2015. — Vol. abs/1512.01100. — arXiv: 1512.01100. — URL: http://arxiv.org/abs/1512.01100.

95. *Katiyar, A.* Investigating LSTMs for Joint Extraction of Opinion Entities and Relations / A. Katiyar, C. Cardie // Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). — Stroudsburg : ACL, 2016. — P. 919—929. — URL: https://www.aclweb.org/anthology/P16-1087.

96. *Marrese-Taylor, E.* Mining fine-grained opinions on closed captions of YouTube videos with an attention-RNN / E. Marrese-Taylor, J. Balazs, Y. Matsuo // Proceedings of the 8th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis. — Stroudsburg : ACL, 2017. — P. 102—111. — URL: https://www.aclweb.org/anthology/W17-5213.

97. *Tang, D.* Aspect Level Sentiment Classification with Deep Memory Network / D. Tang, B. Qin, T. Liu // Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing. — Stroudsburg : ACL, 2016. — P. 214—224. — URL: https://www.aclweb.org/anthology/D16-1021.

98. *Xu, Z.* Graph Enhanced Memory Networks for Sentiment Analysis / Z. Xu, R. Vial, K. Kersting // Proceedings of the European Conference on Machine Learning and Knowledge Discovery in Databases. Vol. 10534. — Cham : Springer, 2017. — P. 374—389.

99. Recursive Neural Conditional Random Fields for Aspect-based Sentiment Analysis / W. Wang [et al.] // Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing. — Stroudsburg : ACL, 2016. — P. 616—626.

100. Large-scale Opinion Relation Extraction with Distantly Supervised Neural Network / C. Sun [et al.] // Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers. — Stroudsburg : ACL, 2017. — P. 1033—1043. — URL: https://www.aclweb.org/anthology/E17-1097.

101. *Miwa, M.* End-to-End Relation Extraction using LSTMs on Sequences and Tree Structures / M. Miwa, M. Bansal // Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). — Stroudsburg : Association for Computational Linguistics, 2016. — C. 1105—1116.

102. Semantic Role Labeling with Pretrained Language Models for Known and Unknown Predicates / D. Larionov [et al.] // Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019). — Varna, Bulgaria : INCOMA. — P. 619—628.

103. Towards an integrated pipeline for aspect-based sentiment analysis in various domains / O. De Clercq [et al.] // Proceedings of the 8th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis. — Stroudsburg : ACL, 2017. — P. 136—142.

104. *Katiyar, A.* Going out on a limb: Joint Extraction of Entity Mentions and Relations without Dependency Trees / A. Katiyar, C. Cardie // Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). — Stroudsburg : ACL, 2017. — P. 917—928.

105. Joint Aspect and Polarity Classification for Aspect-based Sentiment Analysis with End-to-End Neural Networks / M. Schmitt [et al.] // Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. — Stroudsburg : ACL, 2018. — P. 1109—1114.

106. *Yamada, H.* Statistical Dependency Analysis with Support Vector Machines / H. Yamada, Y. Matsumoto // Proceedings of the Eighth International Conference on Parsing Technologies. — 2003. — P. 195—206.

107. *Zhang, Y.* A Tale of Two Parsers: Investigating and Combining Graph-based and Transition-based Dependency Parsing / Y. Zhang, S. Clark // Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing. — Stroudsburg : ACL, 2008. — P. 562—571.

108. *Rumelhart, D. E.* Learning representations by back-propagating errors / D. E. Rumelhart, G. E. Hinton, R. J. Williams // Nature. — 2012. — Vol. 323, no. 6088. — P. 533—536.

109. *Kingma, D. P.* Adam: A Method for Stochastic Optimization / D. P. Kingma, J. Ba // Proceedings of the 3rd International Conference on Learning Representations. — arXiv.org, 2017. — URL: http://arxiv.org/abs/1412.6980.

110. *Williams, R. J.* A Learning Algorithm for Continually Running Fully Recurrent Neural Networks / R. J. Williams, D. Zipser // Neural Computation. — 1989. — Vol. 1, no. 2. — P. 270—280. — URL: https://doi.org/10.1162/neco.1989.1.2.270.

111. Professor Forcing: A New Algorithm for Training Recurrent Networks / G. Anirudh [et al.] // Proceedings of an Advances in Neural Information Processing Systems 29. — Red Hook : Curran Associates, 2016. — P. 4601—4609.

112. *Goldberg, Y.* A Dynamic Oracle for Arc-Eager Dependency Parsing / Y. Goldberg, J. Nivre // Proceedings of COLING 2012. — Mumbai : The COLING 2012 Organizing Committee, 2012. — P. 959—976.

113. *Goldberg, Y.* Training Deterministic Parsers with Non-Deterministic Oracles / Y. Goldberg, J. Nivre // Transactions of the Association for Computational Linguistics. — 2013. — Vol. 1. — P. 403—414.

114. *Johansson, R.* Syntactic and Semantic Structure for Opinion Expression Detection / R. Johansson, A. Moschitti // Proceedings of the 14th Conference on Computational Natural Language Learning. — Stroudsburg : ACL, 2010. — P. 67—76. — URL: https://www.aclweb.org/anthology/W10-2910/.

115. *Cohen, J.* A Coefficient of Agreement for Nominal Scales / J. Cohen // Educational and Psychological Measurement. — 1960. — Vol. 20, no. 1. — P. 37—46.

116. Dropout: A Simple Way to Prevent Neural Networks from Overfitting / N. Srivastava [et al.] // Journal of Machine Learning Research. — 2014. — Vol. 15, no. 56. — P. 1929—1958. — URL: http://jmlr.org/papers/v15/srivastava14a.html.

117. *Gal, Y.* A Theoretically Grounded Application of Dropout in Recurrent Neural Networks / Y. Gal, Z. Ghahramani // Proceedings of the 30th International Conference on Neural Information Processing Systems. — Red Hook : Curran Associates, 2016. — P. 1027—1035.

118. ГОСТ Р ИСО/МЭК 12207-2010. Информационная технология. Системная и программная инженерия. Процессы жизненного цикла программных средств. — М. : Стандартинформ, 2011. — 5 с.

119. *Schach, S. R.* Object-oriented and classical software engineering. / S. R. Schach. — N.Y. : McGraw-Hill Education, 2011. — 688 p.

120. Is It a Bug or an Enhancement? A Text-Based Approach to Classify Change Requests / G. Antoniol [et al.] // Proceedings of the 2008 Conference of the Center for Advanced Studies on Collaborative Research: Meeting of Minds. — New York, NY, USA : ACM, 2008. — P. 304—318.

121. *Pagano, D.* User feedback in the appstore: An empirical study / D. Pagano, W. Maalej // 2013 IEEE 21st International Requirements Engineering Conference (RE). — Los Alamitos, CA, USA : IEEE Computer Society, 2013. — P. 125—134.

122. *Iacob, C.* Retrieving and Analyzing Mobile Apps Feature Requests from Online Reviews / C. Iacob, R. Harrison // Proceedings of the 10th Working Conference on Mining Software Repositories. — San Francisco, CA, USA : IEEE Press, 2013. — C. 41—44.

123. SCARE — The Sentiment Corpus of App Reviews with Fine-grained Annotations in German / M. Sänger [et al.] // Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16). — Portorož, Slovenia : ELRA, 2016. — P. 1114—1121.

124. *Artetxe, M.* Massively Multilingual Sentence Embeddings for Zero-Shot Cross-Lingual Transfer and Beyond / M. Artetxe, H. Schwenk // Transactions of the Association for Computational Linguistics. — 2019. — Vol. 7. — P. 597—610. — URL: https://transacl.org/ojs/index.php/tacl/article/view/1742.

125. UMAP: Uniform Manifold Approximation and Projection / L. McInnes [et al.] // Journal of Open Source Software. — 2018. — Vol. 3, no. 29. — P. 861.

126. *Zimek, A.* A survey on unsupervised outlier detection in high-dimensional numerical data / A. Zimek, E. Schubert, H. Kriegel // Statistical Analysis and Data Mining. — 2012. — Vol. 5, no. 5. — P. 363—387.

127. *Campello, R. J. G. B.* Density-Based Clustering Based on Hierarchical Density Estimates / R. J. G. B. Campello, D. Moulavi, J. Sander // Proceedings of Advances in Knowledge Discovery and Data Mining, 17th Pacific-Asia Conference Part II. Vol. 7819 / ed. by J. Pei [et al.]. — Berlin : Springer, 2013. — P. 160—172.

Публикации автора по теме диссертации

19. *Ехлаков, Ю. П.* Модель извлечения пользовательских мнений о потребительских свойствах товара на основе рекуррентной нейронной сети / Ю. П. Ехлаков, Е. И. Грибков // Бизнес-информатика. — 2018. — Т. 46, № 4. — С. 7—16.

20. *Грибков, Е. И.* Нейросетевая модель для обработки запросов пользователей на этапе эксплуатации и сопровождения программного продукта / Е. И. Грибков, Ю. П. Ехлаков // Бизнес-информатика. — 2020. — Т. 14, № 1. — С. 7—18.

21. *Грибков, Е. И.* Нейросетевая модель на основе системы переходов для извлечения составных объектов и их атрибутов из текстов на естественном языке / Е. И. Грибков, Ю. П. Ехлаков // Доклады ТУСУР. — 2020. — Т. 23, № 1. — С. 47—52.

22. *Грибков, Е. И.* Нейросетевая модель на основе системы переходов для извлечения и анализа тональности пользовательских мнений / Е. И. Грибков, Ю. П. Ехлаков // Искусственный интеллект и принятие решений. — 2020. — № 1. — С. 99—110.

23. *Грибков, Е. И.* Набор данных и модель глубокого обучения для анализа текстов отзывов пользователей / Е. И. Грибков, Ю. П. Ехлаков // Наука. Технологии. Инновации. Сборник научных трудов. В 9-ти частях. — Новосибирск : НГТУ, 2018. — С. 180—184.

24. *Грибков, Е. И.* Модель обработки запросов пользователей на этапе эксплуатации и сопровождения программного продукта / Е. И. Грибков, Ю. П. Ехлаков // Электронные средства и системы управления. Материалы докладов Международной научно-практической конференции. — Томск : В-Спектр, 2019. — С. 141—143.

25. *Грибков, Е. И.* Модель извлечения структурированных объектов и их атрибутов из текстов на естественном языке / Е. И. Грибков // Сборник избранных статей научной сессии ТУСУРа (Томск, 22–24 мая 2019 г.): в 2 ч. — Томск : В-Спектр, 2019. — С. 54—56.

26. *Грибков, Е. И.* Модель на основе системы переходов для извлечения составных объектов из текстов / Е. И. Грибков, Ю. П. Ехлаков // Сборник избранных статей научной сессии ТУСУР, Томск, 13–30 мая 2020. — Томск : В-Спектр, 2020. — С. 52—55.

27. «Quiddi Semantics» / Е. И. Грибков [и др.] // Свидетельство о государственной регистрации программы для ЭВМ №2019612276 от 14.02.2019.

28. «Quiddi Support Analyst» / Е. И. Грибков [и др.] // Свидетельство о государственной регистрации программы для ЭВМ №2020614799 от 24.04.2020.

Список рисунков

Список таблиц

РОССИЙСКАЯ ФЕДЕРАЦИЯ

СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2019612276

Quiddi Semantics

Правообладатель: ***Общество с ограниченной ответственностью «ТомскСофт» (RU)***

Авторы: ***Безходарнов Илья Владимирович (RU), Грибков Егор Игоревич (RU), Трошин Максим Вадимович (RU), Тушминцев Антон Александрович (RU)***

Заявка № **2019611004**
Дата поступления **05 февраля 2019 г.**
Дата государственной регистрации
в Реестре программ для ЭВМ ***14 февраля 2019 г.***

Руководитель Федеральной службы
по интеллектуальной собственности

Г.П. Ивлиев

РОССИЙСКАЯ ФЕДЕРАЦИЯ

СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2020614799

Quiddi Support Analyst

Правообладатель: ***Общество с ограниченной ответственностью «ТомскСофт» (RU)***

Авторы: ***Безходарнов Илья Владимирович (RU), Грибков Егор Игоревич (RU), Трошин Максим Вадимович (RU), Тушминцев Антон Александрович (RU)***

Заявка № **2020613825**
Дата поступления **03 апреля 2020 г.**
Дата государственной регистрации
в Реестре программ для ЭВМ ***24 апреля 2020 г.***

Руководитель Федеральной службы
по интеллектуальной собственности

Г.П. Ивлиев

Томсксофт
Программное обеспечение

ООО «ТомскСофт»

Россия, 634034, г. Томск,
ул. Нахимова д.8
тел. 8 (3822) 90-22-53
tomsksoft@tomsksoft.ru
www.tomsksoft.ru

АКТ
о внедрении результатов диссертационной работы
на соискание ученой степени кандидата технических наук
Грибкова Егора Игоревича

Настоящий акт свидетельствует о том, что результаты диссертационной работы Грибкова Е.И. «Нейросетевые модели на основе системы переходов для извлечения структурированной информации о продуктах из текстов пользователей» были использованы компанией ООО «ТомскСофт» при разработке алгоритмов анализа текста для программных систем «Quiddi Semantics» и «Quiddi Support Analyst».

ПС «Quiddi Semantics» является частью сервиса для агрегации и анализа отзывов пользователей о товарах «Quiddi.ru» и используется для анализа текстов отзывов покупателей интернет-магазина «AliExpress». На данный момент сервисом проанализированы тексты отзывов о 5 миллионах продуктов на русском языке из интернет-магазина «AliExpress». Результаты обработки отзывов предоставляются пользователям сервиса «Quiddi.ru» в виде услуги.

ПС «Quiddi Support Analyst» находится на стадии рабочего прототипа, который в дальнейшем будет использован в качестве основы для разработки программного продукта для автоматизации работы службы технической поддержки.

Председатель комиссии Безходарнов И.В.
Члены комиссии Иванов Е.О.
Романов А.С.

ОБЩЕСТВО С ОГРАНИЧЕННОЙ ОТВЕТСТВЕННОСТЬЮ
ТомскСофт
ИНН 7017049173
ТОМСК
РОССИЯ

Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования
«ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ»

УТВЕРЖДАЮ
Проректор по научной работе инновациям
канд. техн. наук, доцент
Лощилов А.Г.

«10» 09 2020

СПРАВКА

об использовании результатов диссертационной работы на соискание ученой степени кандидата технических наук Грибкова Егора Игоревича

Комиссия в составе председателя Гриценко Ю.Б., начальника инновационного управления, членов: Журавлевой Н.Л., начальника отдела организации и планирования НИОКР, Ехлакова Ю.П., профессора кафедры автоматизации обработки информации, научного руководителя проекта, подтверждают, что результаты диссертационной работы Грибкова Е.И. «Нейросетевые модели на основе системы переходов для извлечения структурированной информации о продуктах из текстов пользователей» использованы при выполнении государственного задания ТУСУРа (проект FEWM-2020-0036 «Методическое и инструментальное обеспечение принятия решений в задачах управления социально-экономическими системами и процессами в гетерогенной информационной среде»).

При личном участии Грибкова Е.И. были получены следующие результаты, включенные в состав отчетных материалов:

– модель для извлечения составных объектов и их атрибутов из текстов на естественном языке;

– нейросетевая модель для извлечения и анализа пользовательских мнений из текстов отзывов о потребительских свойствах товаров;

– нейросетевая модель для обработки запросов пользователей на этапе эксплуатации программного продукта.

Председатель комиссии Ю.Б. Гриценко

Члены комиссии Н.Л. Журавлева

Ю.П. Ехлаков

Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования
«ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ»

УТВЕРЖДАЮ
Проректор по учебной работе
канд. техн. наук, доцент
Сенченко П.В.
«16» 09 2020

СПРАВКА

об использовании в учебном процессе результатов диссертационной работы на соискание ученой степени кандидата технических наук Грибкова Егора Игоревича

Комиссия в составе председателя Салминой Н.Ю., декана факультета систем управления, членов: Сидорова А.А., заведующего кафедрой автоматизации обработки информации (АОИ), Потаховой И.В., заместителя заведующего кафедрой АОИ по учебно-методической работе, подтверждают, что результаты диссертационной работы Грибкова Е.И. «Нейросетевые модели на основе системы переходов для извлечения структурированной информации о продуктах из текстов пользователей» используются в учебном процессе кафедры АОИ при организации занятий по дисциплинам «Интеллектуальные вычислительные системы», «Анализ больших данных», «Нейронные сети и их применение» при подготовке магистров по направлению 09.04.04 «Программная инженерия».

Изучение нейросетевых моделей, предложенных Грибковым Е.И., позволяет студентам ТУСУРа приобрести компетенции в области современных методов обработки естественного языка с помощью моделей на основе нейронных сетей, в том числе при извлечении структурированной информации из текстов пользователей.

Председатель комиссии Н.Ю. Салмина

Члены комиссии А.А. Сидоров

И.В. Потахова