МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
«УЛЬЯНОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

На правах рукописи

ДУДАРИН ПАВЕЛ ВЛАДИМИРОВИЧ

ИССЛЕДОВАНИЕ И РАЗРАБОТКА МОДЕЛЕЙ И МЕТОДОВ НЕЧЕТКОЙ КЛАСТЕРИЗАЦИИ КОРОТКИХ ТЕКСТОВ

Специальность: 05.13.01 - Системный анализ, управление
и обработка информации (информационные технологии и промышленность)

ДИССЕРТАЦИЯ
на соискание учёной степени
кандидата технических наук

Научный руководитель:
доктор технических наук, профессор
Ярушкина Надежда Глебовна

Ульяновск – 2021

ВВЕДЕНИЕ

Кластерный анализ является одним из важнейших разделов системного анализа данных и применяется в различных проблемных областях - технических, естественнонаучных, социальных. Кластеризация является примером задачи обучения без учителя и сводится к разбиению исходного множества объектов на подмножества классов таким образом, чтобы элементы одного класса были максимально схожи между собой, а элементы различных классов - отличались. Исследованиям в данной области посвящены работы известных зарубежных и российских ученых: Basu S., Hinton G., Manning Ch.D., Hastie T., Kaufman L., Picard R.W., Воронцова К.В., Хорошевского В.Ф., Ярушкиной Н.Г. и др.

Традиционные методы кластерного анализа работают с объектами, заданными в виде векторов признаков. При работе с текстами первым шагом алгоритма кластеризации является определение пространства признаков и построение в нем векторов имеющихся текстов. Как правило, получаемые векторы имеют большую размерность и при работе с ними традиционные методы кластерного анализа не обеспечивают достаточную эффективность. В случае работы с короткими текстами размерность векторов не уменьшается, а лишь добавляется свойство разреженности к векторам признаков, что создает дополнительные трудности при их обработке методами кластерного анализа. Под короткими текстами в данном исследовании подразумеваются тексты, состоящие из одного или нескольких предложений с общим числом слов в диапазоне от 5 до 100. Кроме того дополнительными факторами осложняющими решение задачи кластеризации для коротких текстов являются: синонимия, омонимия, более частое, по сравнению с обычными текстами, использование аббревиатур, сленговых выражений и неологизмов и самое главное - частичное или полное отсутствие контекста у коротких текстов.

Высокая размерность получаемых пространств признаков в случае работы с текстами объективна, так как тексты это сложные многомерные и многоплановые структуры, потенциально содержащие различные смыслы, эмоциональные оттенки, авторские характерные черты, стиль изложения и многое другое. При большом разнообразии возможных характеристик подход четкой кластеризации, в котором каждому объекту сопоставляется один и только один кластер, является не достаточно эффективным. Эксперту, проводящему процедуру кластеризации, в ходе анализа результатов важно знать и понимать альтернативные варианты соотнесения объекта с кластером. Поэтому, в случае работы с текстами, наиболее предпочтительными являются методы нечеткой кластеризации.

Кластеризация текстов допускает значительное число возможных принципов для разбиения на классы: тематика, автор, стиль, эмоциональная окраска, правовой статус и комбинация различных факторов. Методы не позволяющие учесть интенцию эксперта оказываются в общем случае не эффективными для решения описанной задачи. Альтернативным является подход, при котором эксперт включается в процесс кластеризации и на различных ее этапах задает ограничения на основе промежуточных результатов, которые учитываются на дальнейших стадиях кластеризации. Такие методы классифицируются как методы интерактивной кластеризации с использованием обратной связи от эксперта. Интерактивные методы обеспечивают сокращение суммарных затрат времени эксперта на обработку результатов кластеризации и позволяют повысить точность кластеризации, за счет выявления скрытого знания эксперта на ранних этапах кластеризации. Учет дополнительной информации позволяет алгоритму выбрать правильное направление хода процесса разбиения на кластеры.

Актуальность проблемы

Стремительный рост массивов информации, состоящих из наборов коротких текстовых фрагментов, способствует интенсификации исследований в

области развития методов обработки текстов с применением машинного обучения. Проблеме ежегодно посвящается значительно число исследований. Большая часть проводимых исследований относится к текстам на английском языке. Исследований в области русского языка значительно меньше, что объясняется не только меньшим числом исследователей занимающихся вопросами русского языка, но и объективно большей сложностью русского языка для автоматизированной обработки. Недостаточная разработанность стандартных средств кластеризации для коротких текстов и низкая эффективность существующих методов на текстах на русском языке затрудняет их использование в российских автоматизированных системах поддержки принятия решений и управления. Это подтверждается отсутствием стандартных средств кластеризации для коротких текстов в ведущих NLP(Natural Language Processing, Обработка Естественного Языка) пакетах (например, NLTK).

В данной работе рассматривается пример системы, в которой происходит генерация большого количество коротких текстов - система стратегического планирования Российской Федерации. В ней участники формируют документы стратегического планирования, в рамках которых определяются ключевые показатели эффективности. Формулировки ключевых показателей эффективности образуют набор данных, состоящий из коротких текстов. В рамках данной системы остро стоит задача формирования и актуализации классификатора основанного на данном наборе. Эта задача может быть решена с помощью кластеризации.

На основании вышеизложенного можно сформулировать вывод о том, что исследования в области интерактивной нечеткой кластеризации коротких текстов на русском языке являются важной и актуальной задачей.

Объект исследования

Объектом исследования в диссертационной работе является кластеризация наборов данных, состоящих из коротких текстов на русском языке и экспертная информация, поступающая в ходе интерактивной обработки текстов.

Предмет исследования

Предметом исследования являются модели и методы нечеткой кластеризации коротких текстов и обработки экспертной информации.

Цель работы

Повышение эффективности нечеткой кластеризации коротких текстов путем разработки модели, метода и алгоритма в системе поддержки принятия решений для кластеризации коротких текстов на русском языке с учетом экспертной информации. Эффективность определяется точностью кластеризации и сокращением времени и трудоемкости работы выполняемой экспертом при использовании предложенного решения.

Для достижения поставленной цели необходимо решить следующие задачи:

- провести исследование моделей и методов машинного обучения для обработки текстов для выявления новых подходов к повышению эффективности четкой и нечеткой кластеризации коротких текстов;
- разработать метод расширения словаря языковой модели на базе нейронной сети;
- разработать метод для обработки экспертной информации в ходе нечеткой интерактивной кластеризации коротких текстов;
- сформулировать перечень этапов программы проведения испытаний метода нечеткой интерактивной кластеризации коротких текстов;

- составить алгоритм автоматизации работ по нечеткой интерактивной кластеризации коротких текстов в системе поддержки принятия решений;
- провести апробацию разработанных модели, методов и алгоритма нечеткой интерактивной кластеризации коротких текстов в качестве элементов функционирующей системы поддержки принятия решений.

Методы исследования

При решении задач исследования были применены методы теории вероятностей, математической статистики, методы машинного обучения, кластерный анализ, теория нечетких множеств, численные методы. При разработке программного модуля были использованы методы объектно-ориентированного программирования.

Область исследования

Область исследования соответствует паспорту специальности 05.13.01. – «Системный анализ, управление и обработка информации (технические науки)», а именно:
п. 4 – разработка методов и алгоритмов решения задач системного анализа, оптимизации, управления, принятия решений и обработки информации;
п. 13 - методы получения, анализа и обработки экспертной информации.

Научной новизной обладают:

- предложенная архитектура искусственной нейронной сети, отличающаяся от известных тем, что позволяет решать задачу кластеризации на базе скрытого пространства признаков языковой модели;
- предложенный метод обработки текстов для расширения словаря языковой модели на базе нейронной сети с использованием нечеткого иерархического классификатора, отличающийся от известных тем, что позволяет учитывать семантическую близость слов;

– предложенный метод обработки обратной связи от эксперта, отличающийся от известных тем, что позволяет корректировать весовые коэффициенты нейронной сети и проводить интерактивную кластеризацию наборов коротких текстов;
– разработанный алгоритм, автоматизирующий применение предложенных модели и методов для выполнения нечеткой интерактивной кластеризации наборов коротких текстов, интегрированный в систему поддержки принятия решений (СППР).

Достоверность результатов работы

Достоверность полученных результатов обеспечена математически строгим выполнением расчетов, подтверждена вычислительными экспериментами и результатами практического использования.

Теоретическая значимость диссертационной работы

Теоретическая значимость диссертационной работы заключается в разработке новых моделей и методов с использованием нейронных сетей и языковых моделей для решения задачи нечеткой кластеризации наборов данных состоящих из коротких текстов.

Практическая значимость диссертационной работы

Практическая значимость диссертационной работы заключается в разработке программного модуля системы поддержки принятия решений на языке Python, позволяющего осуществлять интерактивную нечеткую кластеризацию наборов данных состоящих из коротких текстов и применение его в задаче системного анализа набора коротких текстов в рамках НИР в интересах Министерства экономического развития РФ для Системы стратегического планирования РФ.

Основные научные положения, выносимые на защиту:

1. Предложенная архитектура искусственной нейронной сети, позволяет эффективно решать задачу кластеризации на базе пространства признаков языковой модели русского языка;
2. Предложенный метод обработки текстов для расширения словаря языковой модели на базе нейронной сети с использованием нечеткого иерархического классификатора повышает точность кластеризации;
3. Предложенный метод учета обратной связи от эксперта, используемый для корректировки весовых коэффициентов нейронной сети позволяет проводить интерактивную кластеризацию наборов коротких текстов;
4. Разработанный алгоритм на основе предложенных моделей и методов реализован в системе поддержки принятия решений и автоматизирует применение предложенных моделей и методов для выполнения нечеткой интерактивной кластеризации наборов коротких текстов.

Реализация и внедрение результатов работы

Основные теоретические и практические результаты диссертационной работы использованы в рамках фундаментальных и прикладных научных исследований Министерства экономического развития РФ по темам: “Разработка рекомендаций по совершенствованию информационного обеспечения участников стратегического планирования в части осуществления мониторинга и контроля реализации документов стратегического планирования с использованием Федеральной информационной системы стратегического планирования (ФИС СП)” и “Разработка методического обеспечения интеллектуальной системы проверки уведомления об утверждении (одобрении) документа стратегического планирования или внесении в него изменений при ведении федерального государственного реестра документов стратегического планирования Федеральной информационной системы стратегического планирования (ФИС СП)”. Результаты НИР внедрены в системе ГАС “Управление”.

Архитектура искусственной нейронной сети и алгоритм нечеткой кластеризации коротких текстов, методы расширения словаря языковой модели и корректировки весов нейронной сети для учета обратной связи эксперта в интерактивной кластеризации, а также программная реализация метода нечеткой интерактивной кластеризации на языке Python внедрены в системе Планета.Аналитика 4.0 (включена в реестр отечественного ПО) компании ООО “ИБС “Экспертиза”.

Апробация работы

Основные положения и результаты диссертационной работы доложены и обсуждены на конференциях и конгрессах:

- Всероссийская научно-практическая конференция “Нечеткие системы и мягкие вычисления” (Санкт-Петербург 2017);
- Международная конференция “Интеллектуальные информационные технологии в технике и на производстве” IITI (Варна 2017, Сочи 2018, Острава 2019);
- Всероссийской научной конференции «Нечеткая логика и мягкие вычисления в промышленности» (Ульяновск: 2017, 2018, 2019);
- “Национальная Конференция по Искусственному Интеллекту” (Москва 2018, Ульяновск 2019);
- Международная конференция “World Conference on Soft Computing” (Баку 2018);
- Международная конференция “Mexican International Conference on Artificial Intelligence” (Гвадалахара 2018);
- Международная конференция “European Society for Fuzzy Logic and Technology” (Прага 2019).
- Международная конференция по компьютерной лингвистике и интеллектуальным технологиям “Диалог” (Москва, 2019).
- I Национальный конгресс по когнитивным исследованиям, искусственному интеллекту и нейроинформатике (Москва, 2020).

Публикации по теме диссертации

Основные результаты диссертационного исследования опубликованы в 19 печатных работах, в том числе 6 статей в российских рецензируемых научных журналах из Перечня, рекомендованного ВАК РФ, 7 публикаций в изданиях индексируемых в Scopus и Web of Science, 6 в материалах научных конференций.

Сведения о личном вкладе автора

Постановка задач исследования осуществлялась совместно с научным руководителем. Все основные теоретические и практические исследования диссертационной работы проведены лично автором. Подготовка к публикации некоторых результатов проводилась совместно с соавторами, вклад соискателя был определяющим.

Структура и объем работы

Диссертация изложена на 136 страницах машинописного текста, содержит 46 рисунков, 9 таблиц, состоит из введения, четырех глав, заключения, списка использованной литературы из 128 наименований на 15 страницах и 3 приложений на 6 страницах.

ГЛАВА 1. Сравнительный анализ моделей и методов нечеткой кластеризации коротких текстов

Со стремительным развитием технологий web 2.0, все больше и больше коротких текстов генерируется различными видами веб-сайтов. Facebook (посты и статусы с ограничением в 142 символа, Twitter – с ограничением в 140 символов, Windows Love Messenger с ограничением в 128 символов, Yahoo! Answers со средней длиной поста в 500 символов, ВКонтакте, Instagramm и MicroBlog – лишь несколько примеров таких веб-сайтов. Объемы сообщений на перечисленных web-ресурсах исчисляется миллионами, при этом регулярно возникают новые тематики, меняются тренды, изменяются смысл слов и целых фраз. Также традиционно к наборам данных коротких текстов относятся наборы заголовков новостей, например набор данных новостного агентства Reuters содержащий около 800 тысяч заголовков статей, состоящий из более чем 100 тематик. Бытовым примером коротких текстов является набор данных по объявлениям в сети Internet компании “Avito”, в нем насчитывается около 500 000 объявлений по 4-ем крупным категориям и ряду подкатегорий. По данному набору компания проводит различные соревнования, в том числе по точности классификации, которая на момент написания данной работы достигала 88%. На Рис. 1.1 представлены результаты одного из проведенных соревнований по классификации объявлений.

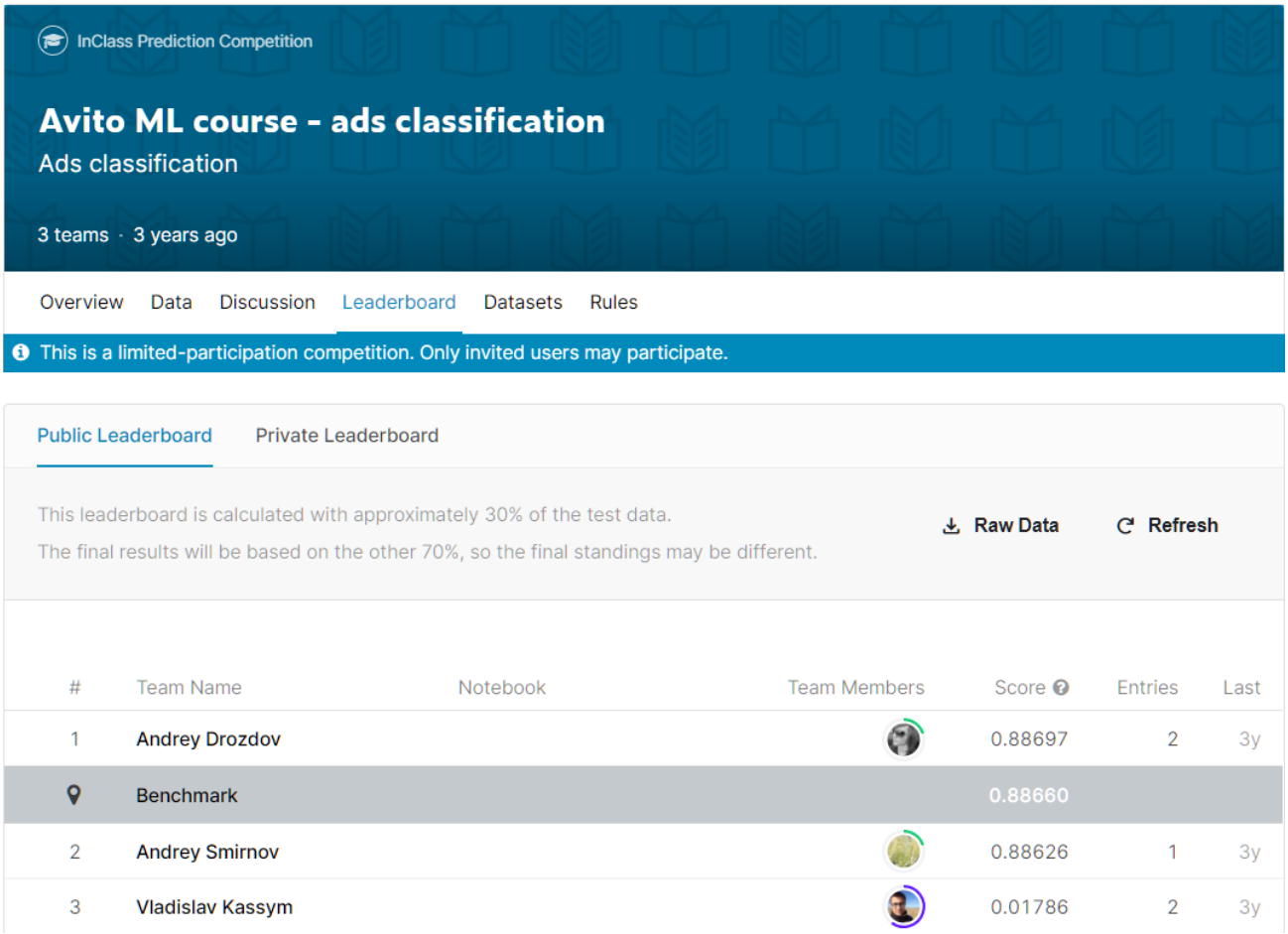


Рис. 1.1. Результаты соревнований по классификации коротких текстов

В последние годы (с 2016 года) в Российской Федерации появилась Система Стратегического Планирования, содержащая набор ключевых показателей эффективности (КПЭ, KPI) системы стратегического планирования Российской Федерации закрепленной федеральным законом № 172-ФЗ от 28.07.2014 г. (изм. от 22.05.2019 N 641,от 18.11.2019 N 1468) программное обеспечение для которой разработано в рамках НИР в интересах Министерства экономического развития РФ по теме: Разработка рекомендаций по совершенствованию информационного обеспечения участников стратегического планирования в части осуществления мониторинга и контроля реализации документов стратегического планирования с использованием федеральной информационной системы стратегического планирования (ФИС СП). Общее количество документов около

600 000. По 11 нормативно установленным категориям предлагалось построить классификатор в рамках каждой категории, на базе результатов кластеризации.

Таким образом, в данной работе под короткими текстами подразумеваются тексты произвольной тематики, состоящие из одного или нескольких предложений и содержащие от 5 до 100 слов.

1.1. Обзор современных методов четкой и нечеткой кластеризации

Методы кластеризации относятся к группе методов машинного обучения без учителя. “Четкая” кластеризация заключается в разбиении исследуемого набора данных o = {o1, o2, o3, …} на группы классов c = {c1, c2, …} — таким образом, чтобы элементы одного класса существенно отличались друг от друга по заданному набору параметров p = {p1, p2, p3, ….} — от элементов других классов, и были максимально схожи с элементами своего класса [104,113,70,57].

Нечеткая кластеризация (также называемая мягкой кластеризацией) - это форма кластеризации, в которой каждая точка данных может принадлежать более чем одному кластеру с определенной мерой принадлежности [41,102,107,24,22].

Классические методы кластеризации успешно применяются на практике [118, 119, 114, 72, 75] и показывают высокие результаты [128,127, 37]. Помимо классического метода k-mean [10] и более продвинутых, таких как hdbscan [20,77], существуют методы на основе роевого [23,113] и генетического алгоритмов [64] оптимизации, на основе метода главных компонент [59], кластеризации графов [36,96] и других математических моделей [39,27,75,85]. Методы и метрики применяемые для оценки эффективности алгоритмов кластеризации [55,93,124,4] позволяют сравнивать работу методов одной и разной природы. На эталонных наборах данных используемых для проверки методов кластеризации, классические методы показывают 60-70 и более процентов точности [91]. Тем не менее, существует большое количество современных методов демонстрирующих намного лучшие результаты (state-of-the-art results). Большин-

ство этих методов основываются на использовании сетей с глубинным обучением (deep neural network) [73, 99, 109, 111, 46]. Такое превосходство объясняется способностью сетей обучаться на смежных предметных областях или схожих задачах (transfer learning, learning to cluster) и строить сложные нелинейные преобразования для получения пространства признаков (representation learning, embedding learning) одновременно содержащего максимум информации и “удобного” для алгоритма кластеризации (например, сильное понижении размерности входных данных) [110]. Но самым главным вкладом использования нейронных сетей в методы кластеризации является возможность построения непрерывной кластеризации (end-to-end clustering), в которой отсутствует явное разделение алгоритма на две фазы: построение пространства признаков и разбиения на группы [43, 78]. При таком подходе обучение сети подходящему представлению данных происходит одновременно с итерациями разбиения множества на кластеры или построения иерархии из них [101]. В ряде методов авторы показывают возможность дальнейшего переноса полученных знаний сети на смежные задачи, например использование сети обученной для кластеризации одного вида изображений на другой вид изображений.

1.2. Анализ современных моделей и методов обработки естественного языка

Тексты, являясь многомерными объектами, представляют особенную сложность для алгоритмов кластеризации, т.к. для них в большинстве подходов формируются пространства признаков большой размерности, с которыми не справляются традиционные методы кластеризации. Например, наиболее простыми и распространёнными способами обработки текста на естественном языке являются методы, основанные на подходе “мешка слов” [79]. Данный подход заключается в том, что все слова, используемые в исследуемом корпусе текстов, первоначально считаются равнозначными и независимыми. Это позволяет перейти от работы с естественным языком к работе с векторным пространством размерности N, где |N| = числу различных слов в корпусе, слова упорядочива-

ются в рамках словаря корпуса текстов, таким образом, каждое слово можно однозначно идентифицировать его номером в словаре. Каждому слову в таком пространстве сопоставляется кодирующий вектор (one-hot вектор) в котором все компоненты равны 0, за исключением компоненты с номером соответствующим номеру слова в словаре, эта компонента полагается равной 1.

Иногда вместо слов используют токены (произвольные части слов, в зависимости от алгоритма получения токенов) [116] или леммы (исходные формы слова), что позволяет снизить размерность исследуемого объекта. Тем не менее, даже количество лемм в корпусах текстов исчисляется тысячами и десятками тысяч.

Очевидно, что слова в тексте и в целом в естественном языке не являются независимыми, они связаны синтаксически и семантически. Учет этой связи позволяет точнее моделировать текст, использовать модели меньших размерностей и получать более качественные результаты. Так в 1998 году был представлен проект решающий задачу присвоения семантических ролей [9]. Эта форма поверхностного семантико-синтаксического анализа до сих пор активно используется и исследуется. В 2001 году была представлена модель условных случайных полей [53]. Этот класс методов разметки последовательностей “получил награду test-of-time (испытание временем) на международной конференции по машинному обучению (ICML) 2011. Слой условных случайных полей является основой современных передовых моделей, решающих проблемы разметки последовательностей взаимосвязанных объектов в таких задачах, как распознавание именованных сущностей” [71].

Широко известный метод латентного размещения Дирихле [17] впервые опубликован в 2003 году. LDA - один из наиболее широко используемых методов в машинном обучении. В классификации и кластеризации LDA является стандартным способом тематического моделирования.

Вместе с развитие методов обработки искусственного языка развивались и корпуса текстов. Например, проект OntoNotes - большой многоязычный корпус с множественными аннотациями был представлен в 2006 году [50]. Корпус

OntoNotes использовался для обучения множества задач, среди которых: синтаксический анализ на основе грамматики зависимостей и разрешение кореференции. В 2008 году Милн и Виттен показали, как Wikipedia (онлайн энциклопедия Википедия) может использоваться для обогащения наборов данных для методов машинного обучения. С тех пор Википедия служит одним из главных ресурсов для обучения моделей для обработки естественного языка.

В коллекции собираются не только тексты, но и результаты их обработки. Например, в 2016 году в проекте Universal Dependencies [84] были собраны многоязычные синтаксические деревья. К январю 2019 года Universal Dependencies насчитывал более 100 синтаксических деревьев на более чем 70 языках.

Таким образом, объем современных корпусов текстов и вычислительные мощности способствуют тому, что современные методы обработки естественного языка переходят от построения частных моделей для решения локальных задач к построению обобщенных языковых моделей для решения группы задач для корпуса текстов или целиком естественного языка. Эти подходы отражают общую тенденцию перехода к многозадачному обучению и переноса знаний, особенно широко используемому в нейронных сетях. Современные подходы к языковому моделированию рассмотрены ниже в этой же главе.

1.3. Анализ современных методов интерактивной кластеризации

Обучение без учителя возможно благодаря информации, содержащейся в самих данных, которую и призваны выявить методы кластеризации [58, 1]. Тем не менее, на практике исследователь редко не обладает никакими знаниями об исследуемом наборе данных [5], будь то экономические данные, данные собранные с датчиков, приборов или каким-либо иным образом компьютерной программой. В большинстве случаев решения практических задач участие исследователя необходимо либо для построения корректного разбиения на группы, либо принятия решения о структуре иерархии [81], либо способствует существенному повышению качества результата за счет знаний, не включенных в

пространство признаков обрабатываемых данных [49,53]. Особенно это актуально при обработке текстовой информации. Тексты, являясь многомерными объектами, представляют особенную сложность для алгоритмов кластеризации [44]. Без участия эксперта, без выявления его скрытых интенций невозможно заранее определить, какое именно разбиение ожидается в результате работы алгоритма [54,106]. Помимо очевидной группировки по тематике, тексты могут быть сгруппированы на основание того от чьего лица ведется повествование, по целевой аудитории текста, по правовому статусу текста или комбинации различных признаков. Таким образом, для получения качественного результата работы алгоритма кластеризации требуется включение эксперта в процесс кластеризации как органичной части алгоритма кластеризации. При этом, желательно, чтобы это не требовало понимания внутренних деталей работы алгоритма от эксперта, и причинно-следственная связь между действиями эксперта и результатами работы алгоритма была бы явной [123].

В современной научной литературе сложилась практика обозначения методов кластеризации, в которых используется та или иная дополнительная информация, не включенная в набор данных, методами кластеризации с частичным привлечением учителя (semi-supervised) кластеризацией с ограничениями (constrained clustering) [13, 29, 53]. При этом в подавляющем большинстве таких методов информация дана a priori и подается на вход алгоритму кластеризации совместно с набором данных в виде частично промаркированных объектов [14], заданных ограничений на пары объектов [30], ограничения на структуру иерархии кластеров, перенос знания в виде предобученной нейронной сети (transfer learning) [106], например, на задаче классификации в схожей предметной области и т.д. При этом и ограничения на объекты и метки могут быть заданы не жестко (soft labels) [82].

Однако, существуют методы предполагающие получение дополнительной информации непосредственно в процессе кластеризации их подробный обзор произведен в работе [6]. Такие методы называются методами интерактивной кластеризации. Одним из первых таких методов стал нечеткий метод [87].

В зависимости от характера взаимодействия и получаемой информации они подразделяются на: активную кластеризацию как пример активного обучения [31, 38, 125]; кластеризация с подкреплением, получаемой в виде обратной связи от среды в которой происходит кластеризация [7]; интерактивная кластеризация с обратной связью (interactive clustering under feedback, mixed-initiative clustering), подразумевающая получение обратной связи от пользователя в виде оценки результатов или указаний по корректировке алгоритма. Последние методы позволяют выявить скрытые интенции пользователя и получить по настоящему полезную кластеризацию, т.к. хорошо соответствуют тезису: “пользователь узнает правильный результат, когда увидит его” [25].

Исследователи отмечают, что к интерактивным методам зачастую ошибочно относят и методы кластеризации с интерактивными операциями: методы интерактивной визуализации результатов кластеризации, методы подбора выбора алгоритмов кластеризации и т.п. [6].

Для полноты картины следует упомянуть методы вспомогательной кластеризации (assisting clustering) [12], в которых ведущая роль отдана исследователю, именно он определяет количество кластеров и их характеристики, а алгоритм предлагает варианты их наполнения и корректировки структуры. Однако этим методы на данный момент не получили значительного распространения.

Методы интерактивной кластеризации с обратной связью можно разделить на два множества по тому на что направлена обратная связь от исследователя. В первом более многочисленном семействе методов исследователь интерактивно и итеративно может влиять на параметры алгоритма кластеризации, метрику схожести (близости), модифицировать пространство признаков [71]. Во втором множестве методов исследователь взаимодействует непосредственно с результатами кластеризации, указывая какие кластеры необходимо объединить или разъединить, какие элементы добавить или исключить из кластера, каким образом образовать новый кластер или куда отнести элементы, выпадающие из кластеризации [6, 9]. Подход, предлагаемый в данной работе, относится именно ко второму множеству, что позволяет исследователю не погружаться

в детали реализации алгоритма и использовать новые появляющиеся методы, не меняя характер своей работы.

Систематизация методов кластеризации с участием исследователя, которые относятся большому семейству методов кластеризации с привлечением учителя (semi-supervised clustering), может быть представлена следующим образом:

- Кластеризация с ограничениями (constrained clustering)
    - Интерактивная кластеризация (interactive clustering)
        - Активная кластеризация (active clustering)
        - Кластеризация с подкреплением (reinforcement clustering)
        - Интерактивная кластеризация с обратной связью от пользователя (interactive clustering with user feedback)
            - Обратная связь в виде корректировки параметров или вида целевой функции
            - **Обратная связь в виде оценки результатов кластеризации**
        - Вспомогательная кластеризация (assisting clustering)

Первым этапом интерактивной кластеризации, очевидно, является обычная кластеризация без учителя. Таким образом, все методы интерактивной кластеризации базируются на методах без учителя, добавляя в них механизмы работы с обратной связью. На рисунке Рис. 1.2 представлена динамика количества публикаций посвященных теме интерактивной кластеризации согласно исследованию [6]. Данное исследование позволяет заметить, что большинство методов интерактивной кластеризации основываются на классических методах кластеризации, таких как: k-means, c-means, вариациях иерархической кластеризации и кластеризации графов. Малое число методов использует нейронные сети, а в случае их использования применяются самоорганизующиеся искусственные нейронные сети SOM (Kohonen self-organized maps).

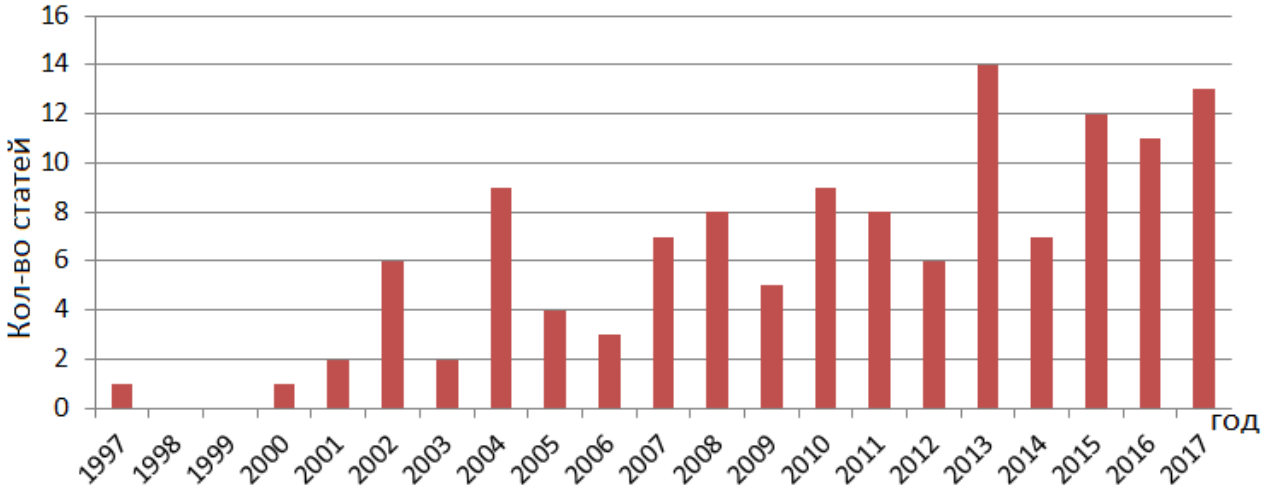


Рис. 1.2. Динамика количества публикаций посвященных теме интерактивной кластеризации

Существуют работы посвященные кластеризации с частично размеченным набором данных на базе нейронных сетей [71, 106], но они используют эту маркировку в процессе первоначального обучения сети [83], а не получают в виде обратной связи. Т.е. не подстраиваются в процессе обработки результатов под нужды исследователя.

1.4. Анализ особенностей обработки коротких текстов

Таблица 1.1. Показывает основные задачи актуальные в области обработки коротких текстов.

Задача	Описание	Методы решения
Распознавание именованных сущностей	Выделение из текста дат, фамилий, наименований географических объектов и т.п.	Морфемный анализ, Языковые модели
Определение тональности текста	Разделение набора данных на тексты с позитивной и негативной тональностью, или по более широкому спектру эмоций	Определение по ключевым словам, Классификаторы на базе языковых моделей

Тематическая классификация	Классификация текстов по заранее определенным тематикам	LDA, Языковые модели

Эти задачи совпадают с основными задачами в области обработки естественного языка (NLP – Natural Language Processing) и решаются схожими методами. При этом обработка коротких текстов отличается дополнительной сложностью. Ниже перечислены и описаны основные проблемы, возникающие при обработке коротких текстов.

Разреженность векторов признаков. При векторизации документов или больших текстов, каждому документу сопоставляется вектор признаков. Элементы этого вектора соответствуют терминам корпуса текстов или другим характерным признакам, подходящим для использования в алгоритмах машинного обучения. При этом само числовое значение элемента вектора является весом соответствующего признака. Веса могут быть рассчитаны различными способами. Наиболее популярной является метрика tf-idf – произведение частоты f(t,d) – количество употреблений термина ‘t’ в документе ‘d’ и обратной частоты idf(t,D) – отношение общего числа документов ‘D’ к количеству документов содержащих термин ‘t’. Математически это выглядит следующим образом: tf_idf(t,d,D) =tf(t,d)*idf(t,D).

В коротких текстах, количество слов крайне мало, и вектор признаков построенный таким образом будет естественным образом сильно разреженным. При этом известные алгоритмы классификации и кластеризации (k-means, HDBScan, тематический анализ [115] и пр.) показывают низкую эффективность при работе с такими векторами.

Полисемия. Наличие более чем одного лексического значения у слова (например, ‘коса’ и менее очевидное ‘паутина’). Таким образом, определение категории для данного слова требует анализа контекста. В большом тексте всегда присутствует необходимый контекст в отличие от коротких текстов, в которых может быть всего несколько слов и контекст определенного слова может

быть не раскрыт или, намерено, оставлен неясным (игра слов, иносказательные высказывания).

Синонимия. Два и более слов имеющих одинаковое или близкое лексическое значение. Например: ‘красивый’, ‘привлекательный’, ‘симпатичный’. В обработке естественного языка синонимы требуют наличия словарей синонимов. В коротких текстах возникает дополнительная трудность – синонимы еще сильнее увеличивают разреженность векторов признаков, при этом они не могут быть объединены в один признак, так как при этом может потеряться оттеночное значение, которое могло быть важно в данном тексте.

Использование аббревиатур, сленговых слов и неологизмов. Зачастую, системы, в которых пользователи создают короткие тексты, такие как Twitter, ограничивают общую длину сообщения, что мотивирует пользователей на использование сокращений и аббревиатур. Дополнительная мотивация сокращать слова у пользователя возникает, в следствии внесения текста через мобильные устройства с неудобной для ввода длинных слов клавиатурой. Также тексты всевозможных социальных сетей изобилуют сленгом и неологизмами. Если рассмотреть набор коротких текстов по КПЭ системы стратегического планирования РФ, то в ней наблюдается большое число аббревиатур и специальных терминов.

Проблема опечаток, грамматические и пунктуационные ошибки. В отличие от больших текстов в различных изданиях, короткие тексты из социальных сетей и прочих систем, предполагающих генерацию текстов пользователями, не предполагают специальной фазы рецензирования и редактуры, в итоге это приводит к тому, что уровень грамотности формулировки таких текстов крайне низок. Большинство объектов в наборах коротких текстов вводятся с мобильных устройств с неудобной для ввода клавиатурой, что приводит к большому числу опечаток. Интеллектуальные помощники могут даже ухудшать ситуацию, заменяя слово с явной опечаткой на другое слово без опечатки, но не подходящее по смыслу. В итоге автор просто не замечает опечатки. Пра-

вила пунктуации способные существенно влиять на смысл высказываний массово не соблюдается в среде социальных сетей.

1.5. Анализ современных методов языкового моделирования

Задача языкового моделирования в узком смысле — спрогнозировать следующее слово в тексте, зная последовательность предшествующих слов. Результат решения данной задачи имеет конкретное практическое применение: интеллектуальные клавиатуры, генерация ответа на e-mail [47], исправление опечаток.

Первоначально были предложены подходы, в основе которых лежит N-граммная модель. Методы сглаживания позволяют обработать N-граммы, которые модель не встречала [45].

Первая языковая модель на основе искусственной нейронной сети, была предложена Йошуа Бенжио [16]. В работе предложена следующая схема функционирования:

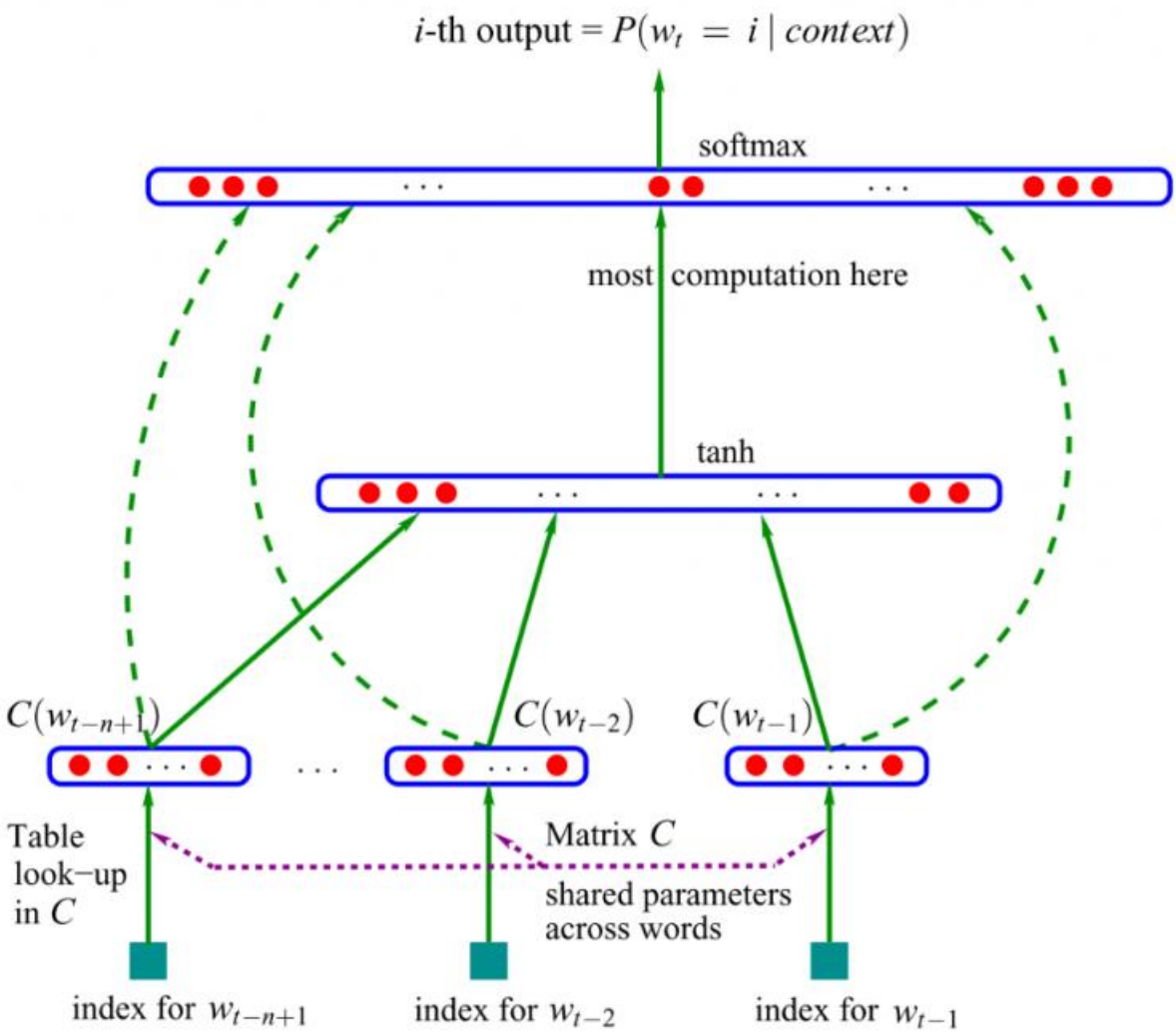


Рис. 1.3. Схема работы первой нейронной языковой модели [8]

На вход данной модели подается набор векторных представлений n предшествующих слов. Сжатые векторные представления называют эмбеддингами (word embedding). Эти векторы конкатенируются и передаются на скрытый слой нейронной сети. Выходные данные скрытого слоя затем передаются в слой с функций softmax.

Позднее для решения задачи языкового моделирования вместо нейронных сетей с прямой связью начали использоваться рекуррентные нейронные сети [73], а еще позднее сети с долгой краткосрочной памятью [30]. В последние годы предложено много новых языковых моделей, расширяющих возможности классических LSTM-сетей. Более детально использование нейронных сетей в задаче языкового моделирования рассмотрено ниже.

1.5.1. Многозадачное обучение

Подход, при котором модель обучается решению различных задач на одном и том же наборе данных называется - многозадачное обучение. В искусственных нейронных сетях многозадачности добиваются за счет переиспользования части слоев сети. При многозадачном обучении сеть создает внутри себя представления данных, пригодных для решения ряда задач. Происходит обучение общим низкоуровневым представлениям. Идея была впервые предложена в 1993 году Ричем Каруаной для прогнозирования пневмонии [21].

Впервые многозадачное обучение нейросетей для обработки естественного языка применили Коллобер и Уэстон [26]. В модели использовалась матрица векторных представлений слов общая для двух под-моделей, решающих различные задачи:

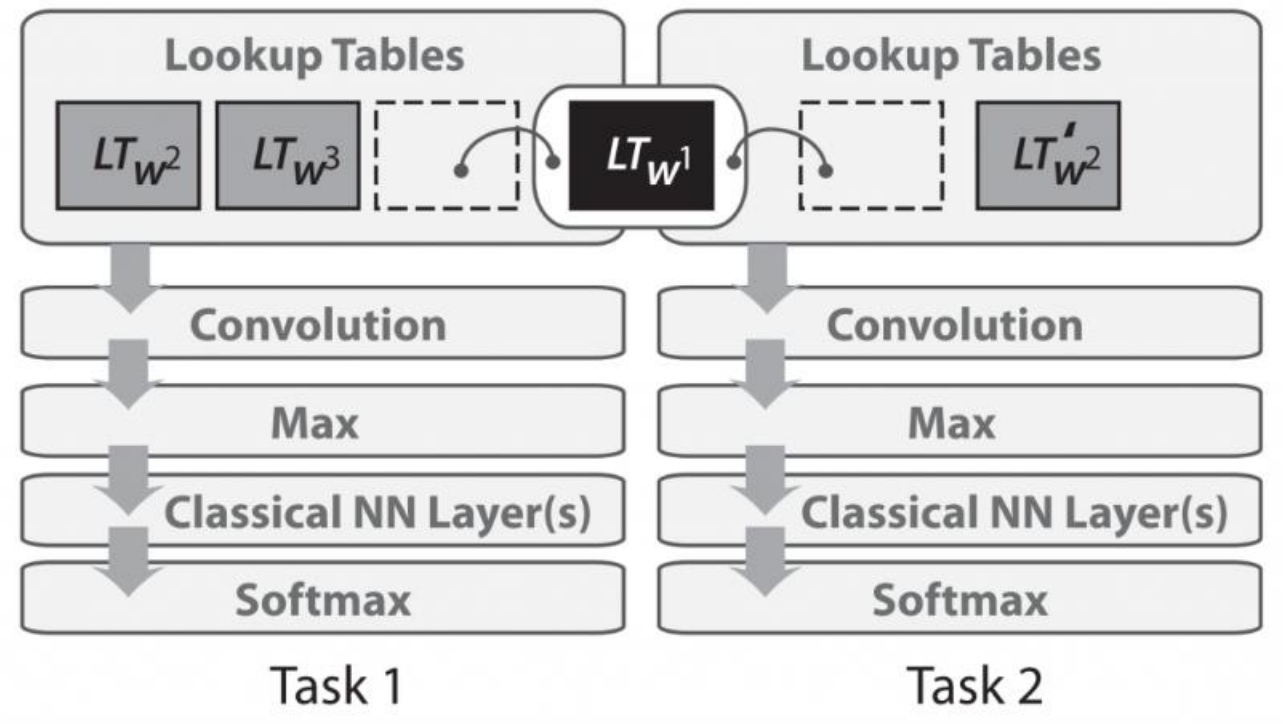


Рис. 1.4. Совместное использование матрицы векторных представлений [11].

Совместное использование одних и тех же векторных представлений слов позволяет моделям взаимодействовать и обмениваться общей информацией низкоуровневого характера, т.е. некоторыми «базовыми» представлениями об элементах текстов.

Многозадачное обучение позволяет решать самые различные задачи, связанные с обработкой естественного языка. При этом для целей получения качественных сжатых векторных представлений, сеть заставляют решать искусственную задачу, например, поиск всех упоминаний животных в тексте. Цель решения такой задачи не в получении высоких результатов, а в улучшении качества прогнозирования слов по последовательности [94].

1.5.2. Языковые модели на основе искусственных нейронных сетей

Рекуррентные нейронные сети (RNN) повсеместно встречаются при обработке естественного языка, так как позволяют учесть последовательность слов. Развитием RNN сетей стали классические LSTM-сети (сети с долгой кратковременной памятью [42]). LSTM-сети позволяют эффективно бороться с возникающей проблемой исчезающего (или взрывного) градиента. До 2013 года обучение RNN являлось трудной задачей. В работе Ильи Суцкевера [84] была предложена схема ячейки LSTM-сети (рис. 5). Двунаправленная LSTM-сеть [33] используется для работы одновременно с левым (предшествующим единице) и правым (последующим) контекстом.

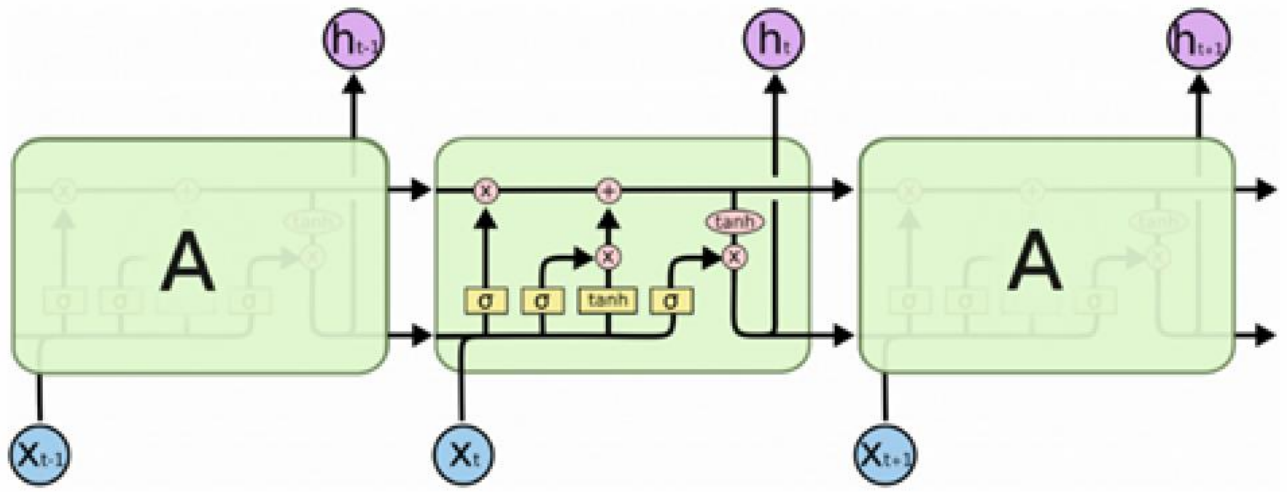


Рис. 1.5. Концептуальная схема LSTM-модели [18].

Широко используемые в компьютерном зрении сверточные сети (CNN) могут использоваться и в обработке естественного языка [48]. Для обработки текстов, весовые коэффициенты сверточной нейросети задаются в двух измере-

ниях, причем фильтры необходимо перемещать только во временном измерении. На Рис. 1.6 представлена схема типичной сверточной нейронной сети, использующейся для обработки естественного языка.

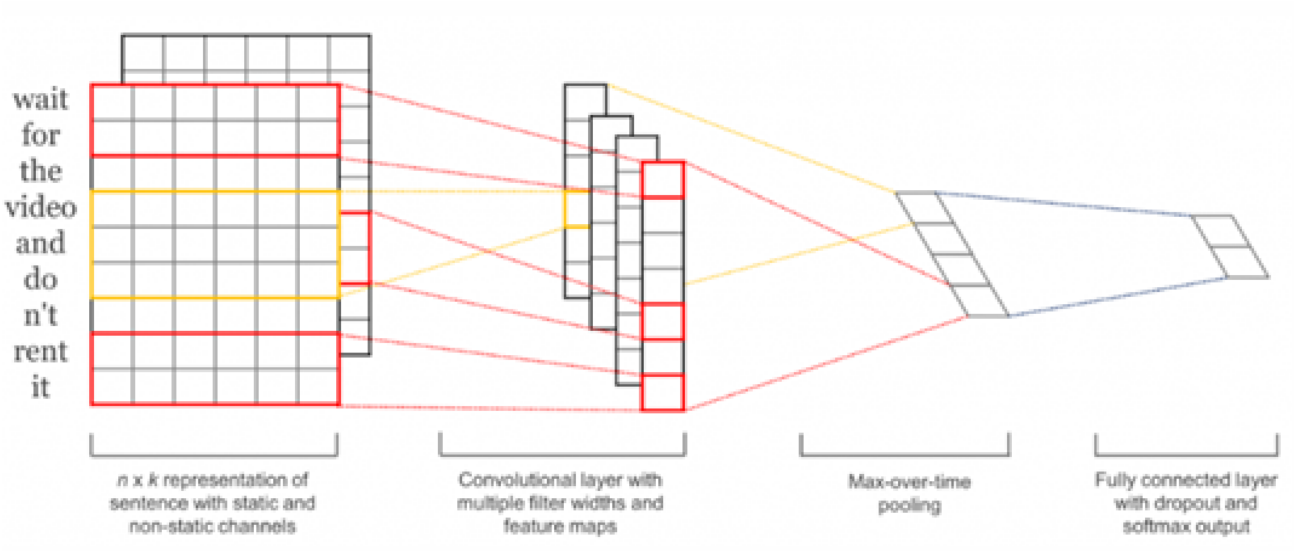


Рис. 1.6. Сверточная нейронная сеть для обработки текстов [25].

Преимущество сверточных нейронных сетей состоит в том, что их можно распараллелить в большей степени, чем РНС, поскольку состояние нейронных элементов на каждом временном шаге зависит только от локального контекста (благодаря операции свертки), а не от всех прошлых состояний, как в РНС. С использованием расширенных сверток, а значит расширением полей восприятия, могут быть расширены и сами СНС, что позволит им охватить более широкий контекст [48]. СНС и LSTM-сети также могут быть объединены и вложены друг в друга [87], а свертки могут быть использованы для ускорения LSTM-сетей [18].

С ростом качества языковых моделей росли и их размеры. На текущий момент количество параметров в языковых моделях варьируется от сотен миллионов до десятков миллиардов. Обучение такой искусственной нейронной сети занимает значительное количество временных и вычислительных ресурсов. Это снижает эффективность решения ряда задач. Также обучение таких громоздких нейронных сетей накладывает ограничения на объемы обучающих выборок, что делает не применимым такие подходы в языковых доменах, где объ-

емы корпусов текстов не такие значительные, как, например, объем текстов Википедии на английском языке. Многие современные языки с ограниченным числом носителей не могут быть обработаны такими языковыми моделями. Аналогичная ситуация для специфичных языковых доменов, например текстов относящихся к узким научным дисциплинам. Для преодоления обозначенных ограничений применяются два основных приема: использование предварительно обученных векторных представлений слов и использование предварительно обученных языковых моделей.

Предварительно обученные векторные представления слов не зависят от контекста, который анализируется в данный момент, и используются только для инициализации весов первого слоя в наших моделях. Это позволяет уменьшить число параметров нейронной сети и получить хорошие стартовые значения для первого слоя, отвечающего за векторные представления слов. В последнее время целый ряд задач обучения с учителем использовался для предварительного обучения нейронных сетей [28, 69]. Для обучения языковым моделям требуется лишь неразмеченный текст; и таким образом, масштабы обучения могут достигать миллиардов токенов, новых доменов и новых языков.

1.5.3. Предварительно обученные языковые модели

Предварительно подготовленные языковые модели были впервые предложены в 2015 году [29], но только недавно их полезность при решении самых разных задач была доказана. Встроенные языковые модели могут использоваться как составляющие целевой модели [88], или языковая модель может быть точно настроена на данные задачи-адресата [89]. Встраивание языковой модели значительно улучшает даже самые передовые решения множества различных задач, таких как: поиск именованных сущностей (NER), вопросно-ответные системы (SQuAD), синтаксический разбор и т.п.

Предварительно обученные языковые модели доказали возможность обучения на значительно меньшем количестве данных. Поскольку для языковых

моделей требуются лишь неразмеченные данные, они особенно полезны при работе с малоресурсными языками с дефицитом размеченных данных [15,103].

Наиболее популярными предварительно обученными сетями на сегодняшний день являются: ULMFiT (русский и английский языки); ELMo (мульти-язычная) и RuBERT (русскоязычная адаптация модели BERT от корпорации “Google”).

1.5.3.1. Языковая модель ULMFiT

Языковая модель Universal Language Model Fine-tuning (ULMFiT) была предложена в 2018 году [52]. Архитектура модели, в основе которой лежат три LSTM слоя, представлена на Рис. 1.7 [52]. Основными преимуществами данной языковой модели являются:

- Небольшое (относительно других рассматриваемых языковых моделей) количество параметров нейронной сети – около 3,5 млн. Это позволяет производить дополнительную настройки модели на видео-ускорителях архитектуры Pascal GTX 1080 Ti в течении суток для корпусов состоящих из нескольких миллионов текстов.
- Высокое качество языковой модели. Перплексия предварительно обученной языковой модели для русского языка на корпусе новостей интернет ресурса “Лента.ру” составляет 21.98, а точность 43%.
- Реализация модели доступна в одном из наиболее популярных пакетов машинного обучения FastAI [51] и входит в стандартный набор подключенных библиотек ресурса GoogleColab – крупнейшего бесплатного ресурса для исследователей в области машинного обучения.
- Доступные реализации модели позволяют проводить дополнительное обучение всех слоев модели, что дает возможность максимальной подстройки модели под решаемую задачу.
- Значительный объем словаря языковой модели. Предварительно обученная модель для русского языка содержит словарь из 60 000 словоформ. Лемма-

тизация не использована (влияние лемматизации на качество языковой модели рассматривается отдельно в Главе 4).

- Словарь данной языковой модели может быть изменен и расширен без существенных потерь качества модели (алгоритм расширения словаря языковой модели представлен в данной работе в Главе 2).

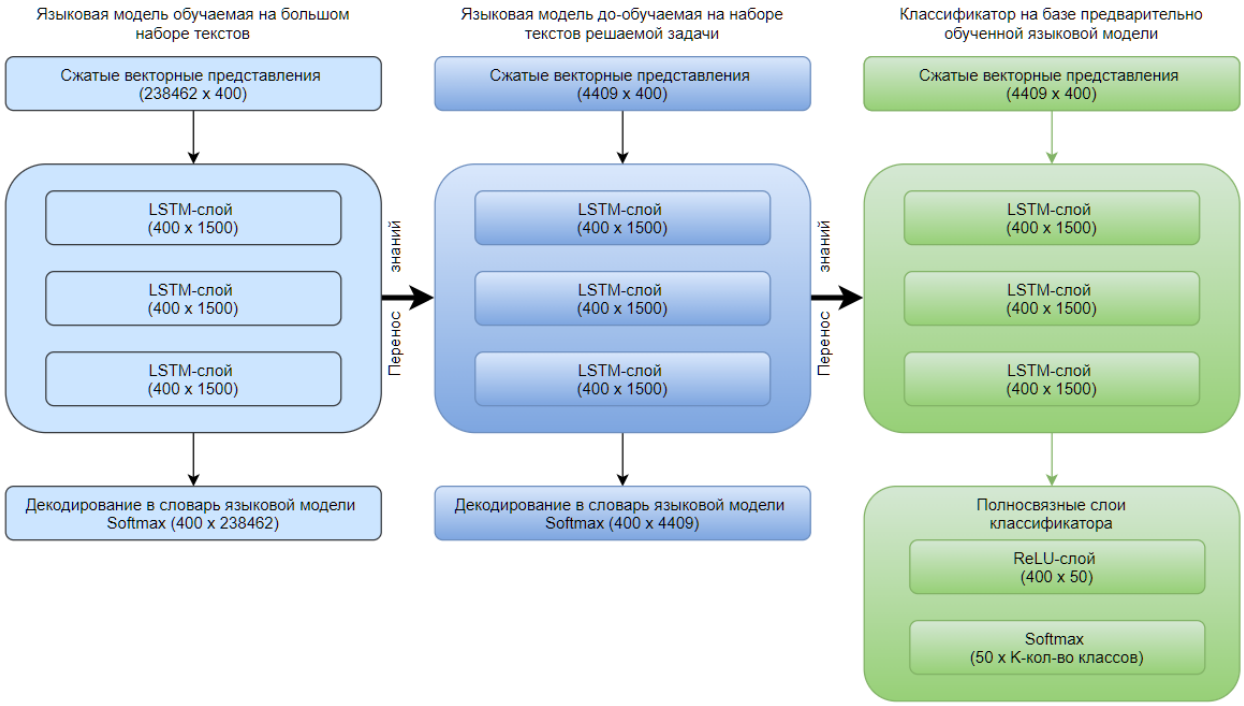


Рис. 1.7. Архитектура и схема работы языковой модели ULMFiT

Также на Рис. 1.7 представлена предложенная в работе [52] принципиальная схема использования языковой модели в задачах обработки текстов. Схема включает в себя два этапа: этап дополнительной настройки параметров языковой модели под исследуемый набор текстов и собственно этап решения поставленной задачи путем дополнения кодера языковой модели несколькими слоями (в основном используются полносвязные слои) и стандартным обучением сети. Эта схема в дальнейшем использована при работе со всеми появляющимися языковыми моделями.

1.5.3.2. Языковая модель ELMo

ELMo, сокращение от Embeddings from Language Model (Peters, et al, 2018) обеспечивает получение контекстуализированного представления слов, пред-

варительно обучая языковую модель неконтролируемым образом. На Рис. 1.8 представлена архитектура языковой модели ELMo [88]. Модель состоит из двухслойного двунаправленного кодера LSTM и основного модуля прогнозирования. Основными достоинствами данной языковой модели являются:

- Как следует из названия модели, векторные представления слов получаемые в результате работы модели учитывают контекст фразы в которой они были использованы.
- Универсальный словарь данной модели состоит из букв языка, для которого происходит обучение модели. Также возможно обучение мультиязычной модели. В таком случае в качестве словаря формируется объединение алфавитов всех языков корпуса текстов используемого для обучения модели.
- Для русского языка существует и мультиязычный и специально обученный варианты модели ELMo.

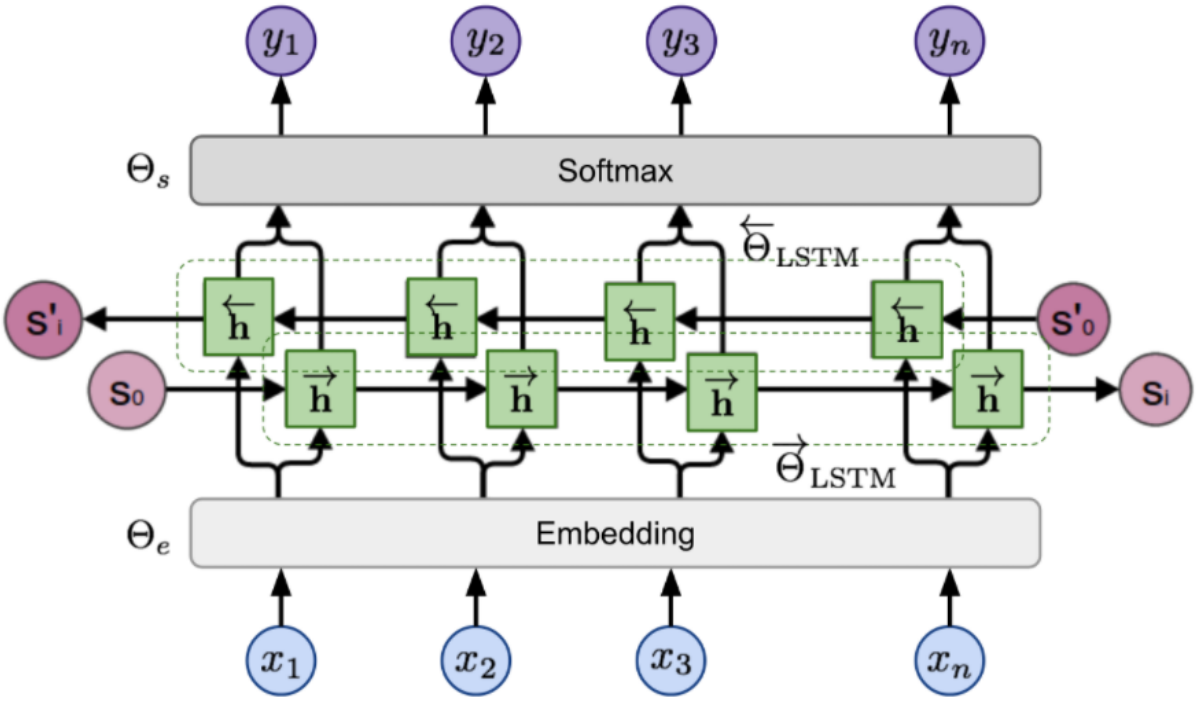


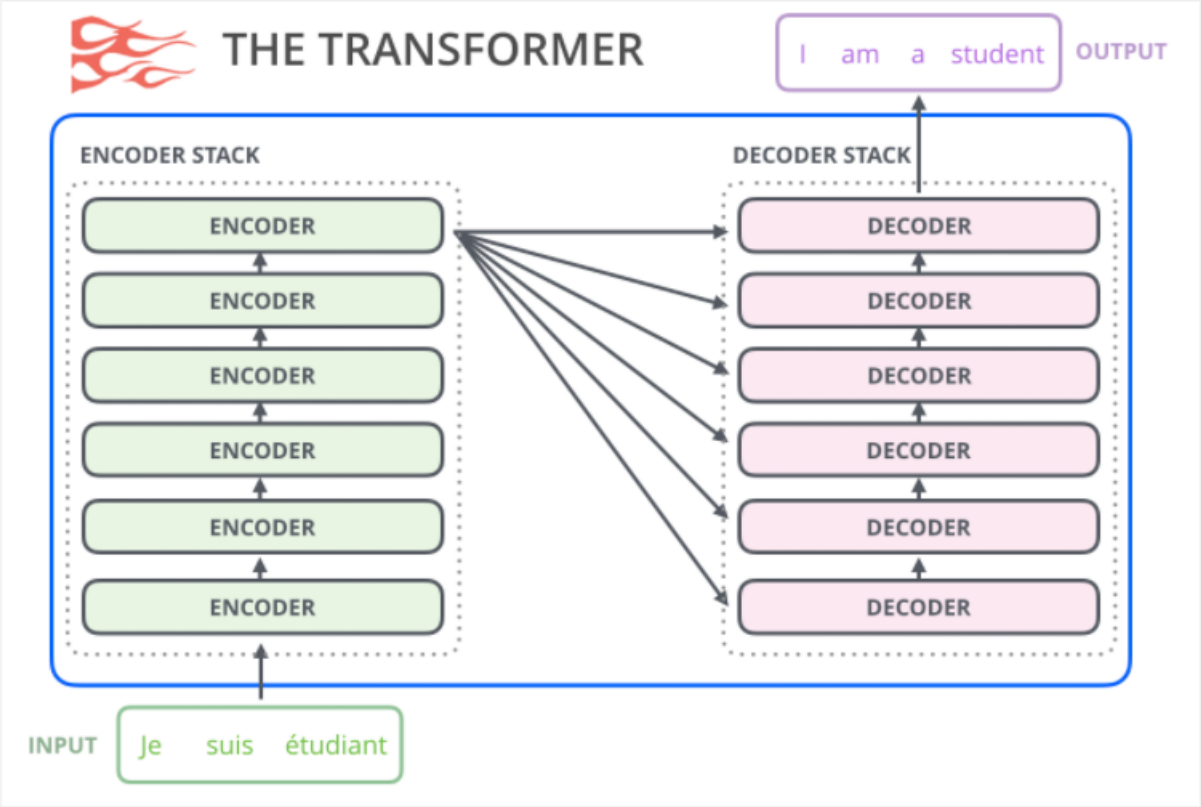
Рис. 1.8. Архитектура языковой модели ELMo [88].

Основным назначением данной языковой модели является получение векторных представлений слов. Модель ELMo успешно используется как альтернатива традиционным моделям word2vec, таким как skip-gram и CBOW [79].

Перплексия версии данной модели обученной на корпусе русскоязычной Википедии составляет 43.69, это ниже модели ULMFiT. Таким образом, данная модель не является наилучшим вариантом языковой модели для кластеризации текстов.

1.5.3.3. Языковая модель RuBERT

В основе модели BERT лежит архитектура нейронных сетей модели sequence-to-sequence, которая в свою очередь основана на архитектуре Transformer представленной на Рис. 1.9 [60,61,32]. Она состоит из последовательно расположенных кодеров и декодеров, позволяющих преобразовывать цепочку слов во входящей последовательности в цепочку слов в исходящей последовательности.



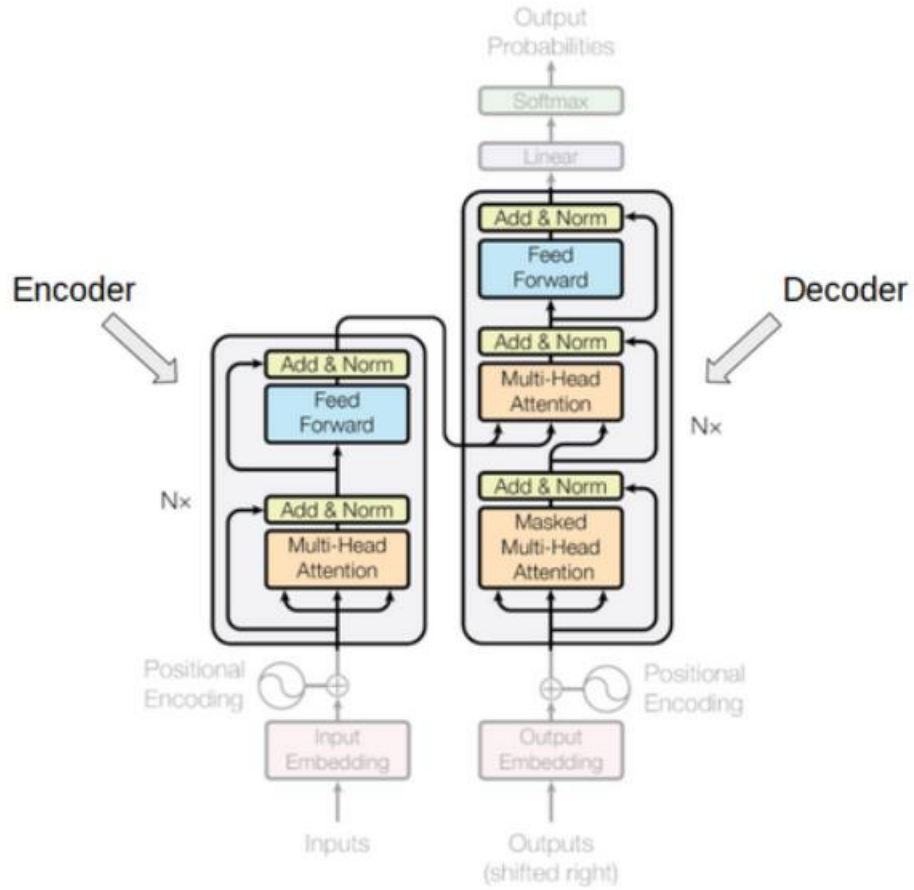


Рис. 1.9. Схема работы и архитектура модели Transformer [132].

Модель BERT (Bidirectional Encoder Representations from Transformers) разработанная и обученная корпорацией “Google”, представлена в двух вариантах с 12 или 24 слоями кодеров в архитектуре Transformer соответственно (Рис. 1.10) [105]. Каждая из модификаций модели BERT от компании “Google” была обучена на двух корпусах текстов: англоязычном и мультиязычном. В 2020 году мультиязычная модель BERT на 12 слоев кодеров дополнительно обучена на русскоязычном наборе текстов интернет-ресурса “Лента.ру” и доступна под названием RuBERT [19]. Основными достоинствами модели RuBERT являются:

- Использование токенов-морфем в качестве словаря языковой модели. Каждое слово на входе в модель раскладывается на ряд токенов – морфем специфичных для используемого естественного языка. Потенциально такой подход позволяет модели работать с ранее неизвестными для модели словами за счет распознавания в них известных морфем.

- Большое число параметров нейронной сети (100 миллионов для базового варианта и 340 миллионов для расширенного варианта модели) позволило построить наиболее полную языковую модель для английского языка. На данной модели были получены рекордные результаты по основным задачам обработки естественного языка.

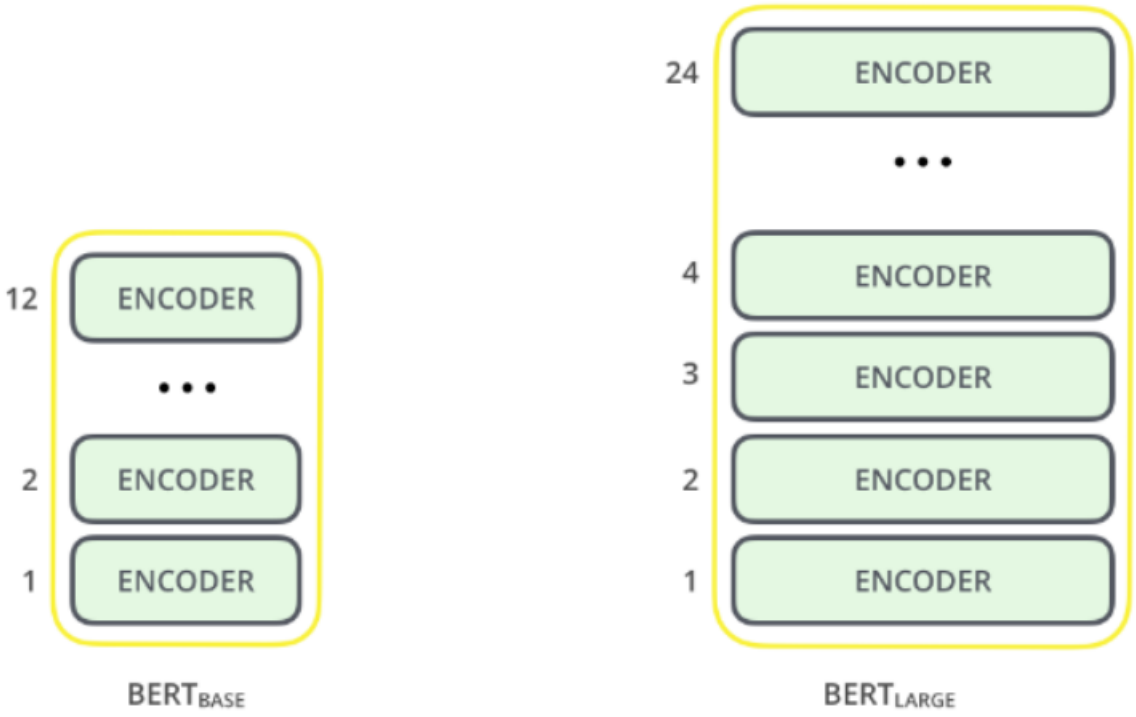


Рис. 1.10. Варианты архитектур модели BERT

Вариант модели для русского языка RuBERT изучен не достаточно полно. Нет данных по перплексии полученной модели, однако следует отметить, что результаты уже проведенных экспериментов по основным задачам обработки естественного языка показывают высокое качество результатов [66]. Малое количество доступных результатов экспериментов можно объяснить скоростью дополнительного обучения модели RuBERT, которая ниже в 3-4 раза относительно ULMFiT. Полное обучение модели BERT для русского языка представляется возможным только для международных корпораций и крупных исследовательских институтов, а значит, данный подход не эффективен для решения прикладных задач индивидуальных исследователей или небольших групп ис-

следователей. Также следует отметить отсутствие возможности смены словаря языковой модели в доступных реализациях модели BERT и не изученность на момент написания работы влияния словаря состоящего из токенов-морфем на качество языковой модели.

Вышеизложенные соображения не позволили использовать модель RuBERT в данной работе. Все эксперименты проведены с помощью, обученной для русского языка, модели архитектуры ULMFiT [51]. При этом важно отметить, что предлагаемый в данной работе алгоритм нечеткой кластеризации коротких текстов не зависит существенным образом от выбора конкретной языковой модели. Только шаг расширения словаря языковой модели использует особенности архитектуры ULMFiT, но при этом и он допускает обобщение на случай использования альтернативных языковых моделей.

1.6. Анализ современных методов кластеризации коротких текстов

Решение задачи поиска близких объектов, в частности коротких текстов, полезно для получения осмысленного результата при решения задач обработки естественного языка. Например, пользователь веб-сайта может потратить значительное время в поисках необходимой информации, которая должна присутствовать в наборе данных. Более того, различные веб-сайты часто публикуют одну и ту же информацию, относящуюся к интересующей теме, что в итоге увеличивает объем информации, которую необходимо просмотреть при поиске [74].

В последние несколько лет проблема кластеризации коротких текстов активно исследуется различными исследователями. В работах [2,6,9,11,14] решается проблема разреженности вектора признаков при кластеризации коротких текстов с помощью внешней базы знаний. Hotho и др. предлагают использовать базу синсетов WordNet для расширения коротких текстов [45]. Некоторые работы предлагают использование Wikipedia как базу знаний для кластеризации коротких текстов [2,11,14]. Результаты выдачи поисковых машин также могут быть использованы как инструмент расширения коротких текстов [108]. Sahami

и Heilman предлагают решать проблему кластеризации коротких текстов с помощью новой меры сходства между ними. Banerjee и др. [11] создали индекс в инструменте Lucene для всех статей в Wikipedia и затем использовали короткие тексты как поисковые запросы для извлечения наиболее подходящих статей как способ расширения коротких текстов. В 2006 году, Gabrilovich [40] показал эффективность использования Wikipedia как дополнительного ресурса для построения векторов признаков в задаче категоризации текстов. Часто используемый подход в литературе заключается в использовании лексических ресурсов (тезаурусов, онтологий), таких как WordNet, для сравнения коротких текстов [2]. WordNet предоставляет собой размеченную лексическую базу данных слов английского языка. С помощью использования семантических отношений между термами WordNet предлагается снимать лексическую неоднозначность при сравнении текстов. Основным недостатком такого подхода является сложность и дороговизна создания и поддержания подобных лексических ресурсов. Для некоторых специфических лексических доменов построение таких ресурсов может быть крайне затруднительно. Janruang и Guha [2] предлагают кластеризацию дерева семантических суффиксов которое строится проходом “в глубину” и “в ширину” с использованием мер семантический близости и сравнения строк. Quan и др. [13] получал отношения между не часто используемыми термами коротких текстов. После чего, векторы признаков для двух коротких текстов модифицировались согласно полученной мере сходства, и финальное расстояние между текстами считалось как косинусное расстояние. Xiaohui и др. [18] решая задачу кластеризации коротких текстов расширяли список ключевых слов с помощью графа концептов. Эффективность метода была показана с использованием метрик “precision, recall и F-score” [56].

При этом результаты работы вышеперечисленных методов детерминированы и полностью определяются исходной выборкой, метрикой и используемым алгоритмом. Результатом работы является четкое или нечеткое разбиение по кластерам и определенная эвристическая оценка качества предлагаемого

разбиения [117, 120]. Исследователь данных имеет возможность выдвигать гипотезы в виде формирования различных наборов признаков для кластеризации. При этом деятельность исследователя данных заключается не только в переборе ряда гипотез, но и в случае подтверждения определенной гипотезы в ее уточнении и развитии. Для решения второй задачи перечисленные методы не подходят, так как варианты уточнения возникают исходя из результатов кластеризации, а их корректировка должна осуществляться на входном векторе признаков.

Отсутствие возможности применять методы кластеризации к результатам предыдущего шага кластеризации затрудняет реализацию итеративного анализа. Следует разработать такой метод кластеризации форматы выходных и выходных данных которого были бы идентичны, и при этом исследователь мог бы дополнять входные данные различными гипотезами.

1.7. Постановка задач исследования

Таким образом, целью данной работы является построение эффективного алгоритма нечеткой интерактивной кластеризации коротких текстов на базе искусственной нейронной сети с современной архитектурой, которая бы включала в себя предварительно обученную языковую модель как блок, отвечающий за преобразование текста в сжатое векторное представление. Данный алгоритм должен иметь возможность учета обратной связи от эксперта, которая бы давалась в виде оценки результатов кластеризации. Для проверки работоспособности и оценки эффективности алгоритма необходимо реализовать программный модуль для системы поддержки принятия решений для автоматизации работ по нечеткой интерактивной кластеризации коротких текстов.

1.8. Выводы по главе

В данной главе были рассмотрены современные подходы к четкой и нечеткой кластеризации коротких текстов. Отмечены принципы построения со-

временных универсальных алгоритмов кластеризации и принципиальные преимущества алгоритмов кластеризации основанных на использовании искусственных нейронных сетях.

Проведен сравнительный анализ современных подходов к обработке естественного языка. Разобрана специфика обработки коротких текстов. Рассмотрены принципы многозадачного обучения и переноса знаний. Отмечена роль языковых моделей, являющихся наиболее эффективным на данный момент инструментом обработки текстов на естественных языках. В ходе анализа доступных предварительно обученных языковых моделей обоснован выбор языковой модели ULMFiT как основной для данной работы.

Также проведен подробный анализ методов кластеризации с ограничениями и в частности методов интерактивной кластеризации. Показана динамика количества исследований в данной области свидетельствующая о нарастающей актуальности этого класса методов. Указаны преимущества интерактивной кластеризации при работе с текстами.

ГЛАВА 2. Исследование и разработка моделей и методов нечеткой интерактивной кластеризации с обратной связью от эксперта

В данной главе предложен метод интерактивной кластеризации с обратной связью на базе современных методов кластеризации. В трех разделах главы представлены: архитектура искусственной нейронной сети позволяющая решить задачу кластеризации коротких текстов на базе языковой модели; метод расширения словаря языковой модели на базе нейронной сети с использованием иерархического классификатора и метод корректировки весов нейронной сети позволяющий учитывать ограничения задаваемые пользователем при решении задачи кластеризации.

2.1. Модель кластеризации коротких текстов

Современные методы кластеризации с использованием нейронных сетей, как правило, используют нейронную сеть для подготовки векторов признаков и затем используется аналитический метод (основанный на формулах с гиперпараметрами) для кластеризации этих признаков. В итоге результат кластеризации обусловлен качеством полученных векторов признаков, т.е. качеством обучения нейронной сети. При этом в последнее время появляются методы позволяющие решать задачу кластеризации непосредственно с помощью нейронной сети, что позволяет объединить процесс получения векторов признаков и собственно кластеризации.

В своем исследовании [1] авторы предлагают обобщенную схему построения современных методов кластеризации с использованием нейронных сетей, в которую укладывается подавляющее большинство методов (Рис. 1.1) [33, 108,112]. Схема построения метода кластеризации принципиально объединяет

этапы конструирования векторов признаков и группировку объектов в кластеры за счет использования общей целевой функции.

Рис. 2.11. Обобщенная схема построения методов кластеризации на базе нейронных сетей

На Рис. 2.12 приведена схема работы алгоритма DEC и архитектура используемой в нем нейронной сети. Подход, использованный в DEC, полностью удовлетворяет вышеизложенной схеме. Первоначальная идея метода изложена в статье [108]. В началае конструируется автоэнкодер, для того чтобы получить сжатые векторные представления на выходе блока кодера. Автоэнкодер предлагалось обучать на исследуемом наборе данных, что как уже было отмечено выше, в случае коротких текстов не всегда возможно. Далее кодер совмещается с блоком отвечающим за определение центров кластеров и продолжается обучение сети с использованием целевой функции в виде меры Кульбака-Лейблера.

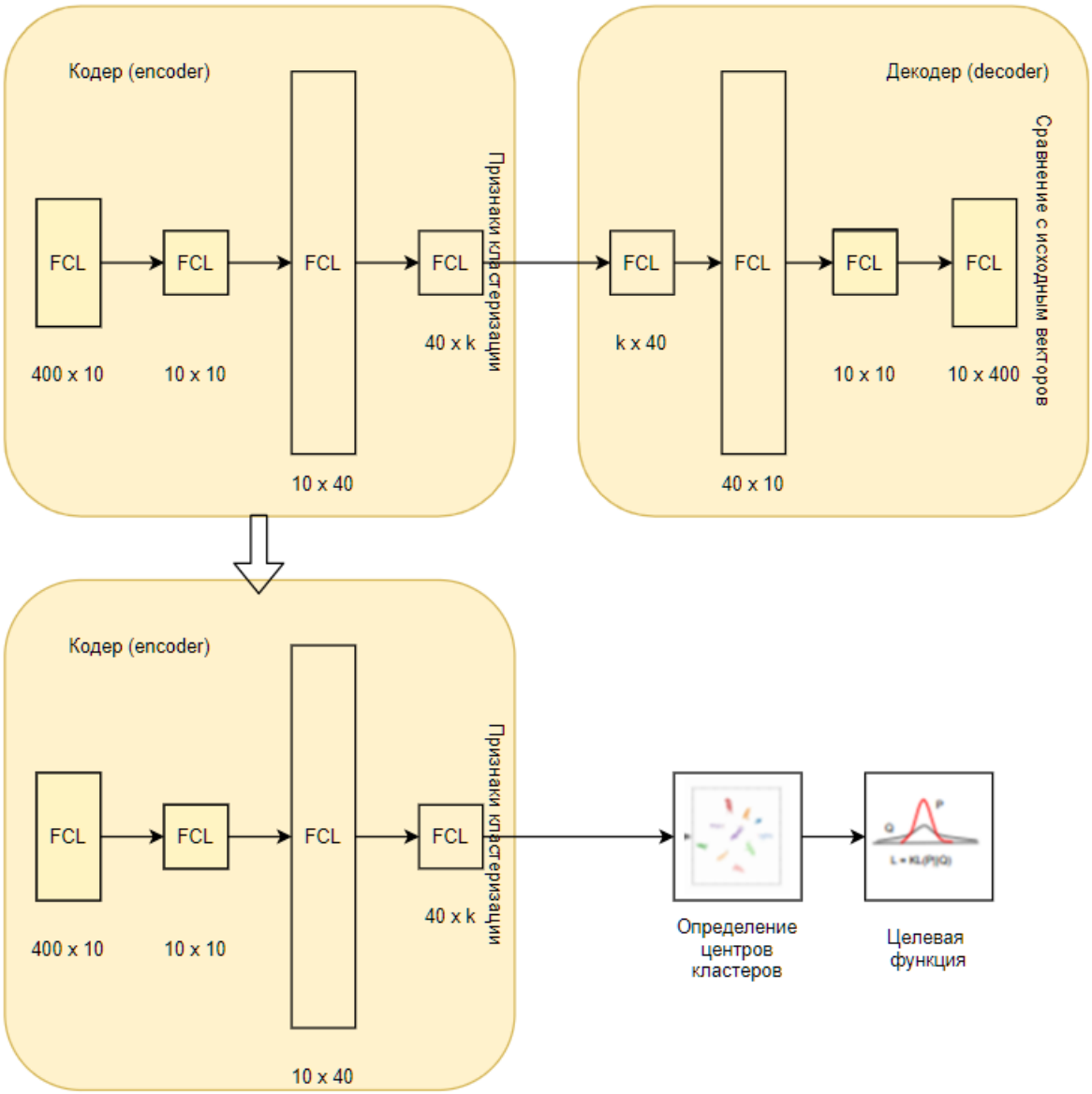


Рис. 2.12. Схема работы и архитектура нейронной сети алгоритма DEC

В результате выполнения диссертационного исследования предлагается для обработки текстов дополнительно перед кодером вышеописанной архитектуры использовать кодер языковой модели. Архитектура предлагаемой нейронной сети представлена на Рис. 2.13. За кодером языковой модели следуют слои, которые отвечают за построение векторов признаков кластеризации на базе векторов признаков текстов. При этом слои отвечающие за получение векторов признаков для кластеризации также инициализируются как часть автоэнкодера.

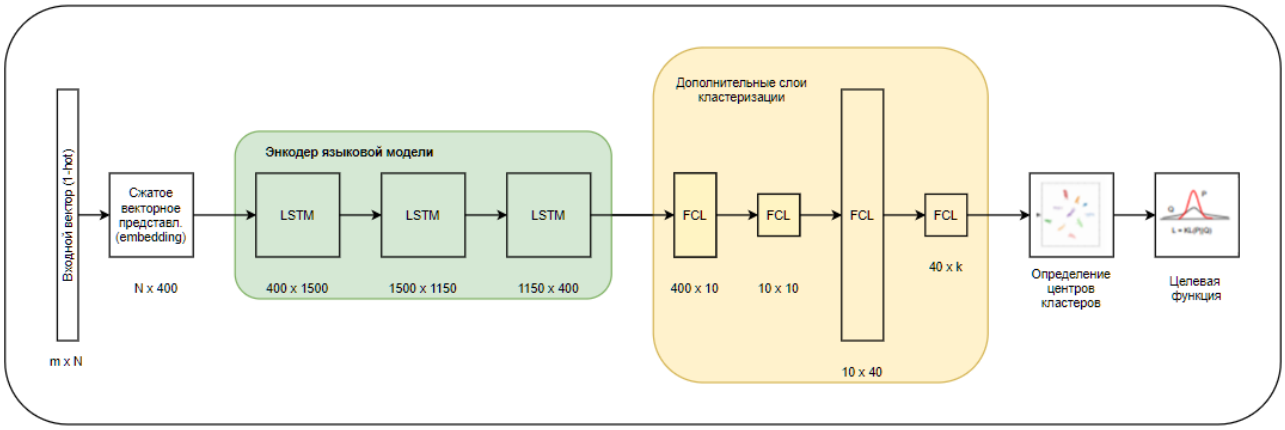


Рис. 2.13. Архитектура нейронной сети для обработки текстов

Рассмотрим подробнее разработанный в ходе выполнения диссертационного исследования метод кластеризации. Как уже было указано выше, в качестве базового выбран метод кластеризации DEC (Unsupervised Deep Embedding for Clustering Analysis), при этом аналогичный подход может быть применен ко множеству других алгоритмов, например, к DEPICT [33]. Входной набор данных $X=\{x_i \mid i \in [0, N)\}$, N – кол-во элементов в наборе. Это множество с помощью энкодера, являющегося частью заранее обученного автоэнкодера, отображается в пространство меньшей размерности: $f_\theta: X \rightarrow Z$, где θ – параметры нейронной сети, Z – скрытое пространство признаков. Пространство признаков называется в данном случае скрытым, т.к. его построение происходит в процессе обучения нейронной сети и затем оно формируется неявным образом в процессе обучения автоэнкодера и решения задачи кластеризации. В данной работе используется кодер следующей структуры: d-10-10-40-k, где d-размерность входного набора данных, k –число кластеров. Результатом работы алгоритма является набор центров кластеров в пространстве Z: $\{\mu_j \in Z \mid j \in [0, k)\}$, где k – заданное число кластеров. Для инициализации весов дополнительных слоев кластеризации используется автоэнкодер, в декодере которого симметрично повторяются слои кодера. Применяется типовая схема обучения автоэнкодера: вначале каждый слой обучается отдельно, и затем дополнительно все слои обучаются вместе. Инициализация центров кластеров происходит при помощи ал-

горитма k-means, который применяется к представлению векторов, полученному в результате обучения автоэнкодера.

Процессы определения оптимального расположения центров кластеров и построения пространства признаков происходят одновременно за счет определения общей функции потерь. Для этого в качестве меры расстояния между элементом и центром кластера используется метрика, основанная на распределении Стьюдента с одной степенью свободы.

$$\frac{(1 + ||z_i - \mu_j||^2)^{-1}}{\sum_{l=0}^{k}(1 + ||z_i - \mu_l||^2)^{-1}}). \tag{2.1}$$

Целевая функция (функция потерь, loss function) или штрафная функция строится как метрика Кульбака-Лейблера (Kullback-Leibler divergence) между фактическим и целевым распределением.

$$L = KL(P||Q) = \sum_i \sum_j p_{ij} \log \frac{p_{ij}}{q_{ij}}. \tag{2.2}$$

В качестве целевого распределения используется следующее распределение:

$$p_{ij} = \frac{{q_{ij}}^2/f_j}{\sum_{l=0}^{k} {q_{il}}^2/f_l}, \qquad f_j = \sum_j q_{ij}. \tag{2.3}$$

Это распределение обладает следующим свойствами: усиливает вклад от элементов с большой долей принадлежности кластеру и нормализует влияние больших кластеров, не позволяя им чрезмерно притягивать к себе удаленные элементы за счет своего размера.

Для обновления весов нейронной сети и пересчета центров кластеров используются следующие частные производные целевой функции:

$$\frac{\partial L}{\partial z_i} = 2 \sum_j \left(1 + \left|\left|z_i - \mu_j\right|\right|^2\right)^{-1} * (p_{ij} - q_{ij}) * (z_i - \mu_j), \quad (2.4)$$

$$\frac{\partial L}{\partial \mu_j} = -2 \sum_i \left(1 + \left|\left|z_i - \mu_j\right|\right|^2\right)^{-1} * (p_{ij} - q_{ij}) * (z_i - \mu_j). \quad (2.5)$$

Зачастую при решении задачи кластеризации имеется гипотеза или требования к равномерности распределения элементов по кластерам. Для этого в целевую функцию можно добавить слагаемое добавляющее штраф за неравномерность распределения. Таким образом, целевая функция будет иметь вид:

$$\begin{aligned} \mathcal{L} &= KL(\mathbf{Q}\|\mathbf{P}) + KL(\mathbf{f}\|\mathbf{u}) \\ &= \left[\frac{1}{N}\sum_{i=1}^{N}\sum_{k=1}^{K} q_{ik} \log \frac{q_{ik}}{p_{ik}}\right] + \left[\frac{1}{N}\sum_{k=1}^{K} f_k \log \frac{f_k}{u_k}\right] \\ &= \frac{1}{N}\sum_{i=1}^{N}\sum_{k=1}^{K} q_{ik} \log \frac{q_{ik}}{p_{ik}} + q_{ik} \log \frac{f_k}{u_k}, \end{aligned} \quad (2.6)$$

2.2. Предобработка словаря произвольного набора текстов для подготовки к использованию предварительно обученной языковой модели

Как уже было показано в Главе 1 для обучения языковой модели была выбрана модель с архитектурой ULMFit. При этом, для ULMFit список входящих токенов (словарный запас нейронной сети реализующей языковую модель) ограничен и его рекомендованное значение (полученное опытным путем) равно 60 000 токенам. Учитывая, что в русском языке количество словоформ существенно больше, чем в английском, то данное ограничение является значимым для решения реальных задач в домене русского языка. Также модели, обучен-

ные на общих корпусах текстов, как правило, не включают в свой словарь неологизмы, жаргонизмы, нецензурную лексику и архаизмы.

В то же время, современные модели распределенных векторов слов содержат 200-400 тысяч лемм (токенов). Совместное использование модели распределенных векторов слов и пред-обученной языковой модели на базе нейронной сети решает обозначенную проблему, не прибегая к ресурсоемким вычислениям, и позволяет получить хорошие результаты.

В данной диссертационной работе предлагается добавить промежуточный слой между словами из словаря целевой предметной области и словарем предварительно обученной языковой модели.

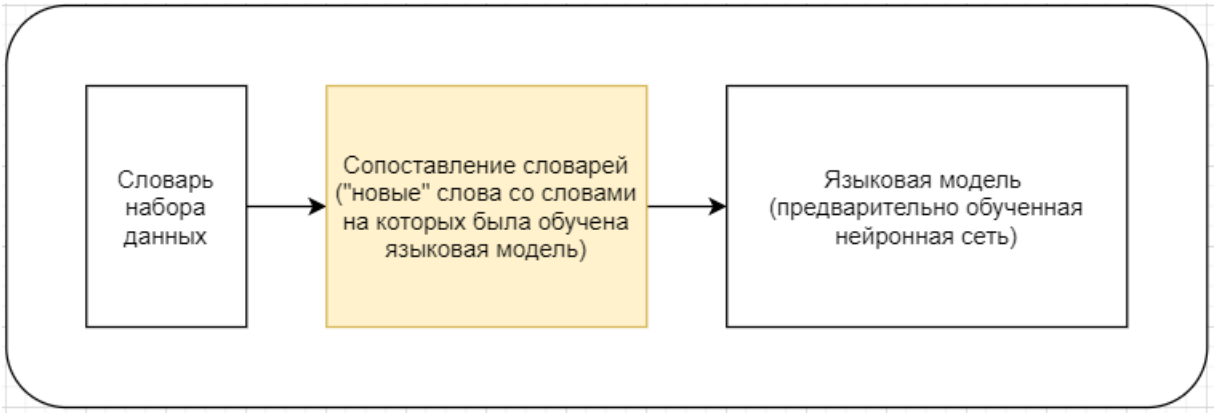


Рис. 2.14. Промежуточный слой для расширения словаря языковой модели

Этот слой состоит из двух частей: иерархического классификатора, который группирует слова из словаря языковой модели и алгоритма определения соответствия, который каждому слову из целевого словаря сопоставляет линейную комбинацию из слов словаря языковой модели

новое_слово = вес1 x слово1 + вес2 x слово2 + … + весN x словоN

Это необходимо для использования слов неизвестных для языковой модели, т.к. первый слой языковой модели на базе нейронной сети это слой, который трансформирует вектор соответствующий одному слову (one-hot) в компактный вектор из n-мерного векторного пространства векторов слов, где n-выбирается как глобальный параметр языковой модели.

Основная задача заключается в определении метода расчета весовых коэффициентов в линейной комбинации. Одним из возможных вариантов может быть выбор ближайшего соседа (наиболее близкое слово по смыслу, например “график” к “диаграмме”). Для определения расстояния между словами зачастую используется модель word2vec обученная на каком-либо корпусе текстов. В данной диссертационной работе используется модель word2vec обученная на корпусе Тайга [98]. Недостатком использования метода ближайшего соседа является то, что слово может иметь достаточно много близких по смыслу и при этом отличающихся друг от друга слов. Например, для “графика”: “диаграмма”, “схема”, “зависимость” и т.д.

Альтернативно, можно было бы использовать средний вектор для N ближайших соседей, но в этом случае “облако” синонимов может заслонить собой другие менее многочисленные кластеры. И в итоге их оттенки значения будут вытеснены. Например, как показано на Рис. 2.15.

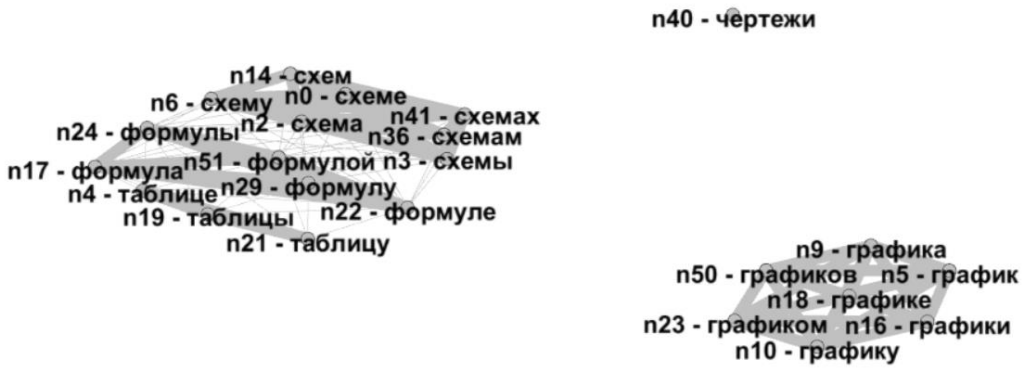


Рис. 2.15. Близкие слова к слову “график” согласно модели word2vec

Таким образом, очевидно, что расчет весовых коэффициентов должен учитывать разнообразие возможных смысловых оттенков слова и сглаживать перекосы вызванные большим количеством синонимов или словоформ.

Учитывая исследованные альтернативы, в данной диссертационной работе предлагается метод с использованием иерархического классификатора се-

мантически близких “известных” слов получаемого с использованием алгоритма ε-кластеризации нечеткого графа [34,125].

Для каждого слова строится иерархический классификатор, состоящий из слов семантически близких к данному. Определим $W_{lm}(i)$ - как множество слов включенных в список токенов языковой модели семантически близких к i-му “неизвестному” слову. Задача заключается в организации этих токенов в древовидную структуру, где листьям будут соответствовать отдельные слова $w_i \in W_{lm}(i)$, а промежуточным узлам будут соответствовать кластеры (подмножества слов) которые включают все слова из нижележащих узлов иерархии $w_{kj} \in C_k \subset W_{lm}(i)$. $|W_{lm}(i)|=N_i$. Эта задача может быть решена с помощью иерархической модификации ε-кластеризации нечеткого графа [95].

2.2.1. Построение иерархического классификатора

На первом шаге для множества семантически близких слов к целевому (новому, незнакомому) слову строится нечеткий граф с весами на ребрах равными семантической близости между словами. В данном методе требуется определить меру схожести для объектов (семантической близости), обозначим ее как μ. Эта мера может быть получена с использованием открытых моделей распределенных векторов слов и использованием евклидова или манхеттенского расстояния на них. В данной работе использована модель word2vec обученная на корпусе Тайга [98].

2.2.1.1. Определения и общие положения

Для удобства и краткости изложения в дальнейшем в работе термин “предложение” будет использоваться как синоним короткого текста, это не ограничивает общности для случаев двух или трех предложений в тексте.

Положим SL – список предложений, где ||SL|| = S, и S – количество предложений в наборе данных.

Первый шаг кластеризации – определение свойства, по которым будет проводиться кластеризация. Традиционно, существует два способа выбора свойств для текстов:

- Агрегирующая мера близости отражающая расстояния в одно- или многомерном пространстве между предложениями целиком.
- Мера близости для частей предложения, например слов.

В данной работе выбран второй способ. Свойствами текста являются слова предложения, для этого есть несколько причин:

- Количество различных слов, а тем более лемм слов, намного меньше, чем количество предложений в наборе данных.
- Использование слов в качестве свойств естественным образом порождает нечеткое соотнесение предложений (текстов) к получаемым кластерам.
- Расстояние, основанное на семантической связи слов, легче для понимания экспертами, чем синтетические метрики, таким образом, результаты данной кластеризации имеют больший потенциал для последующего использования в экспертном анализе.

2.2.1.2. Предварительная обработка текста

Выделение слов в тексте (токенизация, tokenization) и удаление стоп слов. Пусть STL(s) – список отдельных слов предложения $s \in SL$ получившийся в результате процедуры токенизации. Для программной реализации на языке python была использована библиотека NLTK [86].

Зачастую, на этапе предварительной обработки текстов проводится лемматизация токенов. В одном из экспериментов в данной дисссертационной работе показывается, что в случае использования языковой модели лемматизация ухудшает точность языковой модели. Поэтому в данной работе лемматизация не используется.

2.2.1.3. Построение нечеткого графа

Пусть $W_{SM}(w_1, w_2)$ – симметричная матрица расстояний между словами w_1, w_2, где $w_1, w_2 \in W$; $W_{SM}(w_1,w_2) = W_{SM}(w_2, w_1)$. Для программной реализации на языке python использован сервис RusVectores [67]. Этот сервис позволяет получить скалярную величину в диапазоне [0, 1], выражающую расстояние (семантическую близость) между двумя словами русского языка.

Пример:

W_{SM}(‘хоккей’, ‘футбол’) = 0.632;

W_{SM}(‘хоккей’, ‘автомобиль’) = 0.017;

Следуя определению, данному в работах Розенфилда [92], Раймонда Т. Йена и С.Й. Банга [90], нечеткий граф FG это пара <V, R>, где V - множество вершин и R – нечеткое отношение на V. Нечеткий граф называется симметричным, если R обладает свойствами рефлексивности и симметричности. В дальнейшем в данной статье будет рассматриваться симметричный нечеткий граф.

Матрица $W_{SM}(w_1, w_2)$ может быть трактована, как матрица нечеткого отношения R, таким образом может быть построен нечеткий граф FG, узлами которого являются слова из W.

Пример:

Для демонстрации результатов был использован набор данных из 20 000 предложений, содержащий 6500 лемм слов. На Рис. 2.16 представлен полученный граф. Толщина ребер соотносится с весом ребра, из соображений наглядности ребра с наименьшими весами были удалены.

Рис. 2.16. Нечеткий граф с 6500 русскими леммами слов и 500 000 ребер

Как уже было отмечено выше, нечеткий граф слов имеет только одну сильно связную компоненту. Таким образом, традиционные подходы поиска сообществ не позволяют получить содержательные результаты кластеризации подобного рода графов.

2.2.1.4. Иерархическая кластеризация нечеткого графа

Определение 1 (λ-срез графа). Пусть FG = <V, R> - симметричный нечеткий граф, и μ_R-функция принадлежности отношения R. Тогда G(λ) = <V, E(λ)> называется λ-срезом графа FG, где $E(\lambda) = \{e = v_i v_j \mid \mu_R(v_i, v_j) >= \lambda, v_i, v_j \in V\}$.

Утверждение 1. Если FG – симметричный нечеткий граф и G(λ) - λ-срез графа FG, тогда связные компоненты графа G(λ) – представляют собой четкую кластеризацию графа FG.

Доказательство данного утверждения достаточно очевидно и может быть найдено в [92] и [90].

Так как связные компоненты λ-среза графа попарно не пересекаются, то рекурсивно примененная процедура разбиения в результате даст иерархическую кластеризацию.

Ниже представлен алгоритм иерархической кластеризации:

Шаг 1. Инициализация параметров и переменных алгоритма

- λ_0 – начальное значение λ-среза графа ;
- $\Delta\lambda$ – шаг алгоритма ;
- λ – текущее значения уровня среза, вначале равно λ_0;
- λmax – максимальное значения уровня λ-среза;
- Kmax – максимальная глубина рекурсии;
- BouquetSize – предпочтительное количество потомков в иерархии у каждого узла. Этот параметр дает контроль над шириной иерархии.
- MinSubGraphWeight – минимальный вес подграфа для включения в иерархию в качестве отдельного узла. Этот параметр дает контроль над степенью детализации кластеризации, т.е. над количеством узлов в результирующей иерархии.
- FCG(λ) – текущий λ–срез графа FG, вначале совпадает с $G(\lambda_0)$ - λ_0-срез графа FG;
- FSG – текущий подграф, являющийся одной из связных компонент текущего λ –среза графа FCG(λ);
- HG – результат работы алгоритма – иерархия, как четкий граф. Вначале имеет один корневой узел;

Шаг 2. Шаг рекурсии начинается с проверки условия остановки рекурсии. Если одно из следующих условий верно, то переход на шаг 3:

- Достигнута максимальная глубина рекурсии Kmax.
- Достигнут максимальный уровень среза λmax.

Для каждой связной компоненты (FSG) текущего λ–среза (FCG) выполнять:

- Если вес подграфа

$$SGW = \sum_{v \in SG} WFN(v) \leq \text{MinSubGraphWeight}.$$

, то данный подграф не включается в иерархию в качестве отдельного узла, слова данного подграфа трактуются как шум, и шаг рекурсии заканчивается. Переход к пункту 3.

- Добавить новый дочерний узел соответствующий FSG в иерархию HG.
- Установить новое значение $\lambda = \lambda + \Delta\lambda$.
- Построить λ-срез графа FSG.
- Если не возможно получить более одной связной компоненты с весом SGW ≥ MinSubGraphWeight в построенном λ-срезе, то шаг рекурсии заканчивается. Переход к шагу 3.
- Если количество связных компонент в λ-срезе меньше BouquetSize значит можно провести еще один срез, поэтому происходит рекурсивное выполнение шага 2.

Шаг 3. Ожидать завершения всех ветвей рекурсии, после чего алгоритм останавливается.

В результате HG содержит иерархический классификатор набора лемм слов (W). Для каждой вершины v в HG в ходе алгоритма кластеризации определяется соответствующий подграф FSG(v) = <FSGV(v), FSGE(v)> графа FG.

Таким образом, при помощи алгоритма ε-кластеризации нечеткого графа строится иерархический классификатор, позволяющих выделить и классифицировать смысловые оттенки слова. Одним из основных достоинств использования подхода основанного на графе является легкость интерпретации результатов человеком. Классификатор может быть легко скорректирован экспертом предметной области, это позволит эксперту уточнить смысл определенных слов, сместить акценты в соотнесении со словарем языковой модели. На Рис. 2.17 слева представлен нечеткий граф для слова “график” и получившийся иерархический классификатор. На Рис. 2.18 можно увидеть детали полученного классификатора. Классификатор состоит из двух главных ветвей с тематиками

“аналитика” и “планирование”. Каждый уровень имеет номер который указывает на шаг иерархической кластеризации в ходе которого он был получен и говорит о том, что все слова в рамках кластера на этом уровне имеют взаимную близость больше чем ε-значение этого уровня. В данном примере эти уровни равны 0.3, 0.4 и 0.5 соответственно.

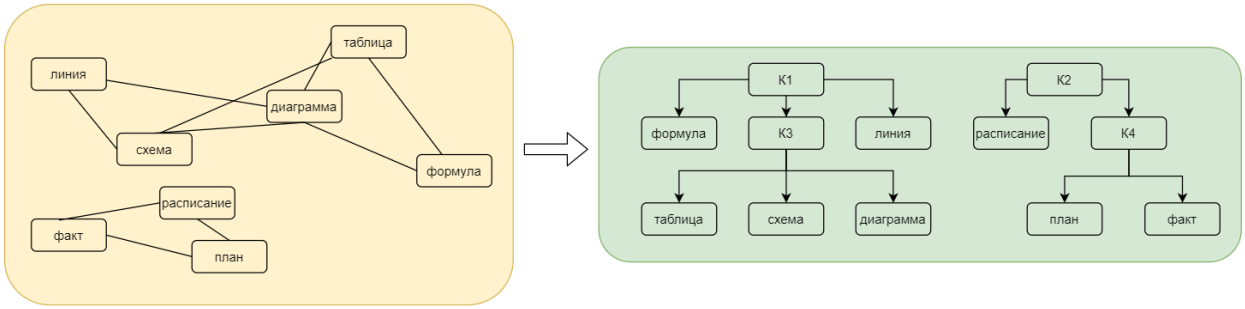


Рис. 2.17. Определение весовых коэффициентов линейной комбинации векторных представлений слов

2.2.1.1. Определение весовых коэффициентов линейной комбинации векторных представлений слов

Задача состоит в построении входного вектора языковой модели для каждого слова из целевого словаря. Вектор имеет N компонент, где N – число входных нейронов сети $N=|W_{lm}|$.

Для каждого слова w возможны два случая. Первый – слово уже включено в словарь языковой модели $w=w_i \in W_{lm}$, тогда соответствующий вектор будет иметь вид v = (0, 0, ..., 1, 0, ..., 0), где компонент с 1 имеет индекс i (one-hot вектор). Второй случай – слово отсутствует в словаре языковой модели $w \notin W_{lm}$. В этом случае для i-го слова строится иерархический классификатор, как было описано выше и затем строится вектор как линейная комбинация для вектора “неизвестного” слова по следующему алгоритму:

- Находим все слова в классификаторе по условию $W=\{w_j \mid \mu(w,w_j) > \theta=0{,}75, j \in [1, N]\}$
- В полученном иерархическом классификаторе отмечаются все узлы (слова и/или кластеры) являющиеся предками слов из множества W.

- Выбирается минимальный уровень L на котором находится хотя бы одно слово (не кластер) из множества W.
- Строится множество $WL = \{w_j,c_i \mid \mu(w,w_j) > \theta$ или $\mu(w,c_j) > \theta, w_j,c_i \in L\}$.
- Для каждого элемента множества WL определяем вес $\eta(w_{Li}) = \eta_p(w_{Li}) * (\mu(w, w_{Li}) / |\sum_{(wLj \in W\_L)} [\![\mu(w,w_{Lj})]\!]|)$, где η_p – вес родительского узла (в случае минимального уровня вес считается равным 1).
- Повторяются шаги 4 и 5 последовательно для всех уровней выше: $L=L+\delta$. Таким образом, будут определены веса для всех элементов множества W.

Приведенный алгоритм проиллюстрирован на Рис. 2.18. Вначале рассчитывается близость слова 'график' к другим словам. Наиболее близкие слова: диаграмма, 'расписание' and схема. Затем слой за слоем снизу вверх строится иерархия, до тех пор пока не останется только 2 кластера в уровне. Затем, сверху вниз начинается процесс определения весов для каждого узла, на основании дистанции до центров кластеров. Например, для расстояний (0.6 и 0.49, веса узлов рассчитаны как 0.55 и 0.45 соответственно. В самом конце, определяются значения для каждой компоненты входного вектора языковой модели для слова 'график' как (0, 0, 0.55, 0, 0.22, 0.23, 0, 0, ...).

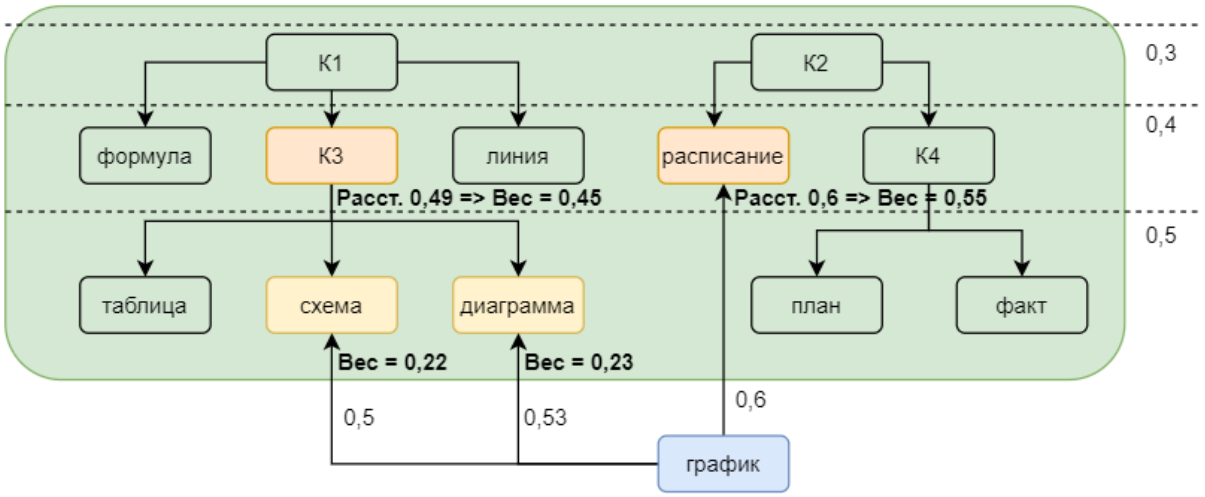


Рис. 2.18. Пример получения весовых коэффициентов для слова “график”

Таким образов все слова целевого словаря получают в соответствие векторы, которые могут быть использованы в предварительно обученной языковой модели.

При необходимости, даже слова, ранее имевшиеся в словаре языковой модели, могут быть заменены другими векторами. Это может быть полезно в случае, слов, чье смысловое значение кардинально отличается в выбранной предметной области от значения в рамках корпуса текстов на котором происходило обучение языковой модели.

2.3. Интерактивное получение обратной связи от пользователя и корректировка результатов кластеризации на ее основании

Придерживаясь идеи, что для исследователя наиболее простой и точной обратной связью будет критика полученных результатов кластеризации, в данной работе предполагается обратная связь двух видов:

1. “элемент X_i должен принадлежать кластеру C_j”
2. “элементу X_i не следует находиться в кластере C_j”.

Одновременно может быть получено произвольное количество таких ограничений, в частности легко задается ограничение “поменять элементы X_i и X_j местами” комбинацией двух ограничений первого вида. На Рис. 2.19 представлен пример формализованной обратной связи от эксперта в виде матрицы обратной связи. На пересечении строки соответствующей элементу (объекту) из набора данных и кластера к которому данный объект был отнесен с наибольшей степенью уверенности нейронной сети (максимальное значение в соответствующем компоненте выходного вектора) ставится обратная связь двух вышеописанных типов в виде “Включить” или “Исключить” соответственно.

	Кластер 1	**Кластер 2**				**Кластер K**
Элемент 1						Включить
Элемент 2		Включить				
Элемент N		Исключить				

Рис. 2.19. Пример матрицы обратной связи от эксперта

Из выше представленной обобщенной схемы методов кластеризации Рис. 2.11 видно, что обработка обратной связи на уровне единой целевой функции (функции потерь, штрафной функции, loss function) позволит одновременно строить представление объектов в соответствии с интенцией исследователя влияя на веса нейронной сети и корректировать ошибки кластеризации влияя на результат следующей итерации кластеризации (например, смещая центры кластеров).

Для представленного метода кластеризации не трудно заметить, что целевая функция направлена на то, чтобы q_{ij} было больше p_{ij}. Если посмотреть на частные производные для обновления весов нейронной сети и векторов центров кластеров

$$\frac{\partial L}{\partial z_i} = 2 \sum_j \left(1 + \left|\left|z_i - \mu_j\right|\right|^2\right)^{-1} * \left(p_{ij} - q_{ij}\right) * \left(z_i - \mu_j\right), \tag{2.7}$$

$$\frac{\partial L}{\partial \mu_j} = -2 \sum_i \left(1 + \left|\left|z_i - \mu_j\right|\right|^2\right)^{-1} * \left(p_{ij} - q_{ij}\right) * \left(z_i - \mu_j\right), \tag{2.8}$$

то можно понять, что в случае отрицательной разницы (p_{ij}-q_{ij}) будет происходить “выталкивание” элемента из кластера, несмотря на отсутствие штрафа со стороны целевой функции.

В данной диссертационной работе для учета обратной связи пользователя при корректировке весов нейронной сети методом обратного распро-

странения ошибки предлагается использовать следующие формулы для расчета градиентов целевой функции:

$$L = KL(P||Q) = \sum_i \sum_j t_{ij} p_{ij} \log \frac{p_{ij}}{q_{ij}}. \quad (2.9)$$

$$\frac{\partial L}{\partial z_i} = 2 \sum_j \left(1 + \left||z_i - \mu_j\right|\right|^2\right)^{-1} * |p_{ij} - q_{ij}| * t_{ij} * (z_i - \mu_j), \quad (2.10)$$

$$\frac{\partial L}{\partial \mu_j} = -2 \sum_i \left(1 + \left||z_i - \mu_j\right|\right|^2\right)^{-1} * |p_{ij} - q_{ij}| * t_{ij} * (z_i - \mu_j), \quad (2.11)$$

Где T = $\{t_{ij}\}$ – матрица множителей обратной связи (подсказок пользователя, tips), в которой:

$$t_{ij} = \begin{cases} > 0, & \text{для включения элемента } i \text{ в кластер } j \\ < 0, & \text{для исключения элемента } i \text{ из кластера } j \\ 0, & \text{иначе.} \end{cases}$$

Матрица T получается тривиальным образом из матрицы обратной связи от эксперта, представленной на Рис. 2.19. На рисунке Рис. 2.20 представлен пример с результатом преобразования матрицы обратной связи от эксперта в матрицу множителей для целевой функции нейронной сети.

	Кластер 1	**Кластер 2**				**Кластер K**
Элемент 1	1	1	1	1	1	**100**
Элемент 2	1	**100**	1	1	1	1
	1	1	1	1	1	1
	1	1	1	1	1	1
	1	1	1	1	1	1
	1	1	1	1	1	1
	1	1	1	1	1	1
Элемент N	1	**-1000**	1	1	1	1

Рис. 2.20. Матрица множителей обратной связи для целевой функции

Абсолютное значение t_{ij} определяет скорость с которой элементы и центры кластера будут стремиться друг к другу или отталкиваться друг от друга. Также на эту скорость влияет выставленный уровень обучения (learning rate) у нейронной сети. В данной работе в экспериментах использовалось значение 100 для включения элемента в кластер и -1000 для исключения элемента из кластера. Эти значения были определены эмпирически, т.к. эксперименты показали, что для случая выталкивания элемента из кластера имеет смысл использовать большие абсолютные величины, чем при притяжении.

Описанный выше подход позволяет достаточно эффективно решать задачу кластеризации, при этом в случае таких многомерных объектов как текст, пользователю необходимо иметь возможность оказывать влияние на ход процесса кластеризации с целью выявления скрытой или явной интенции.

Одним из основных достоинств предлагаемого метода кластеризации является то, что добавляемые ограничения не являются жесткими, не приводят к необходимости решать системы уравнений, которые могут потенциально оказаться несовместными. Любые ограничения пользователя, независимо от их внутренних противоречий, будут учтены в штрафах со стороны целевой функции.

2.4. Выводы по главе

В данной главе представлен метод интерактивной кластеризации с обратной связью на базе современных методов кластеризации. Используемая обобщенная схема построения алгоритма кластеризации и предложенная архитектура нейронной сети позволяют сочетать преимущества современных языковых моделей с высокими показателями точности и перплексии и наиболее эффективных на сегодняшний день универсальных алгоритмов кластеризации. Блоки языковой модели и кластеризации в предложенной архитектуре искусственной нейронной сети со временем могут быть легко заменены на блоки более современных и эффективных языковых моделей и методов кластеризации без необходимости изменения предлагаемого алгоритма.

Разработанный метод позволяет эксперту проводить кластеризацию наборов коротких текстов, выдавая обратную связь по результатам каждого шага интерактивной кластеризации. Процедура сбора обратной связи не предполагает наличия у эксперта специальных знаний о работе нейронной сети и собирается в виде человекочитаемой матрицы обратной связи. Такой подход обладает преимуществами по сравнению с методами кластеризации требующими корректировки метапараметров алгоритма не связанных напрямую с результатами кластеризации. В таких методах эксперт взаимодействует с алгоритмом как с моделью “черного ящика”, что снижает эффективность человеко-машинного взаимодействия.

Также важным преимуществом предложенного метода является возможность осуществлять кластеризацию наборов данных относящихся к различным языковым доменам не совпадающим с доменом, на котором производилось обучение языковой модели, за счет предложенного метода расширения словаря языковой модели. Это свойство позволяет использовать предложенный алгоритм в узко специализированных доменах, а также в доменах не позволяющих получить полноценный корпус текстов для самостоятельно обучения языковой модели.

ГЛАВА 3. Разработка и реализация алгоритма в системе поддержки принятия решений

Апробация и внедрение моделей и методов представленных во второй главе, а также оценка производительности и эффективности проводилась в рамках Федеральной Информационной Системы Стратегического Планирования (ФИС СП) Государственной Автоматизированной Системы “Управление” (ГАС “Управление”). Для этого требовалось разработать алгоритм и реализующий его программный модуль, который бы позволил осуществлять интерактивную кластеризацию наборов коротких текстов на русском языке без специальной предварительной обработки.

3.1. ГАС “Управление”

ГАС “Управление” была создана на основании Положения о государственной автоматизированной информационной системе "Управление", утвержденном Постановлением Правительства Российской Федерации от 25 декабря 2009 г. № 1088 и функционирует в рамках концепции развития государственной автоматизированной информационной системы «Управление» одобренной протоколом подкомиссии по использованию информационных технологий при предоставлении государственных и муниципальных услуг Правительственной комиссии по внедрению информационных технологий в деятельность государственных органов и органов местного самоуправления от 2 октября 2012 №8. Система состоит из открытой и закрытой частей. Закрытая часть предназначена для внутреннего использования различными федеральными и региональными органами исполнительной власти (ФОИВ, РОИВ), а также органами местного самоуправления (ОМСУ). Открытая часть портала ГАС “Управление” доступна по адресу gasu.gov.ru для всех юридических и физических лиц и представлена на Рис. 3.21.

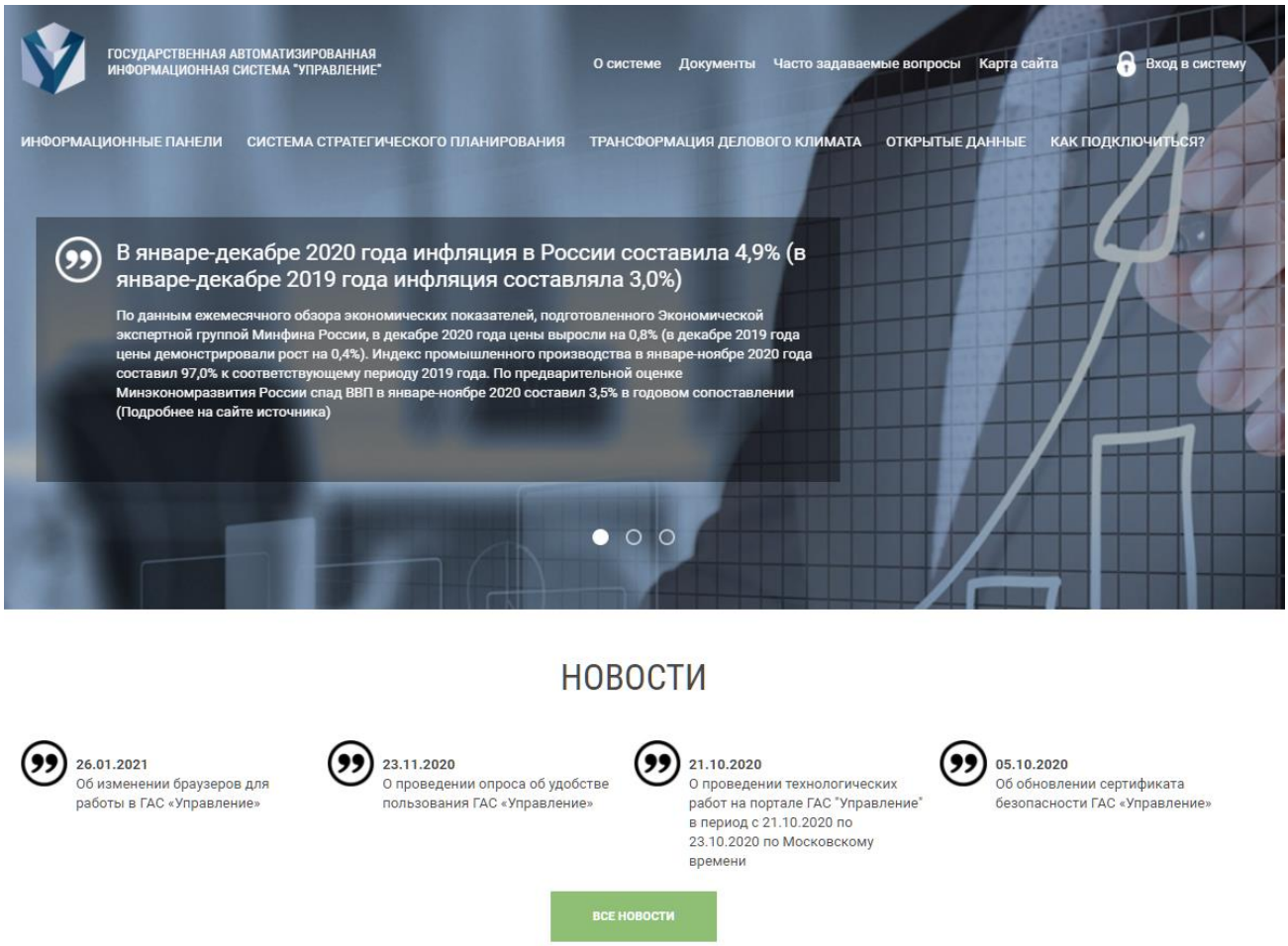


Рис. 3.21. Пользовательский интерфейс ГАС “Управление”.

Основными функциями системы являются: сбор данных показателей по различным тематикам и аспектам функционирования Российской Федерации (РФ) и предоставления аналитики на основании собранных данных. На Рис. 3.22 представлены разделы аналитической части открытого портала ГАС “Управление”, среди них, например, представлена аналитика показателей бюджетной системы РФ и аналитика по мониторингу контрольно-надзорной деятельности.

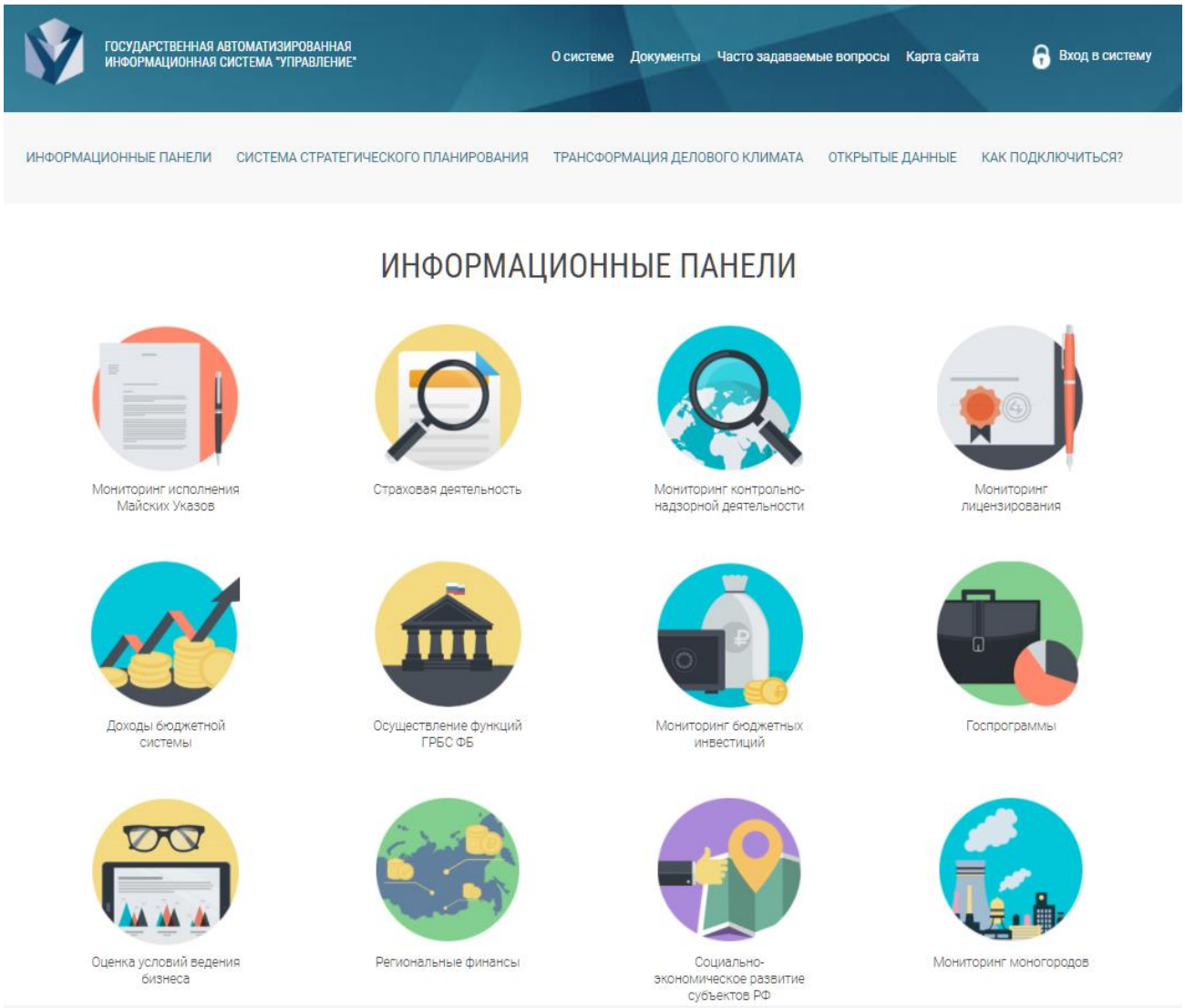


Рис. 3.22. Разделы аналитической части открытого портала ГАС “Управление”.

На Рис. 3.23 представлен пример интерактивной аналитической панели по региональным финансам на примере Ульяновской области.

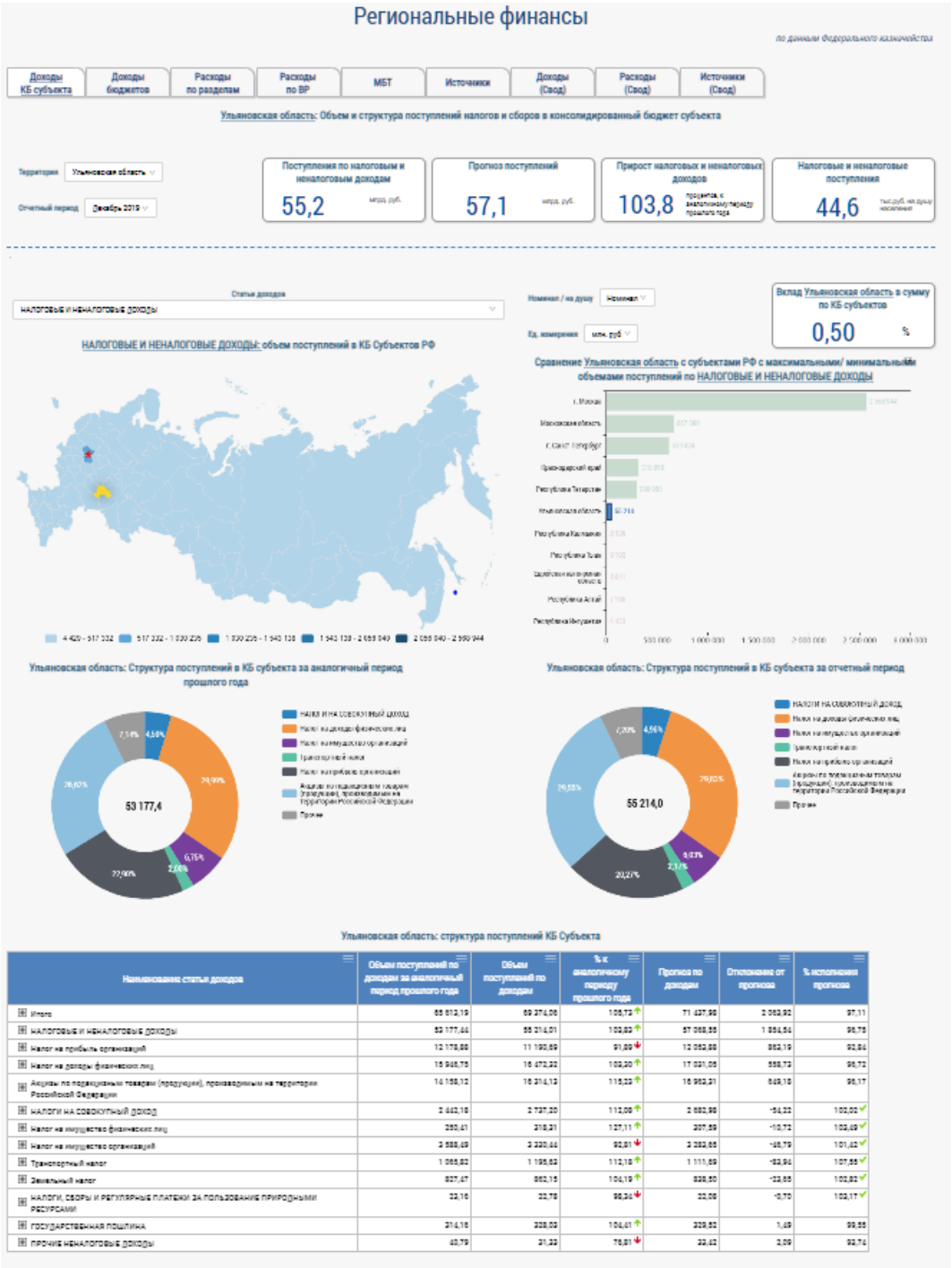


Рис. 3.23. Аналитика региональных финансов в ГАС “Управление”.

ГАС “Управление” поддерживает интеграцию через Систему Межведомственного Электронного Взаимодействия (СМЭВ), а также интегрируется с рядом государственных систем. На Рис. 3.24 представлена техническая архитектура ГАС “Управление”. Реализация программного модуля описанного ниже в

пункте 3.4. проводилась на уровне блоков “Единого хранилища данных” и “Подсистемы ведения реестров, справочников и классификаторов”.

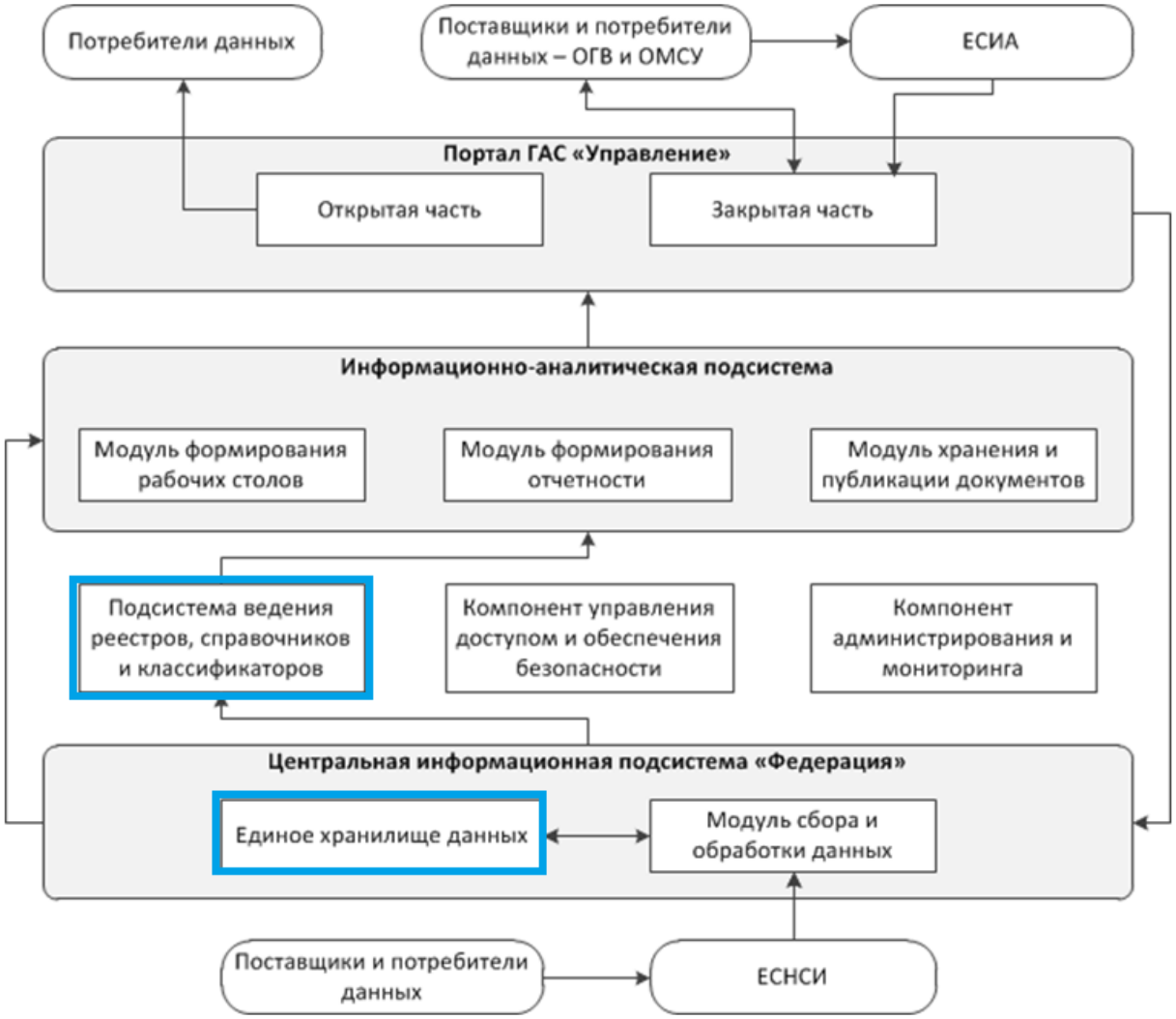


Рис. 3.24. Техническая архитектура ГАС “Управление”.

Помимо обширной интерактивной аналитики, в системе доступно большое количество регламентных отчетов и справок. Данный функционал обеспечивает поддержку принятия управленческих решений на всех уровнях исполнительной власти: федеральном, региональном и муниципальном.

3.2. ФИС Стратегического Планирования

Федеральная Информационная Система Стратегического Планирования Российской Федерации закрепленная федеральным законом № 172-ФЗ от

28.07.2014 г была создана и функционирует в рамках ГАС “Управление”. В установочных документах системы отмечается, что “… важнейшим фактором обеспечения конкурентоспособности российской экономики в современных условиях является наличие эффективно функционирующей системы государственного стратегического управления. <…>. На сегодняшний день стратегический характер приобрела деятельность, направленная на преодоления кризисных явлений в экономике. В этом направлении в соответствии с поручением Правительства Российской Федерации ведётся активная работа по мониторингу мер, направленных на оздоровление экономики. <…>. Эффективное функционирование системы стратегического управления особенно актуально в период финансового кризиса и дефицита Федерального бюджета. Формирование системы государственного стратегического управления позволит изменить подход к среднесрочному прогнозированию, увязать его с прогнозированием долгосрочных тенденций развития, обеспечить координацию разработки, реализации долгосрочных стратегий и программ развития Российской Федерации в целом, а также отдельных регионов и секторов экономики, их взаимную увязку по целям, срокам и мероприятиям. Кроме того, данная система предполагает развитие механизмов «управления по результатам», обеспечивающих четкую взаимосвязь между результатами деятельности органов государственной власти и бюджетными средствами, выделенными на их достижение” [121].

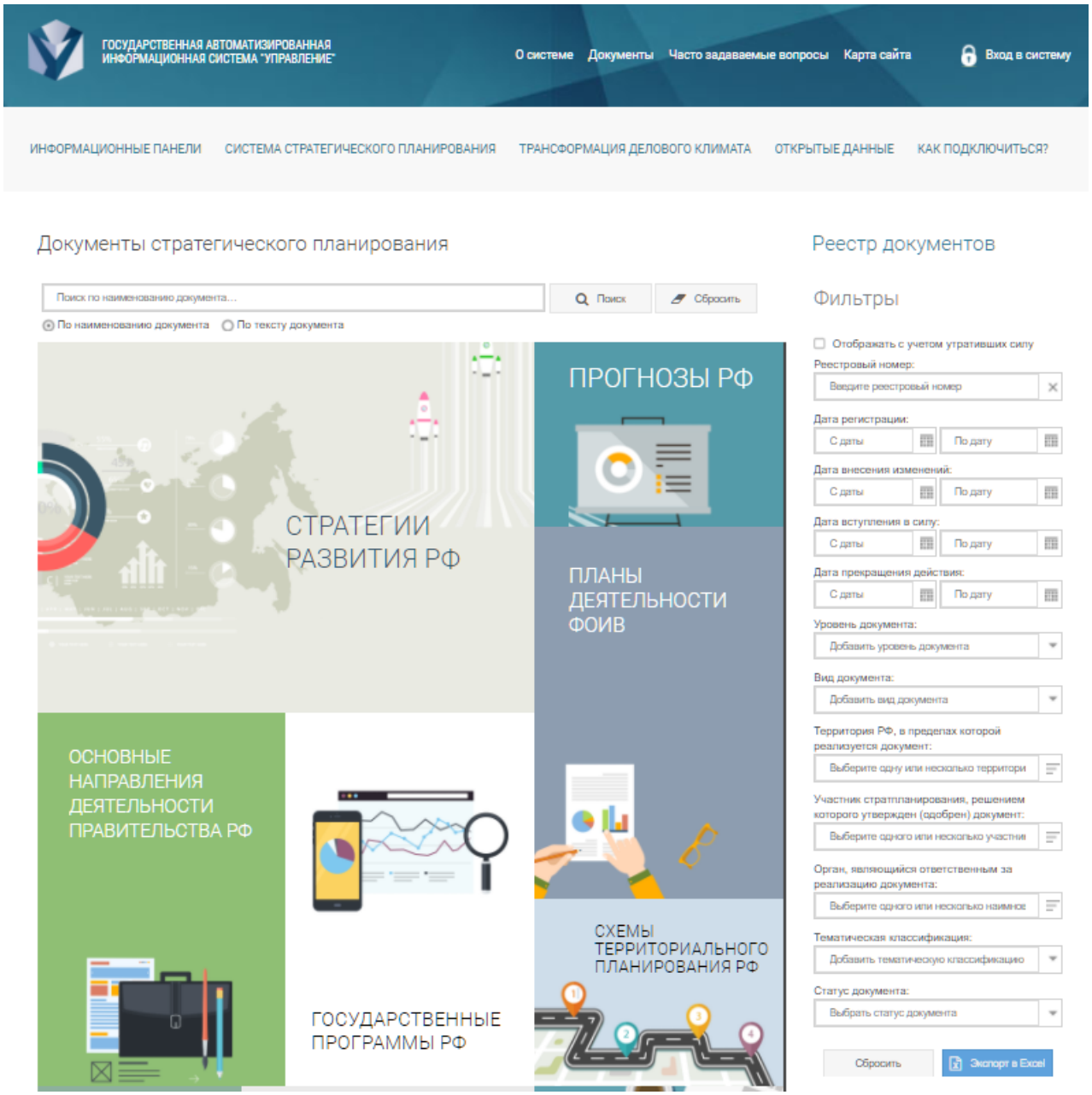


Рис. 3.25. Интерфейс ФИС Стратегического Планирования.

На Рис. 3.5 представлен интерфейс открытой части ФИС СП. В рамках открытой части доступен реестр открытых документов системы СП и поиск по основным атрибутам. В рамках данной диссертационной работы проводилось исследование ключевых показателей эффективности (КПЭ) ФИС СП, доступных в закрытой части системы. На момент написания ФИС СП содержит более 600 000 показателей в действующих документах, и более 1,2 миллиона показателей с учетом документов прекративших свое действие.

На момент начала исследования, в поступающих документах участники системы стратегического планирования самостоятельно проставляли классификацию ключевых показателей эффективности из краткого классификатора ЕМИСС (Единой межведомственной информационно-статистической системы) содержащего 11 классов (тематик). В этом процессе имелось два основных недостатка: малое число классов, не отражающее всего многообразия вводимых показателей и не пригодное для дальнейшего анализа, а также большое число случаев ошибочной классификации, вызванное тем, что классификацию проводят не эксперты. Дополнительным недостатком процесса является длительность этапа проверки и корректировки ошибочной классификации показателей.

Предложенный в данной работе метод нечеткой интерактивной кластеризации коротких текстов позволил усовершенствовать архитектуру ФИС СП. На рис. 7 представлена архитектура до и после проведенных изменений.

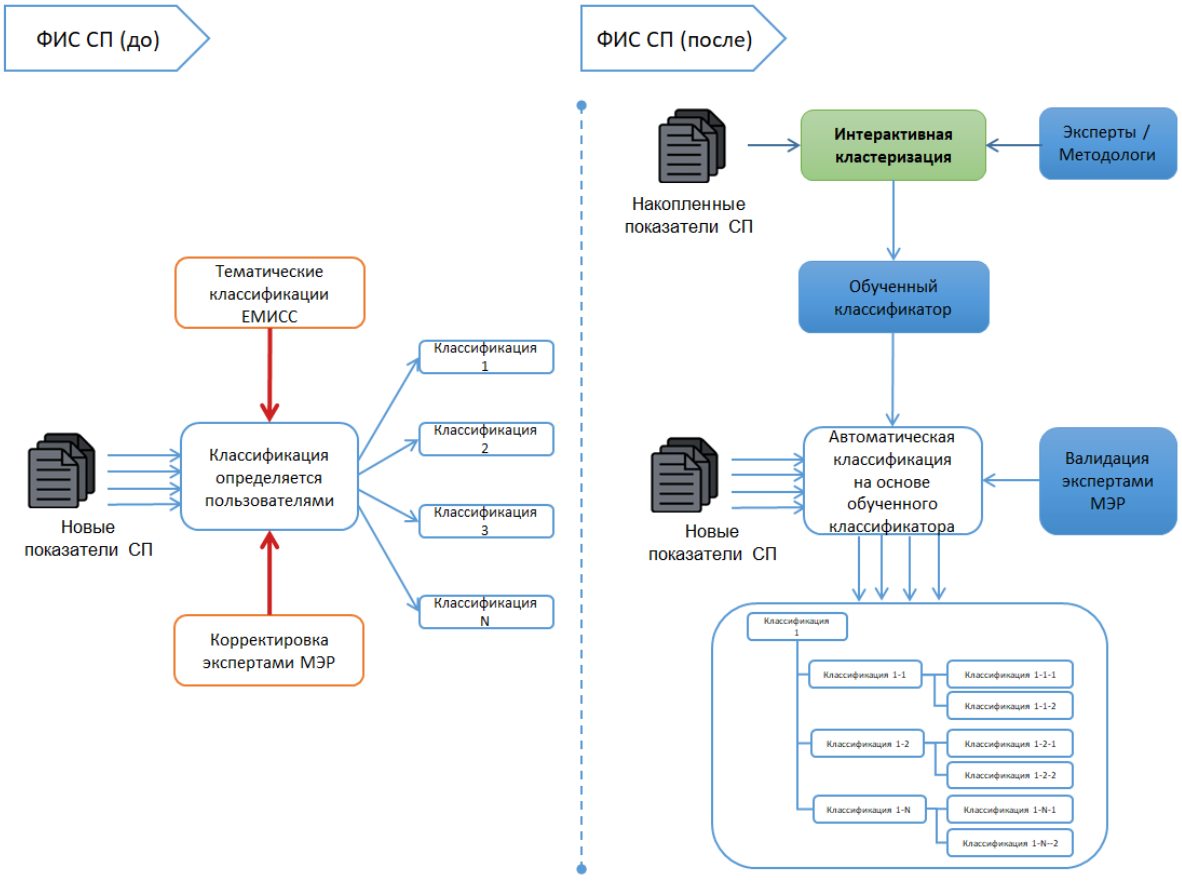


Рис. 3.26. Архитектура ФИС СП до и после проведенных изменений.

В результате проведенной интерактивной кластеризации экспертами методологами был обучен классификатор, который используется для автоматизации классификации показателей стратегического планирования и содержит несколько сотен классов. Помимо более точного классификатора снизилась нагрузка на экспертов проводящих проверку входящих документов благодаря указываемой степени “уверенности” классификатора при соотнесении показателя с тем или иным классом. Оценки эффективности применения предложенного метода представлены в четвертой главе.

3.3. Алгоритм интерактивной кластеризации коротких текстов

Для проведения процедуры кластеризации в результате которой получен классификатор показателей СП разработан алгоритм кластеризации набора коротких текстов, использующий методы описанные в Главе 2. Алгоритм состоит из следующих шагов:

- Предобработка текстов: исправление орфографии, токенизация.
- Расширение словаря языковой модели
- Тонкая настройка языковой модели (дополнительное обучение)
- Инициализация слоев нейронной сети блока кластеризации (обучение автоэнкодера)
- Инициализация центров кластеров (с использованием k-means)
- Первичная кластеризация (синхронное обучение нейронной сети кластеризации коротких текстов и определение центров кластеров)
- Шаги интерактивной кластеризации (цикл до достижения приемлемого для эксперта качества кластеризации):
    - Анализ результатов кластеризации экспертом
    - Получение матрицы обратной связи от эксперта
    - Корректировка весов нейронной сети и корректировка центров кластеров на основании матрицы множителей обратной связи

Следует отметить, что основная вычислительная сложность приходится на первые три шага данного алгоритма. Для исправления орфографии необходим внешний специализированный сервис для языка набора данных. Такие сервисы помимо исправления типичных орфографических ошибок, могут исправлять ошибки с учетом контекста, что положительно сказывается на качестве кластеризации. Шаг тонкой настройки нейронной сети занимает значительное время (часы), т.к. современные языковые модели содержат миллионы параметров и большое количество слоев. Для экономии вычислительных ресурсов может обучаться только определенное количество последних слоев языковой модели, но это, как правило, ухудшает качество языковой модели, снижая точность и перплексию.

Шаги интерактивной кластеризации занимают всего несколько эпох, т.к. условием прекращения обучения является достижение порога минимального количества перемещенных элементов из кластера в кластер. Данная операция занимает время от секунд до нескольких минут, таким образом, интерактивная работа с экспертом может вестись в режиме реального времени.

Предлагаемый алгоритм реализован в виде программного модуля в рамках системы ГАС “Управления” описанной разделе 3.1

3.4. Архитектура программного модуля

С учетом особенностей предлагаемого метода и предполагаемым условиям эксплуатации в рамках ГАС “Управления”, программный модуль должен удовлетворять следующим требованиям:

- пользовательский интерфейс программного комплекса должен поддерживать взаимодействие с пользователем в интерактивном режиме, запоминать введенные пользователем ограничения и учитывать их в последующих итерациях кластеризации;
- программный модуль должен быть реализован в трехзвенной архитектуре, для возможности размещения серверной части на производительных серверах и предоставления клиентского доступа к приложению через браузер ПК;

- программный модуль должен быть реализован на open-source технологиях и работать под ОС Windows, CentOs, Linux Astra, т.е. соответствовать требованиям по импортозамещению;

С учетом заданных ограничений для реализации программного модуля выбран язык Python, позволяющий как реализовать алгоритмическую часть с помощью фреймворка FastAI [51], так и обеспечить потенциальную реализуемость клиент-серверной часть обеспечивающей взаимодействие с пользователем с помощью фреймворка Flask. Интерпретатор данного языка встроен во все ОС семейства Linux, а для Windows имеются удобные IDE для разработчиков. На Рис. 3.27 представлена архитектура разработанного комплекса.

В рамках работы над диссертацией разработан и полностью реализован блок Машинного обучения, позволивший провести все необходимые эксперименты и решить поставленные задачи. Также для проверки применимости результатов данной диссертационной работы в рамках тиражируемого программного продукта “Планета.Аналитика” потребовалось спроектировать блоки Rest-сервисов и пользовательских интерфейсов.

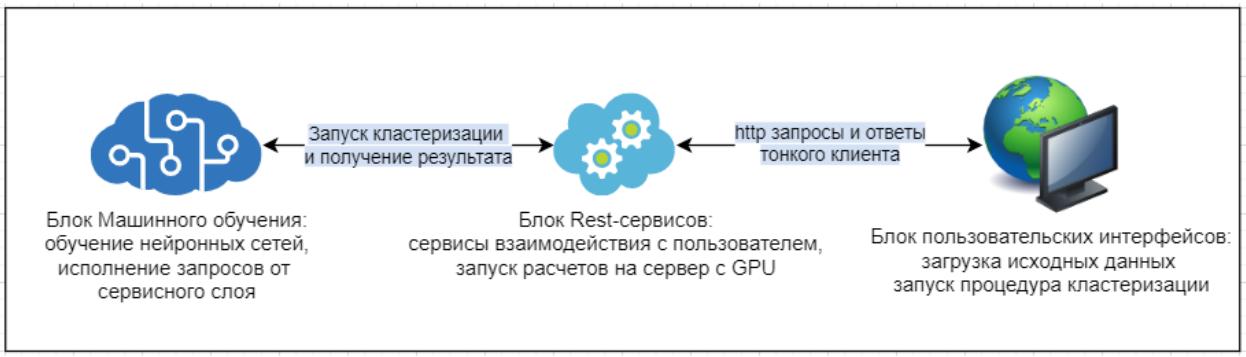


Рис. 3.27. Трехзвенная архитектура программного комплекса для интерактивной кластеризации коротких текстов.

3.4.1. Блок Машинного обучения

Блок Машинного обучения, отвечающий за предварительную обработку текстов, обучение языковой модели и обучения нейронной сети для кластеризации коротких текстов, имеет структуру представленную на Рис. 3.28. Он состоит из двух основных крупных блоков отвечающих за первый и второй этапы

представленного метода соответственно. В рамках каждого блока обучается нейронная сеть, которая впоследствии используется блоком Rest-сервисов для решения задачи кластеризации. В блоке предварительной обработки происходит исправление опечаток при помощи открытого сервиса от Яндекс, удаление знаков препинания, цифр и прочих символов не пригодных для обработки в языковой модели.

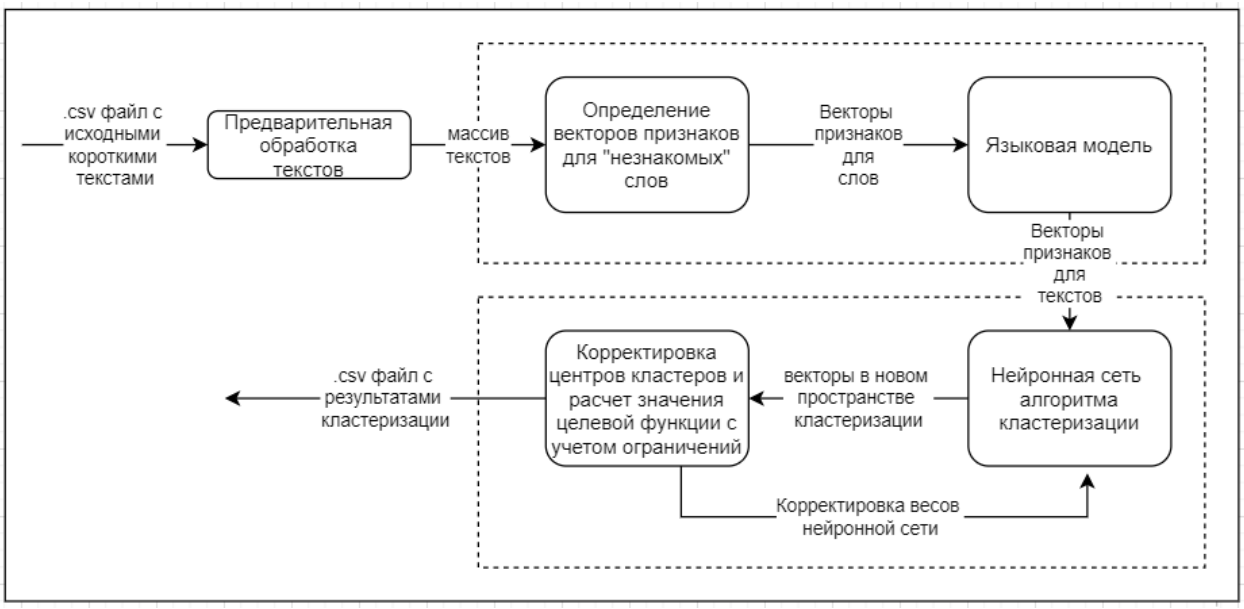


Рис. 3.28. Функциональная структура блока машинного обучения.

Для этапа обучения нейронных сетей и контроля качества обучения использовался инструмент Jupyter Notebook, позволяющий пошагово выполнять команды с мгновенным выводом результата команды. Данный инструмент является стандартом среди специалистов по обработке данных. Каждому блоку из Рис. 3.28 соответствует файл в Jupyter Notebook. На Рис. 3.29 представлен файл с модулем построения векторов признаков для “неизвестных” слов.

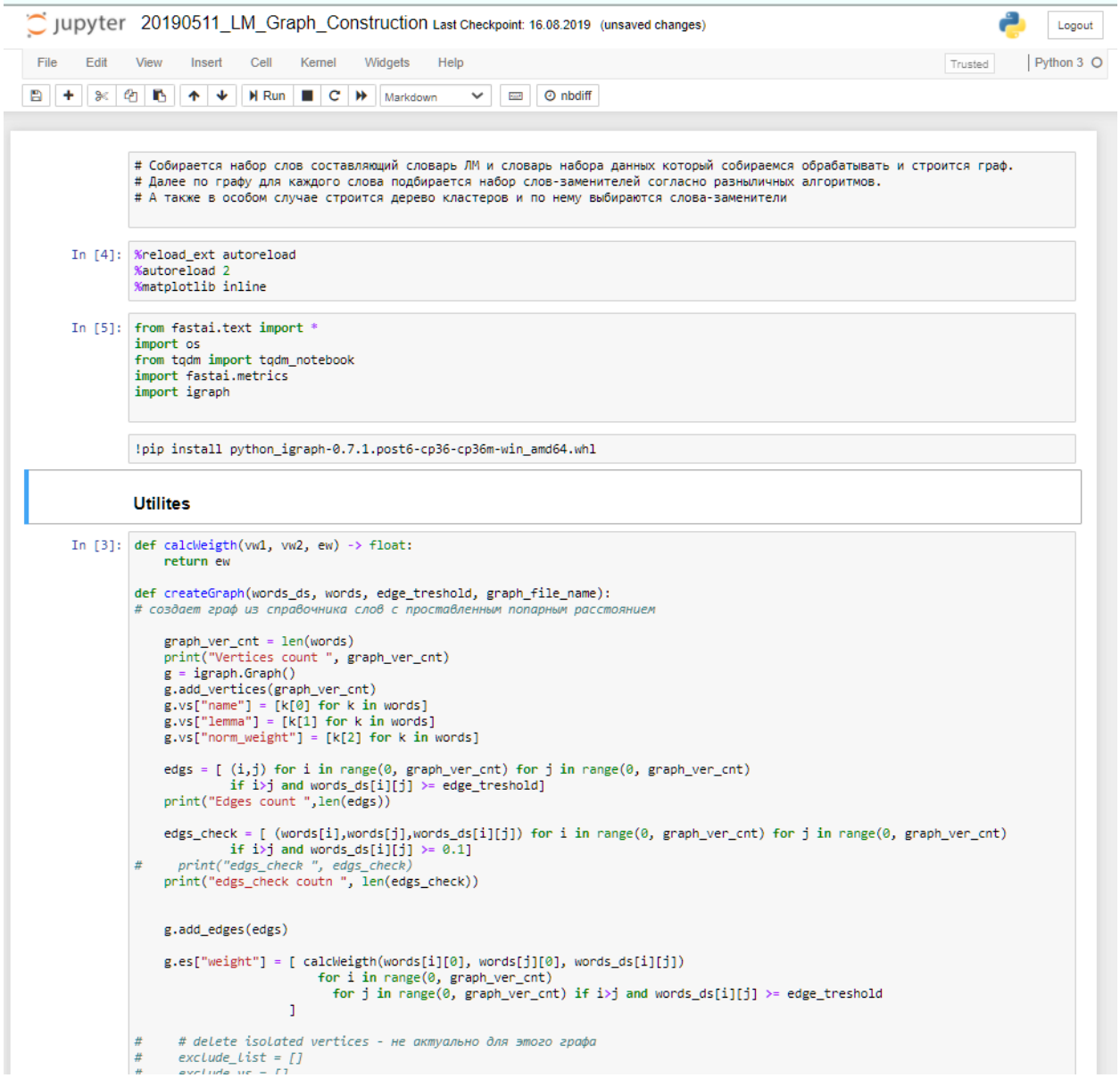


Рис. 3.29. Модуль построения векторов признаков для “неизвестных” слов в Jupyter Notebook.

Для имплементации нейронной сети реализующей языковую модель выбран фреймворк FastAI, на котором доступна модель ULMFit являющаяся одной из самых современных языковых моделей и самой распространенной для языков славянской группы. Данный фреймворк представляет собой набор классов над популярным фреймворком PyTorch, являющимся стандартом для реализации современных нейронных сетей наряду с Keras. При этом PyTorch обладает более гибким API (application programming interface).

Имплементация нейронной сети для кластеризации проводилась при помощи фреймворка MXNet от Apache. При помощи данного фреймворка реализован метод кластеризации DEC взятый за основу в данном исследовании и доработанный для возможности применения в интерактивной кластеризации. Данный фреймворк также функционирует поверх фреймворка PyTorch.

3.4.2. Блок Rest-сервисов

Проект блока Rest-сервисов, реализующий сервисный слой взаимодействия с пользователем и запускающий итерации алгоритма кластеризации, состоит из одного сервиса со следующими методами:

- uploadShortTextDataset - загрузка csv-файла с исходными данными;
- startPreProcessing - запуск процедуры предварительной обработки коротких текстов в блоке машинного обучения;
- startVocabularyUpdate - запуск в блоке машинного обучения процедуры обработки “неизвестных” (для используемой языковой модели) слов из входного набора данных, возвращение числа слов для которых удалось построить векторы признаков на основании имеющегося словаря языковой модели;
- startLMTuning - запуск процедуры дополнительного обучения (тонкой настройки) нейронной сети реализующей языковую модель, возвращение результатов по значению целевой функции и точности работы языковой модели;
- downloadShortTextVec - передача csv-файла с векторами признаков для входного набора данных для скачивания из браузера;
- uploadShortTextVec - загрузка csv-файла с векторами признаков для входного набора данных коротких текстов;
- startClusteringIteration - запуск алгоритма кластеризации с передачей введенных ограничений, возвращение значения функции потерь, индексов качества кластеризации и массива списка кластеров с основными характеристиками;

- getClusterName - получение имени кластера по идентификатору короткого текста;
- getClusterContent - получение списка элементов по идентификатору кластера и набору фильтров (определенный процент случайных текстов, наиболее близкие к центру кластера, наиболее далекие от центра кластера).

3.4.3. Блок пользовательских интерфейсов

Дизайн блока пользовательских интерфейсов, отвечающий за интерактивное взаимодействие с пользователем и визуализацию результатов работы предложенного метода, состоит из трех экранных форм.

Первая экранная форма представлена на Рис. 3.30 и содержит следующие основные управляющие и визуальные элементы:

- Кнопка “Загрузить файл” предназначена для загрузки csv-файла с исходным набором данных
- Кнопка “Предобработка” предназначена для запуска процедуры предварительной обработки исходного набора данных (исправление опечаток, приведение к формату пригодному для обработки языковой моделью).
- Рядом с каждой кнопкой запускающей длительную процедуру на сервере располагается элемент “progress bar”, отражающий динамику работы серверной процедуры.
- Кнопка “Подготовка словаря” предназначена для запуска процедуры построения векторов признаков для слов не входящих в словарь языковой модели.
- Визуальный элемент вывода числа слов добавленных в словарь языковой модели отображает значение возвращенное процедурой после нажатия на кнопку “Подготовка словаря”.
- Кнопка “Открыть папку” открывает серверную папку с файлами для программы Gephi, содержащими граф, построенный для каждого из слов добавляемого в словарь языковой модели.

- Поле для ввода размера шага обучения нейронной сети позволяет влиять на скорость обучения в рамках одной эпохи.
- Поле для ввода количества эпох обучения нейронной сети позволяет влиять на длительность обучения.
- Поле для ввода количества слоев для обучения позволяет определить количество последних слоев нейронной сети, для которых будут применяться корректировки весовых коэффициентов.
- Кнопка “До-обучение” позволяет запустить процедуру тонкой настройки нейронной сети под особенности предметной области и лексики исходного набора данных.
- Визуальные элементы вывода значения целевой функции и рассчитанной точности работы языковой модели на валидационной выборке отображают значения, возвращаемые в результате работы процедуры запускаемой кнопкой “До-обучение”.
- Кнопка “Скачать результат” позволяет получить csv-файл с векторами признаков для каждого из текстов исходного набора. Данный файл является исходным файлом для последующего этапа кластеризации.

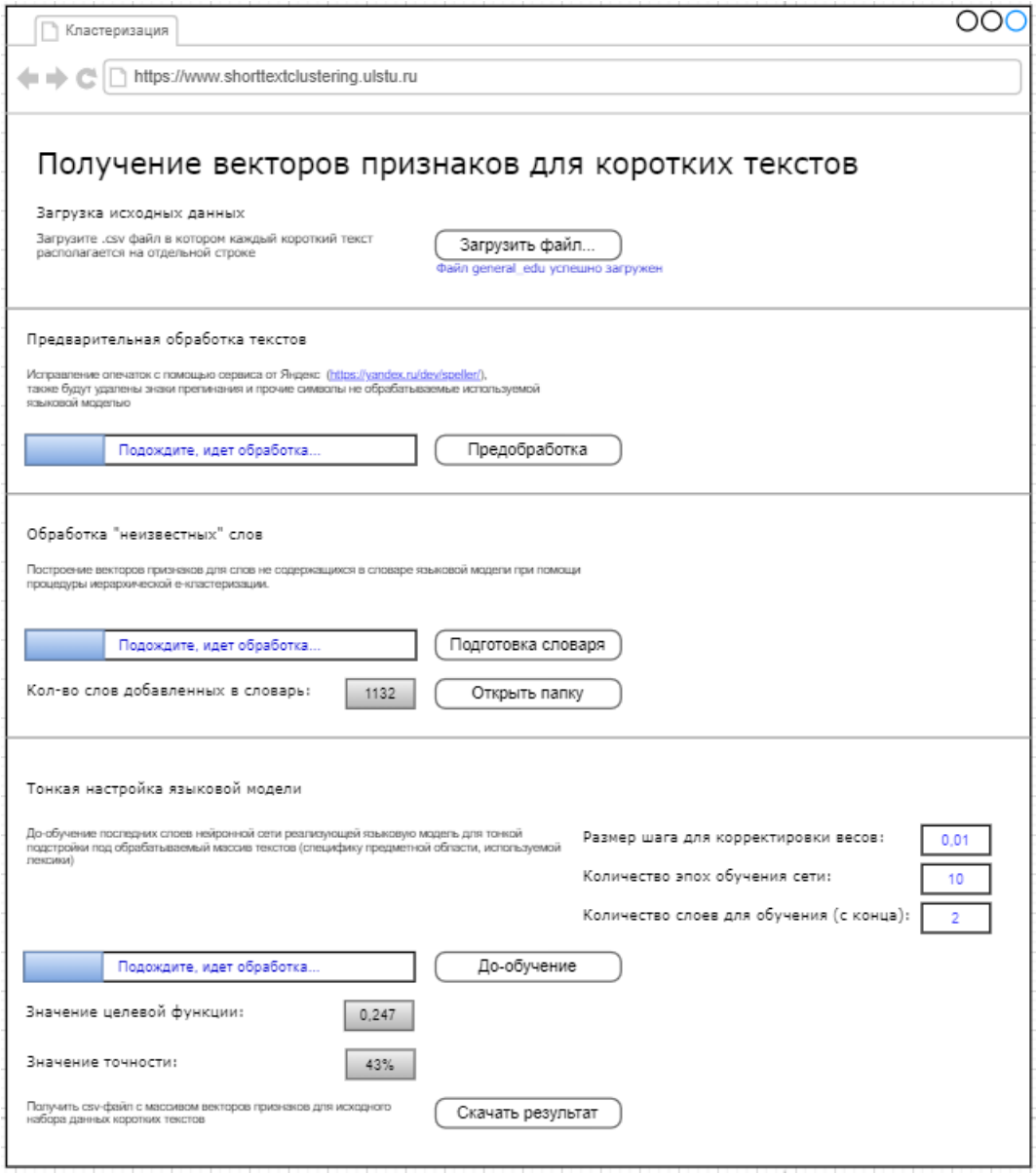


Рис. 3.30. Экранная форма получения векторов признаков для коротких текстов.

Вторая экранная форма представлена на Рис. 3.31 и содержит следующие основные управляющие и визуальные элементы:

- Кнопка “Загрузить файл” предназначена для загрузки csv-файла с исходным набором данных содержащим векторы признаков для коротких текстов.
- Поле для ввода числа кластеров задает гипер-параметр для алгоритма кластеризации.

- Таблица “Ограничения” с помощью кнопок “+” и “-” позволяет задать ограничения для итерации алгоритма кластеризации в формате: (объект; кластер; включить/исключить). Также после выполнения итерации кластеризации в колонке “Новый кластер" будет выведено наименование кластера, в который вошел данный объект (текст).
- Кнопка “Кластеризация” запускает процедуру кластеризации с заданным числом кластеров и указанным набором ограничений.
- В результате работы процедуры кластеризации будут заполнены визуальные элементы со значением целевой функции и индексом качества кластеризации.
- Таблица “Результаты кластеризации” также заполняется после выполнения процедуры кластеризации и содержит информацию о полученных кластерах, числе элементов в каждом, плотности, минимальном и максимально расстоянии элементов от центра кластера, а также позволяет вручную пользователю задать имя кластера. Введенное имя будет запомнено и на последующих итерациях будет применяться к кластеру, имеющему наиболее общее число элементов с кластером из прошлой итерации.

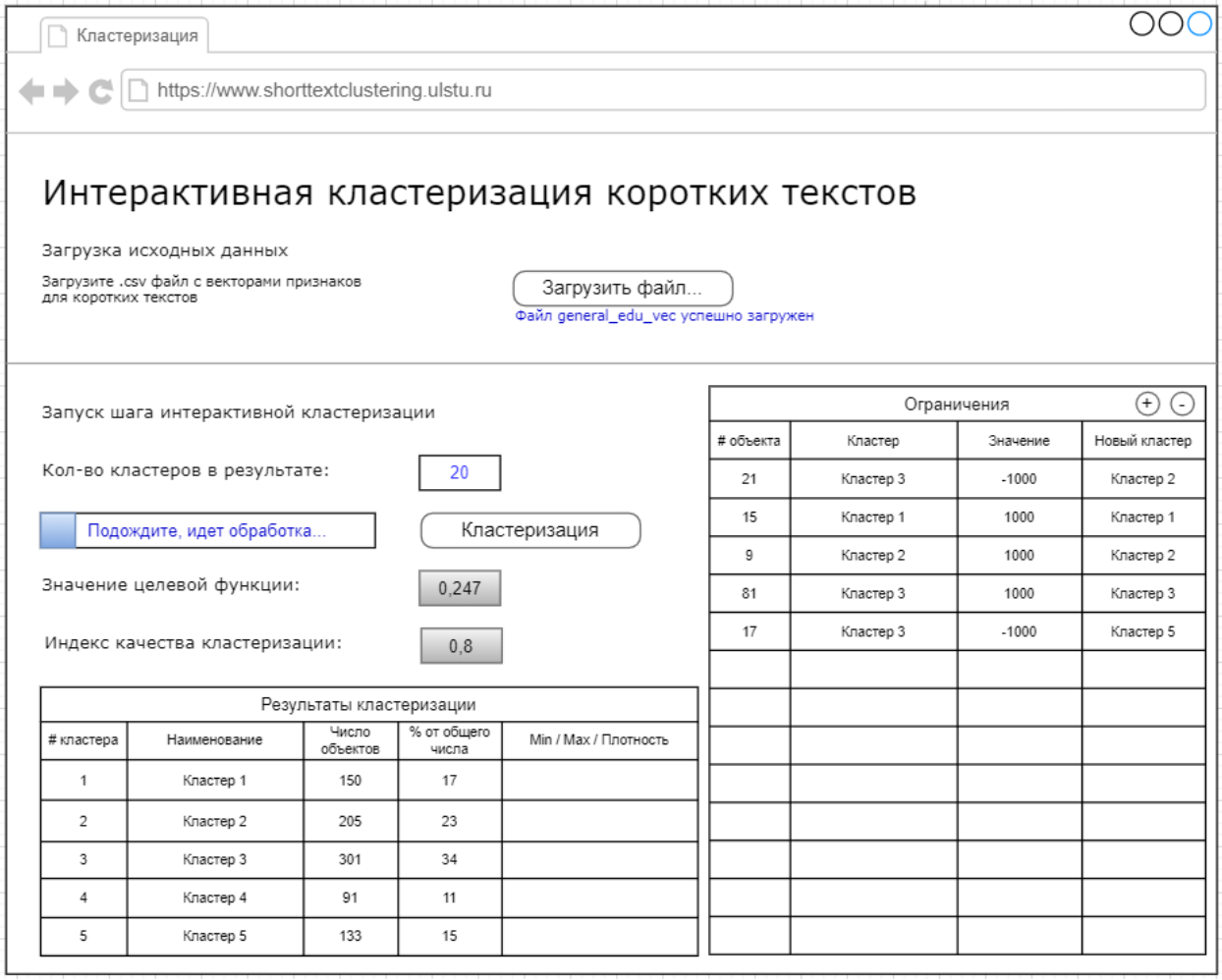


Рис. 3.31. Экранная форма кластеризации коротких текстов.

Третья экранная форма представлена на Рис. 3.32 и содержит следующие основные управляющие и визуальные элементы:

- Блок переключателей управляющих режимом отбора элементов из кластера: может быть выбран процент случайных объектов; определенное количество наиболее близких объектов или наиболее далеких объектов.
- Блок визуальных элементов в правой части формы отображает статистическую информацию о кластере: число элементов в кластере, процент от общего числа элементов и плотность объектов в кластере в евклидовом пространстве кластеризации.
- Кнопка “Показать” запускает процедуру отбора элементов кластера.

- Результат работы процедуры отбора элементов отображается в таблице “Объекты кластеризации”. Помимо номера и самого короткого текста в таблице отображается евклидово расстояние до центра кластера.

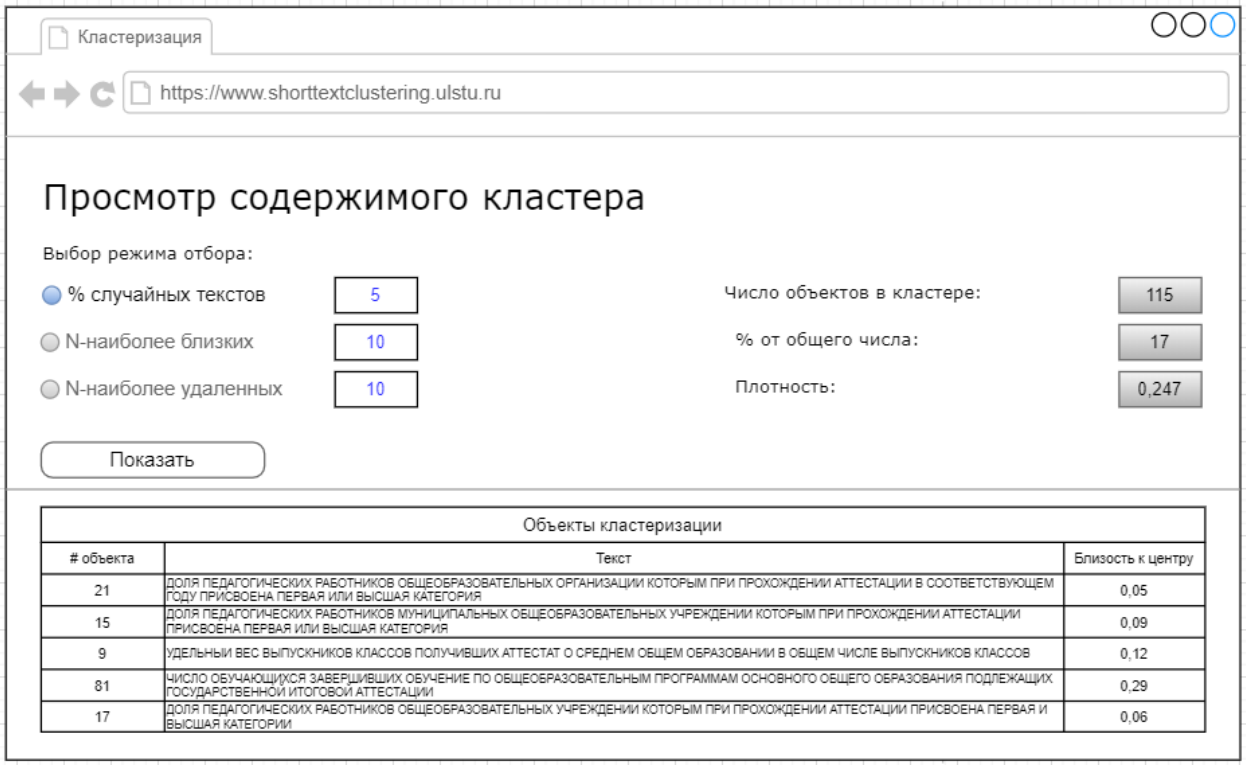


Рис. 3.32. Экранная форма просмотра содержимого кластера.

3.5. Выводы по главе

В данной главе представлено описание системы поддержки принятия решений, в рамках которой проводилась апробация предложенных моделей, методов и алгоритма. В ФИС СП накоплен большой объем коротких текстов требующих экспертного анализа.

Также приводится описание спроектированных блоков программного модуля, позволяющего автоматизировать работу эксперта для решения задачи нечеткой интерактивной кластеризации коротких текстов. Данный программный модуль использовался для апробации моделей, методов и алгоритма представленных во второй главе, оценки производительности и эффективности нечеткой

кластеризации наборов коротких текстов на русском языке без специальной предварительной обработки.

Программный модуль, выполненный в трехзвенной архитектуре, позволил эффективно организовать совместную работу экспертов в ходе решения задачи нечеткой кластеризации набора данных коротких текстов, содержащих показатели эффективности системы стратегического планирования Российской Федерации.

ГЛАВА 4. Проведение численных экспериментов для оценки эффективности алгоритма интерактивной кластеризации коротких текстов

В данной главе представлены результаты ряда экспериментов, подтверждающие работоспособность и эффективность предлагаемых моделей и алгоритма. В первой группе экспериментов показывается работа на сгенерированном наборе данных из простых векторов, процесс кластеризации которых является тривиальным. При этом в силу вариативности исследовательского намерения, любой алгоритм кластеризации может расположить их неправильно. В экспериментах показывается, каким образом исследователь может уточнить свое намерение и кластеризация подстраивается под него. В следующем разделе эксперимент проводится на каноническом наборе данных – “Ирисы Фишера”, для демонстрации того как исследователь может добавить информацию не содержащуюся в данных и тем самым улучшить результаты кластеризации. В третьем разделе проводится эксперимент по кластеризации набора коротких текстов состоящего из объявлений компании “Avito” . Приводится сравнение полученных результатов с результатами других алгоритмов кластеризации. В финальной четвертой части приводится описание результатов решения практической задачи кластеризации коротких текстов, содержащих показатели эффективности системы стратегического планирования Российской Федерации. Представлены полученные результаты и метрики производительности, позволившие успешно решить задачу.

4.1. Демонстрация работы на синтетическом наборе данных.

Для демонстрации работы сгенерирован набор данных из 400 элементов, следующим образом:

1. За основу взяты 4 вектора {(1,0,0,0); (0,1,0,0); (0,0,1,0); (0,0,0,1)}

2. Для каждого из 4-х векторов сгенерированы 125 векторов добавлением к каждой компоненте случайной величины из равномерного распределения U[0, 1/10]. Добавленный случайный шум совсем не большой, т.к. в данном эксперименте задачей ставится показать влияние обратной связи, а не качество работы алгоритма самого по себе.

3. Векторы в выборке расположены последовательно четверками, таким образом, первые 4 вектора содержат по 1 представителю от каждого базового класса. В результатах экспериментов детально будут показаны только первые 12 векторов для краткости и ясности картины результата.

Для указанного набора данных запущен алгоритм кластеризации с разбиением множества на 2 кластера. Таблица 4.1 содержит результаты работы алгоритма кластеризации, при этом для автоэнкодера достигнутое значение функции потерь на проверочной выборке равно 0.000341. На Рис. 4.33 показано распределение первых 12 векторов по кластерам. Для получения проекции на двумерную плоскость использовался метода t-SNE, значения отложенные по осям являются безразмерными величинами. Далее будут показаны три эксперимента для различных видов обратной связи: указание о необходимости включения вектора X_0 в кластер C_0; указание о необходимости исключения вектора X_1 из кластера C_1; комплексная обратная связь по замене векторов X_2 и X_3 местами в кластерах C_0 и C_1 соответственно. Все эксперименты выполняются как первая итерация после первоначальной кластеризации, а не последовательно, исключительно для более удобного сравнения. Последовательное применение, очевидно, возможно, без каких либо ограничений или особенностей в работе алгоритма.

Таблица 4.1. Список первых 4 векторов набора данных

№	Координаты				Кластер
0	**1.0191519**	0.06221088	0.04377278	0.07853585	C_0
1	0.07799758	**1.0272592**	0.02764643	0.08018722	C_0
2	0.09581394	0.08759326	**1.0357817**	0.05009951	C_1

3	0.06834629	0.07127021	0.03702508	**1.0561196**	C_1

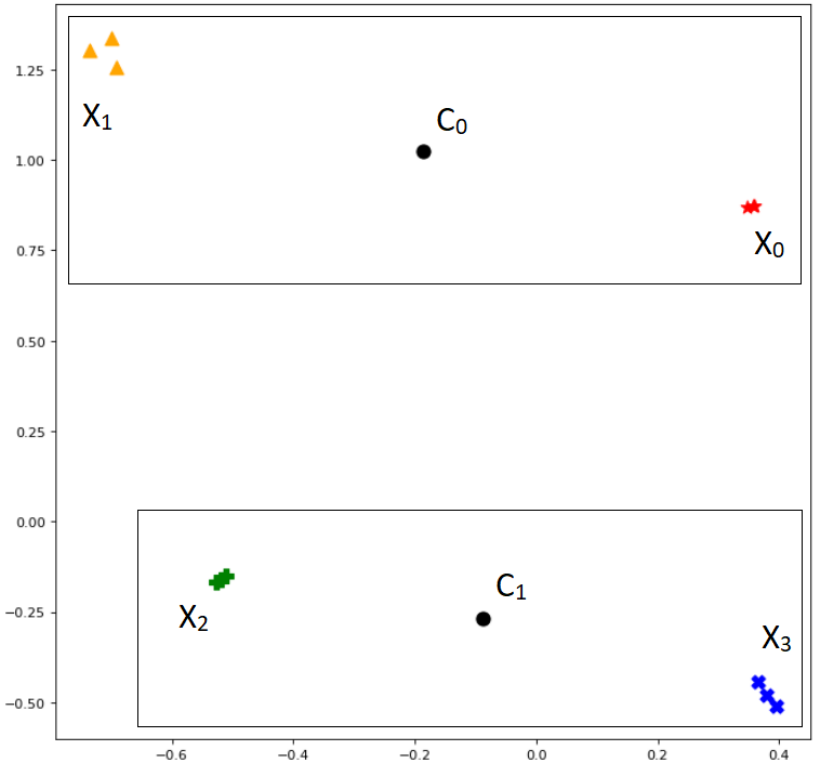


Рис. 4.33. Результат кластеризации для первых 12 векторов

В первом эксперименте предположим, что исследователь обладает информацией, о том, что вектор X_1 семантически ближе к векторам X_2 и X_3, чем к вектору X_0. Поэтому для алгоритма кластеризации формируется обратная связь в виде матрицы $T_{[500,4]} = \{t_{ij} \mid i \in [0, 500), j \in [0, 4)\}$ указывающей, что вектор X_1 должен перейти в кластер C_1.

$$t_{ij} = \begin{cases} 1000, & i = 1, \quad j = 1 \\ 1, & \text{иначе} \end{cases}$$

Результаты 100 эпох работы алгоритма кластеризации представлены на Рис. 4.34, вместе с вектором X1 в кластер C1 переместились и все остальные векторы 2 класса, при этом можно заметить, что взаимное расположение между 3-м и 4-м классами сохранилось. Также сохранилось взаимное расположение большинства векторов внутри каждого класса и относительно центра соответствующего кластера.

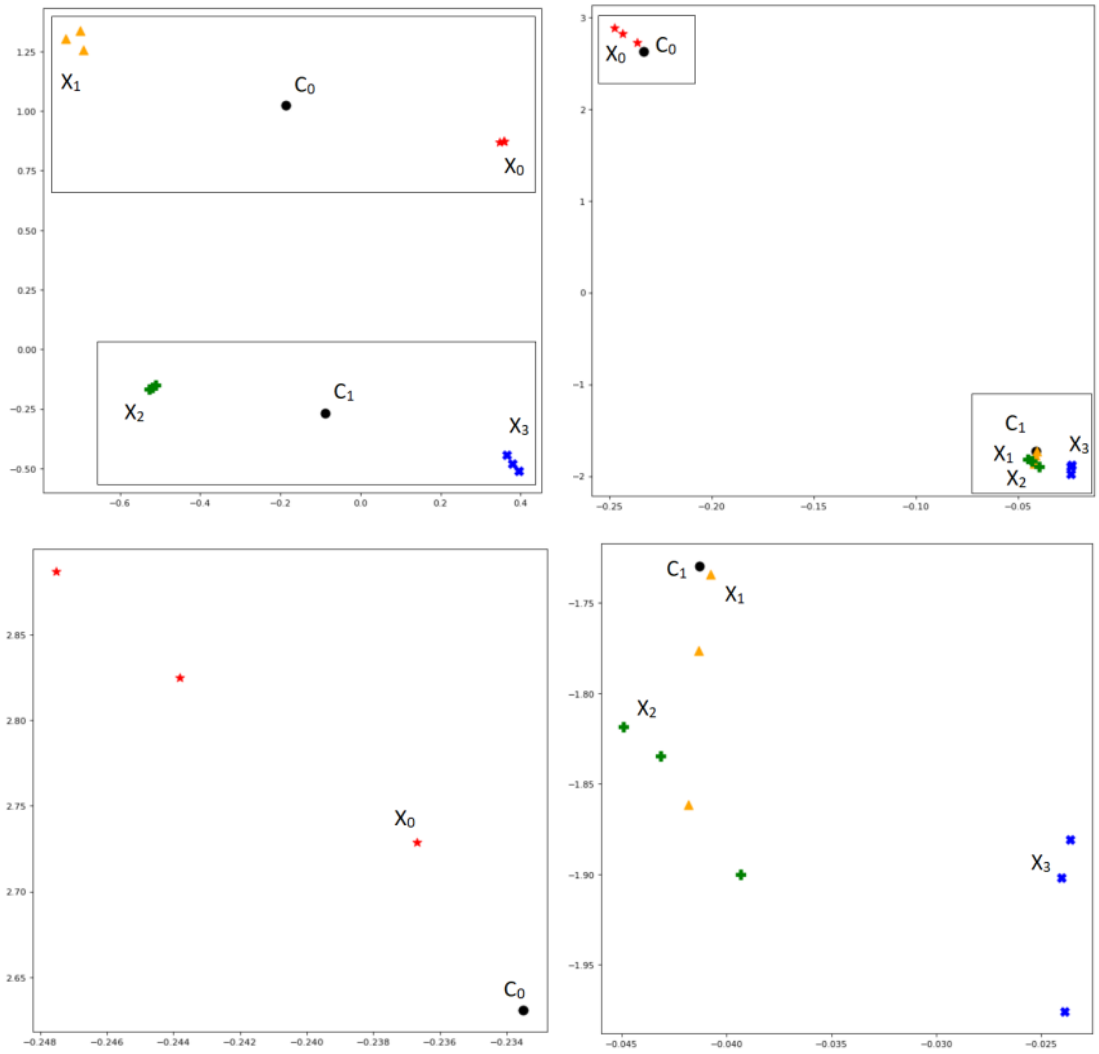


Рис. 4.34. (a,b,c,d). Результат кластеризации для перемещения вектора X1 в кластер C1; a – исходные данные, b – результат кластеризации, c и d – увеличенное представление полученных кластеров

Во втором эксперименте предположим, что исследователь обладает информацией, о том, что вектор X_2 семантически далек от вектора X_3, поэтому он должен выйти из кластера C_1. При этом кластер реципиент не известен исследователю. В случае более чем 2-х кластеров ни одному из кластеров исследователь не отдает предпочтение. Для алгоритма кластеризации формируется обратная связь в виде матрицы $T_{[500,4]} = \{t_{ij} \mid i \in [0, 500), j \in [0, 4)\}$ указывающей, что вектор X_2 должен выйти из кластера C_1.

$$t_{ij} = \begin{cases} -1000, & i = 2, \quad j = 1 \\ 1, & \text{иначе} \end{cases}$$

Результаты 100 эпох работы алгоритма кластеризации представлены на Рис. 4.35. Вектор X_2 покинул кластер C_1 и вместе с вектором X_2 в кластер C_1 переместились и все остальные векторы 3 класса. При этом можно заметить, что т.к. исключение из кластера это, по сути, ослабление силы притяжения между вектором и центром кластера, то исключаемый класс оказался максимально далеко от центра кластера C_1 и достаточно далеко от центра кластера C_0. Также можно заметить предсказуемо более низкую скорость сходимости алгоритма кластеризации при операции исключения из кластера, чем при операции включения в кластер.

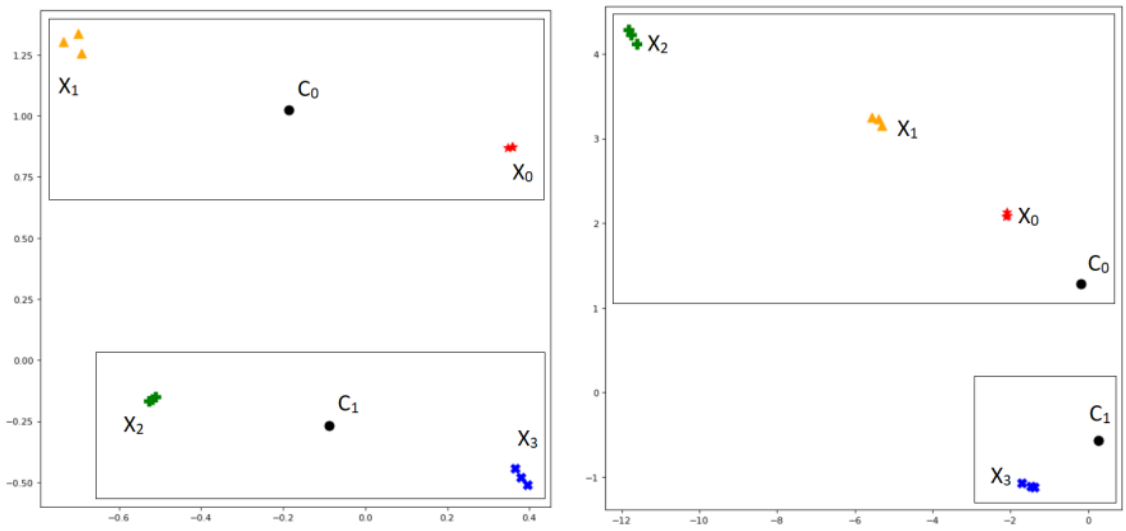


Рис. 4.35. (a,b). Результат кластеризации для исключения вектора X_2 из кластера C_1. a – исходные данные, b – результат кластеризации.

В третьем эксперименте предположим, что исследователь обладает информацией о необходимости изменить расположение сразу двух векторов. Вектор X_1 требуется перемесить в кластер C_1, а вектор X_2 переместить в кластер C_0. Для алгоритма кластеризации формируется обратная связь в виде матрицы $T_{[500,4]} = \{t_{ij} \mid i \in [0, 500), j \in [0, 4)\}$ следующего вида:

$$t_{ij} = \begin{cases} 1000, & i = 1, \quad j = 1 \\ 1000, & i = 2, \quad j = 0 \\ 1, & \text{иначе.} \end{cases}$$

Результаты 100 эпох работы алгоритма кластеризации представлены на Рис. 4.36. Классы 2 и 3 поменялись местами вслед за своим представителями векторами X_1 и X_2.

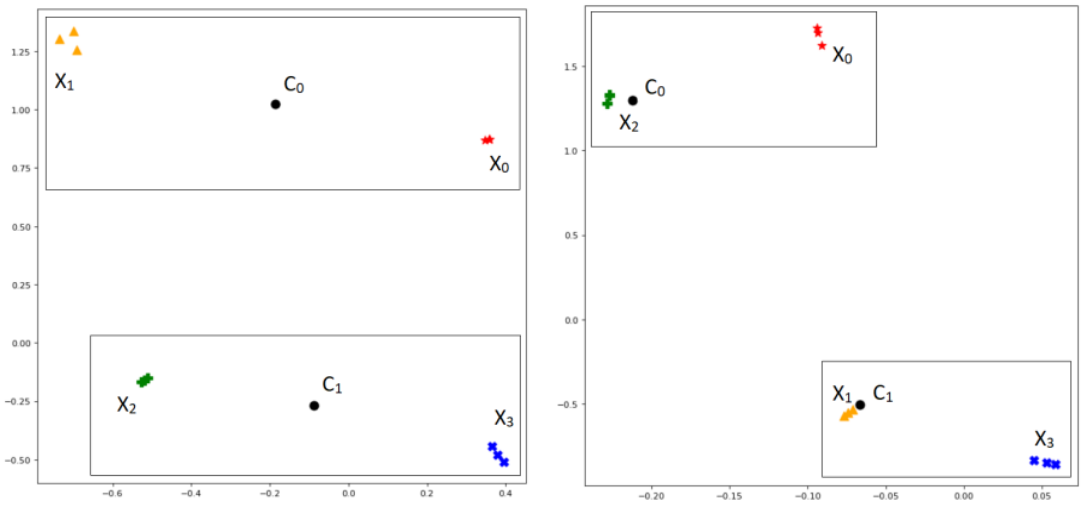


Рис. 4.36. (a,b). Результат кластеризации замены местами векторов X_1 и X_2 в кластерах. a – исходные данные, b – результат кластеризации.

В четвертом эксперименте изменим набор данных, увеличив уровень шума в векторах в десять раз добавлением к каждой компоненте случайной величины из равномерного распределения U[0, 1]. Первые 4 вектора и результат кластеризации указаны в Таблица 4.2. На Рис. 4.37a показаны первые 12 векторов и результат кластеризации. Видно, что векторы X_0 и X_2 соотнесены с кластерами не верно. Для исправления результата алгоритма кластеризации формируется обратная связь в виде матрицы $T_{[500,4]} = \{t_{ij} \mid i \in [0, 500), j \in [0, 4)\}$ следующего вида:

$$t_{ij} = \begin{cases} 1000, & i = 0, \quad j = 0 \\ 1000, & i = 2, \quad j = 1 \\ 1, & \text{иначе.} \end{cases}$$

Результаты 100 эпох работы алгоритма кластеризации представлены на Рис. 4.37b. Ошибочно соотнесенные векторы перешли в корректные классы, при этом остальные представители классов по-прежнему соотнесены корректно, т.к. внесенные изменения объективно улучшили качество кластеризации и

изменения оказали точечное воздействие, в отличие от предыдущих экспериментов.

Таблица 4.2. Векторы с увеличенным уровнем шума

№	Координаты				Кластер
0	**1.1915**	0.6221	0.4377	0.7853	C_0
1	0.7799	**1.2725**	0.2764	0.8018	C_0
2	0.9581	0.8759	**1.3578**	0.5009	C_1
3	0.6834	0.7127	0.3702	**1.5611**	C_1

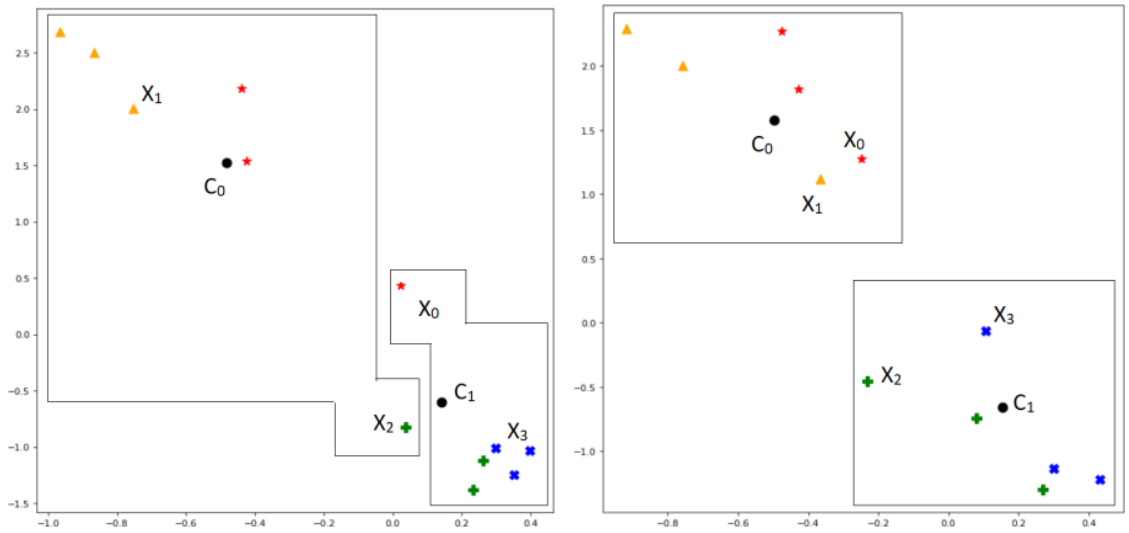


Рис. 4.37. (a,b). Результат кластеризации векторов с увеличенным шумом с точечным исправлением. a – исходные данные, b – результат кластеризации.

4.2. Демонстрация работы на примере набора данных “Ирисы Фишера”

Набор данных “Ирисы Фишера” является классическим для задач классификации и кластеризации [71]. В наборе представлены длина и ширина наружной и внутренней долей околоцветника для трех видов (‘setosa’, ‘versicolor’, ‘virginica’). В Таблица 4.3 представлены примеры 3-х векторов по одному для каждого вида. К исходным данным к каждой компоненте добавлена случайная величина из равномерного распределения U[0, 1/10] для усложнения задачи и генерации 600 примеров из 150 имеющихся.

Таблица 4.3. Пример векторов из набора данных “Ирисы Фишера”

№	Координаты				Кластер
0	5.119	3.562	1.443	0.278	C_0 (setosa)
1	7.077	3.227	4.727	1.480	C_1 (versicolor)
2	6.395	3.387	6.035	2.550	C_2 (virginica)

На Рис. 4.38a представлены результаты работы алгоритма кластеризации для 3-х кластеров. Кластер, соответствующий виду ‘setosa’, отчетливо и безошибочно отделен, а в кластерах двух других видов имеется 13 и 16 перепутанных местами векторов. Виды ‘versicolor’ и ‘virginica’ являются близкими, и даже в задачах классификации их не удается однозначно разделить. Тем не менее, предположим, что исследователь обладает информацией о двух векторах (экземплярах цветка, видовая принадлежность которых ему вполне могла быть известна), которые необходимо поменять местами. В данном примере это векторы с порядковым номером 7 и 50. Для этого формируется обратная связь в виде матрицы $T_{[600,3]} = \{t_{ij} \mid i \in [0, 600), j \in [0, 3)\}$ следующего вида:

$$t_{ij} = \begin{cases} 1000, & i = 7, \quad j = 2 \\ 1000, & i = 50, \quad j = 1 \\ 1, & \text{иначе.} \end{cases}$$

Результаты 100 эпох работы алгоритма кластеризации представлены на Рис. 4.38b (для проекции на плоскость 3-х мерных данных использовался алгоритм t-SNE реализованный в python библиотеке sklearn [86]). Ошибочно соотнесенные векторы перешли в корректные классы, при этом большая часть ошибок также исправилась. Перепутанными остались 8 и 2 векторов во 2-ом и 3-м классах соответственно. Это дает точность (accuracy) равную 0.98(3). Такие результаты трудно достижимы даже для алгоритмов классификации, средним результатов лучших алгоритмов является точность 0.971. Алгоритмы кластеризации зачастую полностью не способны различить 2-ой и 3-ий виды [68].

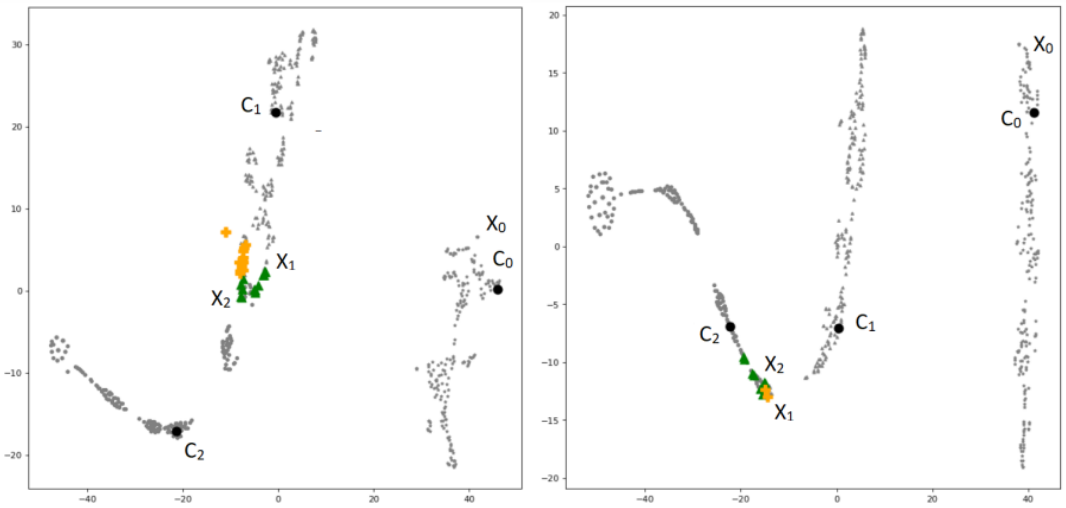


Рис. 4.38. (a,b). Результат изменения распределения по кластерам в наборе “Ирисы Фишера”. Количество неверно соотнесенных векторов уменьшилось с 26 до 10.

4.3. Демонстрация работы на примере набора данных объявлений Avito (Avito ML course - ads classification)

Набор данных от компании Avito для соревнования по классификации рекламных объявлений (Рис. 4.39). Все объявления в наборе разбиты на 4 категории:

- “Бытовая электроника”,
- “Для дома и дачи”,
- “Личные вещи”,
- “Хобби и отдых”.

Количество объявлений: 489 000.

Примеры объявлений (исходная орфография сохранена):

- “Продам мини степпер в идеальном состоянии. Цена в магазине ~ 2700”;
- “Красивое платье Next. Плотная теплая ткань. р.42-44. Смотреть уралмаш или центр.”;

- “Продам новую игру "Приключения Алисы в стране чудес" Игра не вскрывалась. В оригинальной упаковке. Производитель "Десятое королевство””,
- “Продам импортную чугунную ванну марки Roca размеры 1700х700 б/у в хорошем состоянии. Срочно, крайне дешево, с условием самовывоза”.
- “Продаю серьги золото 585, клейма заводские, вес 1,4 гр с фианитами, подходят в основном для молодых девушек, необычные”.

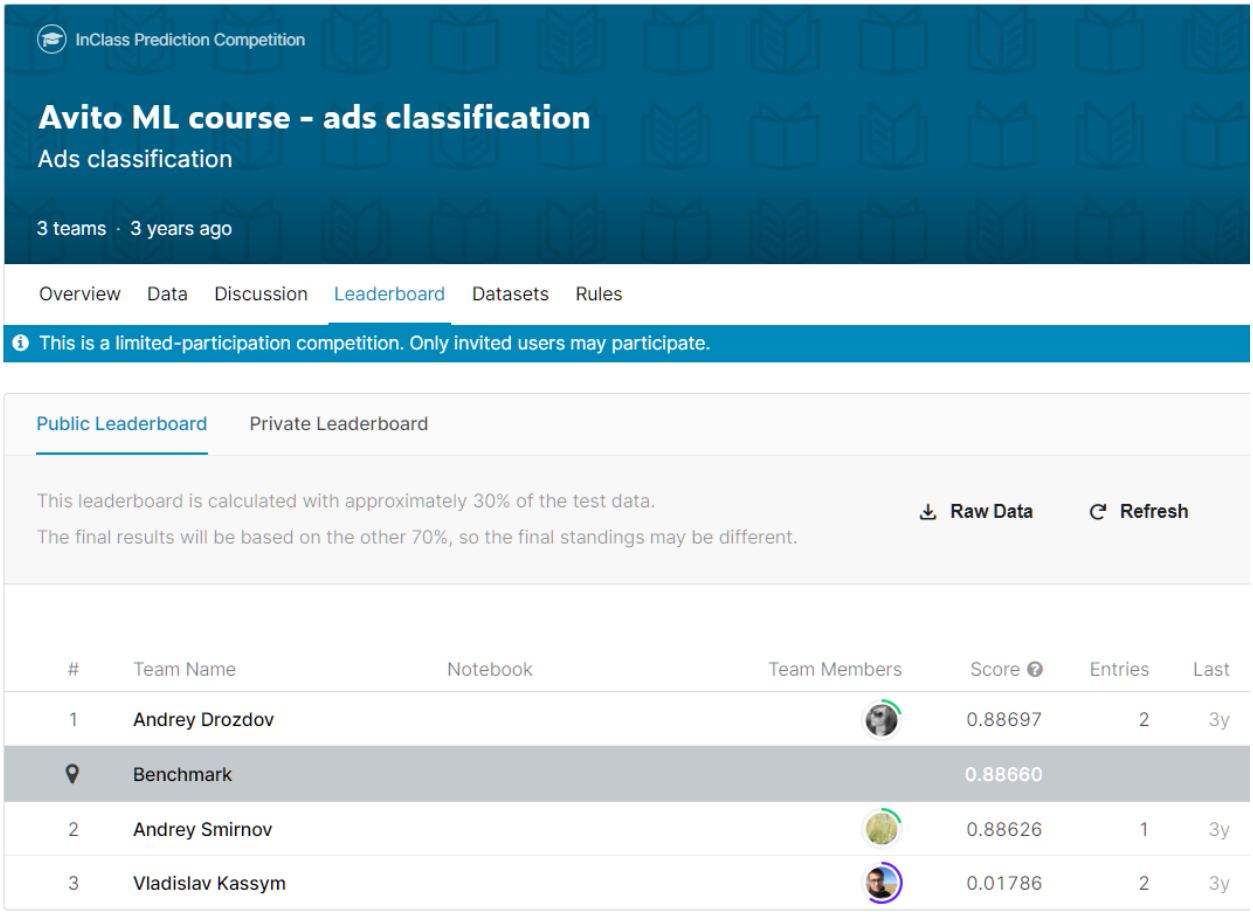


Рис. 4.39. Результаты конкурса проводимого компанией Avito по классификации объявлений.

4.3.1. Фаза подготовки данных для языковой модели

Фаза подготовки данных для языковой модели выглядит следующим образом:

1. На вход метода подается короткий текст.

Пример: “Продам новую игру "Приключения Алисы в стране чудес" Игра не вскрывалась. В оригинальной упаковке. Производитель Десятое королевство”

2. В начале этого шага проводится корректировка орфографии внешним сервисом от Yandex (может быть использован любой другой). Далее незнакомые для предварительно обученной сети слова заменяются токеном “UNK”, также, согласно правилам выбранной языковой модели (в данном случае UMLFit), добавляется ряд других токенов помогающих передать специфику предложения (например, пометка заглавных букв). В итоге текст выглядит следующим образом.

Пример: “xxbos xxmaj продам новую игру xxmaj приключения xxmaj алисы в стране чудес xxmaj игра не вскрывалась . в оригинальной упаковке . xxmaj производитель xxmaj xxunk королевство”.

3. Каждое слово в подготовленном на этапе 2 тексте преобразуется в 1-hot вектор, длина которого равна размеру словаря языковой модели, все компоненты вектора нулевые, за исключением i-ой, соответствующей искомому слову в словаре языковой модели. В итоге на вход сети подается тензор с последовательностью векторов (номера соответствуют порядковому номеру слова в словаре).

Пример: “tensor([[2, 5, 27, 547, 1896, 5, 9268, 5, 36053, 12,
6162, 22122, 5, 933, 23, 9659, 9, 12, 2871, 231,
9, 5, 153, 5, 0, 27770]]) tensor([0])”

Отдельно следует отметить, что в данной процедуре подготовки не используется этап лемматизации, характерный ранее для большинства методов предварительной обработки текстов. Эксперимент, в котором в языковую модель подавались лемматизированные предложения показал, что при переходе к набору текстов “Avito” точность языковой модели (accuracy) упала с 43% до 23%. Частично это объясняется тем, что при обучении исходной языковой модели, к которой применен transfer learning, происходило без лемматизации.

```
In [28]: col_names = ['label','text']

         df_train_res = pd.DataFrame(train_list, columns = col_names)
         df_train_res
```

Out[28]:

	label	text
0	1	Гобелен . Размеры 139x84 см .
1	1	продать недорого 4 стула из светлой прессованн...
2	2	Мини баня МБ-1(мини сауна) , предназначить дл...
3	3	продать эксклюзивный коллекция книга , выпусти...
4	0	продаваться ноутбук acer e5 - 511c2ta . купить...
...	...	...
489512	3	гитара новый в жестком кейс . Цвет чёрный .
489513	3	Резолюции и постановления пленум ЦК ВКПб с 1...
489514	1	входной металлический дверь кайзер e40м. Дверь...
489515	0	продать чехол - книжка на магнит . цена 300 ру...
489516	3	Бас - гитара фирмы cort , модель gb - pj . Осн...

489517 rows × 2 columns

Рис. 4.40. Пример текстов из набора данных Avito после процедуры лемматизации.

4.3.2. Фаза расширения словаря языковой модели

Фаза расширения словаря языковой модели. Как уже было сказано, все незнакомые слова заменяются на токен UNK, который впоследствии подается как среднее всех векторов словаря. При переходе к набору текстов Avito точность языковой модели (accuracy) упала с 43% до 31%. Тюнинг модели позволяет вернуть точность в район 43%. Но такой точности оказывается недостаточно для качественной работы алгоритма кластеризации. Для повышения точности используется описанный во второй главе метод расширения языковой модели. Это позволяет достичь 50-52% точности языковой модели на выбранном корпусе.

4.3.3. Фаза первичной кластеризации

Фаза первичной кластеризации. В начале этой фазы, как описывалось выше, происходит настройка весов в слоях блока кластеризации. На входе у этого блока находятся векторы размерности 400. На Рис. 4.41 приведена проекция векторов получаемых на выходе энкодера нейронной сети на плоскость стандартным алгоритмом t-SNE. Настройка весов блока кластеризации нейронной сети происходит для каждого слоя последовательно. Каждый слой рассматривается как автоэнкодер, и происходит его обучение. После настройки слоев нейронной сети все тексты пропускаются через нейронную сеть и на выходе получается набор векторов размерности 4 (размер 4 взят по количеству искомых кластеров, но это не обязательное условие). Далее для этих векторов методом k-means находятся первичные центры искомых кластеров. На Рис. 4.42 приведена проекция векторов полученных в результате первого шага кластеризации на двумерную плоскость стандартным алгоритмом T-SNE. На этой фазе в результате 10-ти прогнозов усредненное значение точности кластеризации составляет 52,5%.

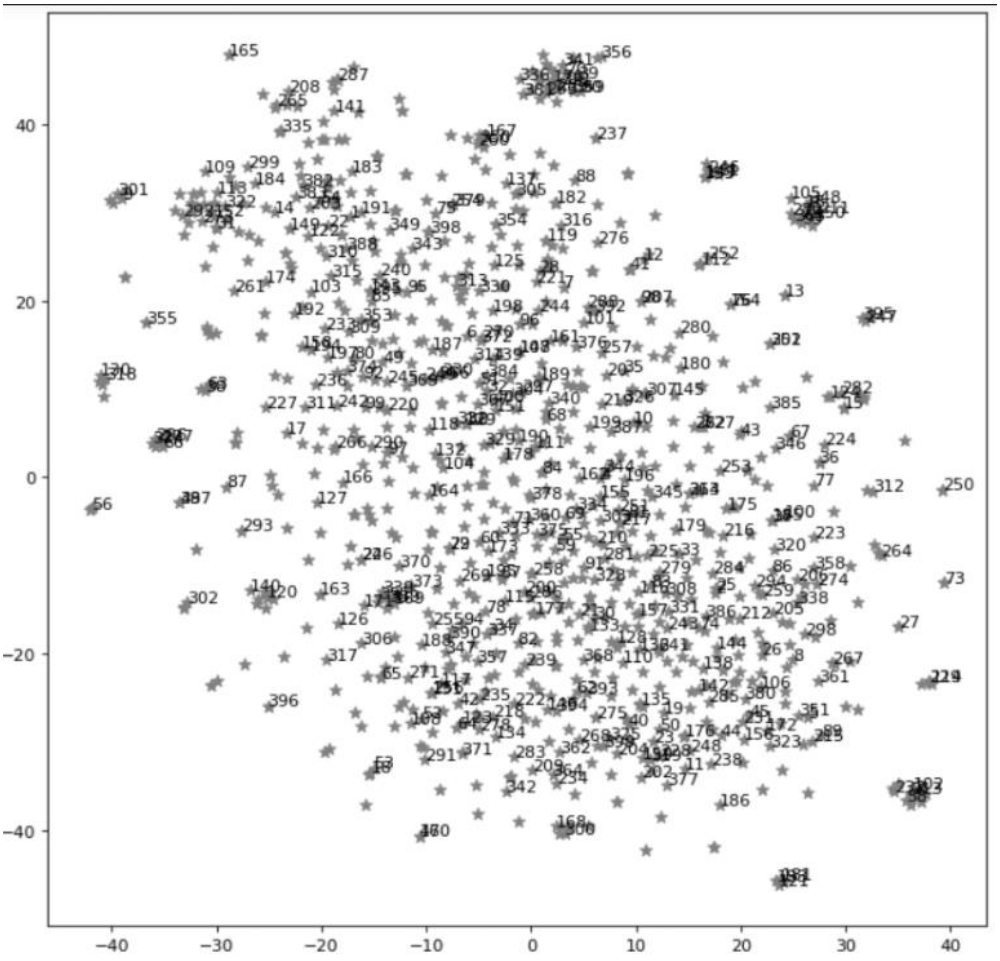


Рис. 4.41. Проекция набора векторов полученных в результате работы блока энкодера языковой модели.

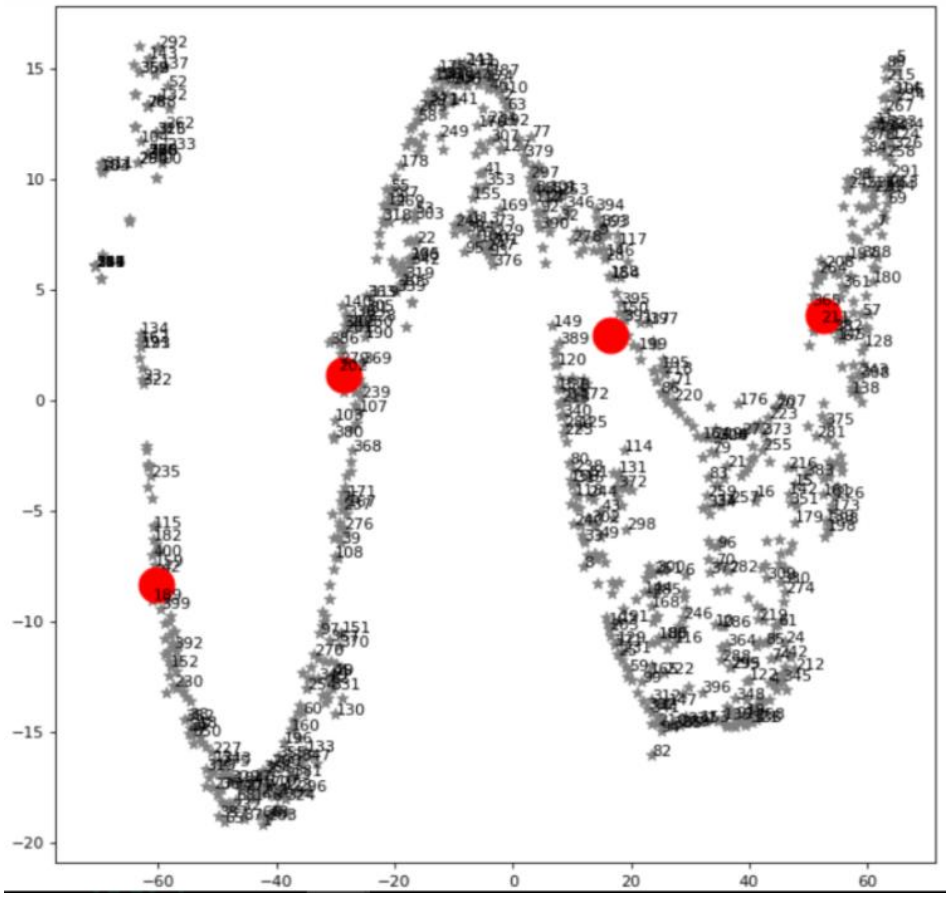


Рис. 4.42. Проекция набора векторов полученных в результате работы первого этапа кластеризации с указанием первичных центров кластеров.

4.3.4. Фаза интерактивной кластеризации

Фаза интерактивной кластеризации. На этой фазе эксперт просматривает результаты кластеризации и дает обратную связь по ним. В Таблицах 4 и 5 представлен пример результатов кластеризации по кластеру номер 3, с выводом наиболее близких и наиболее далеких элементов (векторов соответствующих коротким текстам) от центра кластера. В качестве расстояния в модели используется евклидова метрика. Обратная связь эксперта предоставляется в двух видах: либо “элемент X_i должен принадлежать кластеру C_j” либо “элементу X_i не следует находиться в кластере C_j”. Одновременно может быть применено произвольное количество таких ограничений. В Таблица 4.4 и Таблица 4.5 представлена обратная связь эксперта в колонке “Действие”.

Таблица 4.4. Результаты первичной кластеризации по кластеру номер 3 (“бытовая техника”), наиболее близкие элементы к центру кластера.

ID	Текст	Действие
266109	Полностью исправен , полный комплект. Дисплей под плёнкой , корпус на 4 , косяки на фото . Торг.	
266065	продам nokia c3 с сенсером, сенсер сломан, можете взять его на запчасти я не против и меняю на самбуфер	
262434	Продам новую игру "Приключения Алисы в стране чудес" Игра не вскрывалась. В оригинальной упаковке. Производитель "Десятое королевство"	Добавить в кластер #1
264447	Продается телевизор Сокол с пультом . Включается но непоказывает изображения нет . Только полосы по экрану .	
265652	Планшету только месяц все работает бронь плёнка чуть треснутая без торга	

Таблица 4.5. Результаты первичной кластеризации по кластеру номер 3 (“бытовая техника”), наиболее удаленные элементы от центра кластера.

ID	Текст	Действие
263794	Продам отличный инструмент! НОВЫЙ, без пробега! Полностью работоспособный! Имеются все документы, коробка. За такую цену - это подарок!	Исключить
263995	Купил звукосниматель на алиэкспресс,пока дошла до меня у меня уже купили гитару,звукосниматель совсем новый.	Исключить
264158	Давольно в хорошем состоянии.экран целый. Телевизор "Аврора". Ленинградский завод им Козицкого. Модель 1967 года.	

По результатам полученной обратной связи от эксперта составляется матрица обратной связи, по правилам, описанным во второй главе. На Рис. 4.43 представлен пример формирования матрицы обратной связи на языке Python. Далее происходит одновременное обновление весов нейронной сети и подстройка центров кластеров по формулам (2.7)-(2.8) с учетом построенной матрицы обратной связи и получение обновленных векторов для всех коротких текстов с учетом измененных весов нейронной сети и центров кластеров. Эта фаза повторяется необходимое количество раз, до тех пор пока либо эксперт не перестанет добавлять новые ограничения, либо экспертом будет принято решение о достижении требуемого качества кластеризации. В данном эксперименте было сделано 10 итераций, достигнута точность 80,1%. Результаты финальной итерации представлены на Рис. 4.45.

```
In [100]: l_tips = np.zeros([len(l_X),2])
          l_tips[[1], [1]] = 1000  # элемент 1 должен войти в кластер 1
          l_tips[[2], [0]] = -1000 # элемент 2 должен выйти из кластера 0
          print(l_tips[:5,:])

          [[   0.    0.]
           [   0. 1000.]
           [1000.    0.]
           [   0.    0.]
           [   0.    0.]]
```

Рис. 4.43. Пример заполнения матрицы обратной связи на языке Python.

4.3.5. Анализ качества интерактивной кластеризации

Зависимость качества от количества ограничений. С ростом количества ограничений точность повышается. При добавлении количества ограничений более 5% от общего числа примеров график точности достигает своего максимума. Для сравнения на Рис. 4.44 приведен график результатов схожего эксперимента на англоязычном корпусе Reuters RCV1-v2/LYRL2004 (Cohn, 2008). При этом нужно отметить, что кластеризация проводилась для 125 примеров, а

оцифровка текстов проводилась на основании метрики TF-IDF (словарь 2000 слов), что является существенно менее сложной задаче для кластеризации.

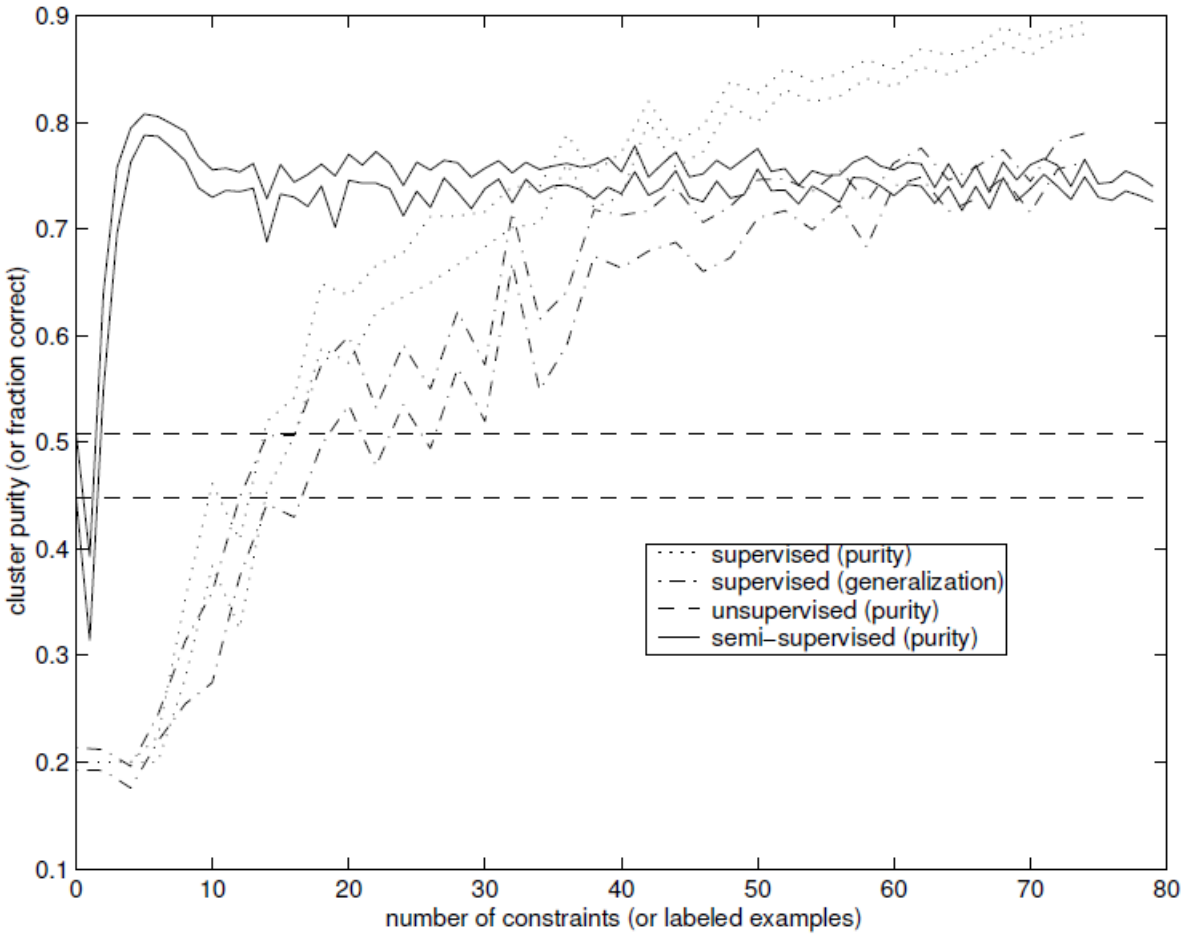


Рис. 4.44. Результаты интерактивной кластеризации корпуса Reuters [134].

Также важно отметить влияние дообучения слоев языковой модели, которое дает до 10% точности и понижает количество необходимых ограничений. Важность дообучения языковой модели также показывает и сравнение качества классификации полученного в результатах соревнования на Kaggle и алгоритмом классификации на базе доученной языковой модели с тем же энкодером что и в методе кластеризации. Полученная точность выше почти на 2,5%.

В рамках данного эксперимента была проведена кластеризация набора коротких текстов “Avito” известными классическими алгоритмами, реализация которых доступна в библиотеке sclearn: k-means, BIRCH, DBScan. На вход этих методов были поданы сжатые векторные представления коротких текстов, полученные из языковой модели после шага дополнительного обучения предлага-

емого в данной работе алгоритма. На графиках видно, что даже на начальном этапе данные алгоритмы уступают предложенному, при этом разница не может считаться значительной. Но уже после первого шага интерактивной кластеризации достигается разница в точности кластеризации более чем на 6%.

Все полученные результаты приведены на Рис. 4.45 для удобства визуального сравнения.

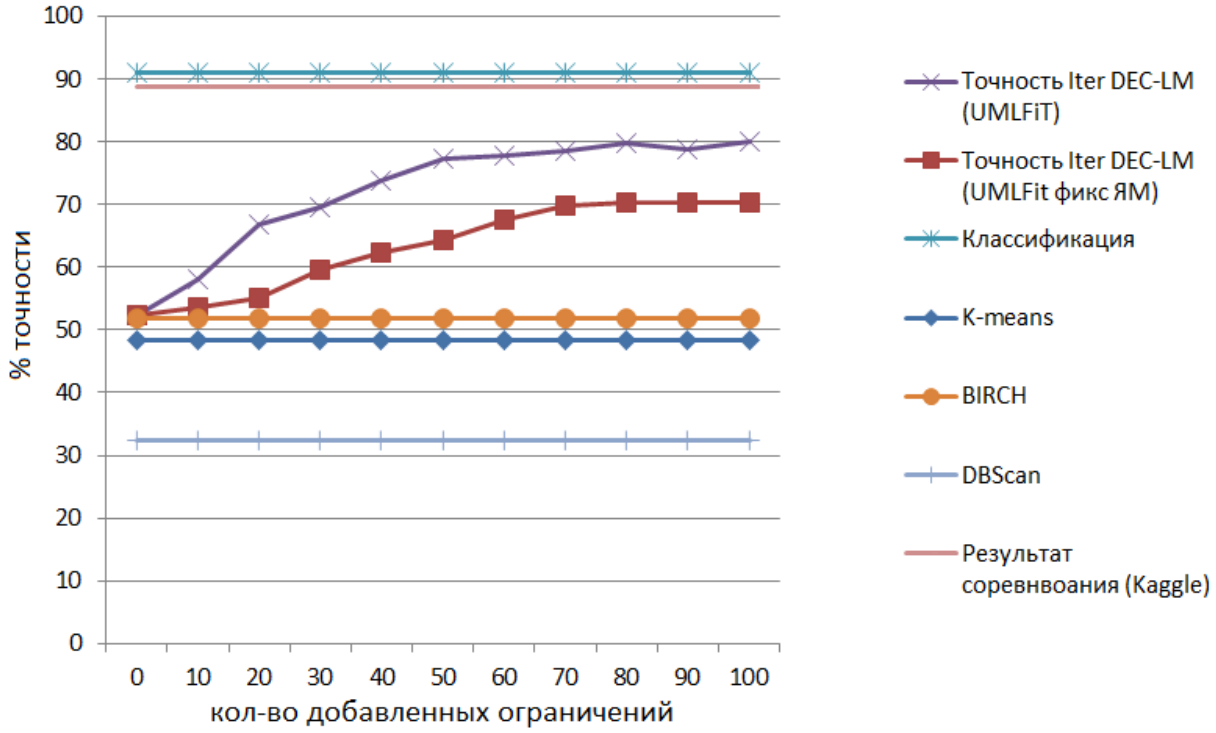


Рис. 4.45. Зависимость точности полученных результатов от количества ограничений от эксперта.

4.3.6. Оценка границ применимости

Оценка границ применимости. В литературе нет однозначного критерия для определения короткого текста. Это может быть и предложение с 5-15 словами, так и тексты размером в несколько абзацев, а иногда и страниц.

В наборе данных Avito были выбраны группы текстов с различными размерами: 5-20 слов, 20-50 слов, 50-100 слов, 100-200 слов, 200-300 слов, 300-500 слов, 500-1000 слов. Эксперименты показывают постепенное снижение точно-

сти кластеризации с увеличением размеров текста (замер проводился после первой фазы кластеризации, без добавления экспертных ограничений). Результаты представлены на Рис. 4.46. Также на Рис. 4.46 показаны результаты двух экспериментов после 5-ти итераций интерактивной кластеризации для диапазонов 5-20 и 500-100. Интерактивная кластеризация при низкой точности первичной кластеризации оказывается малоэффективной, что объясняется низким качеством векторных представлений для длинных текстов.

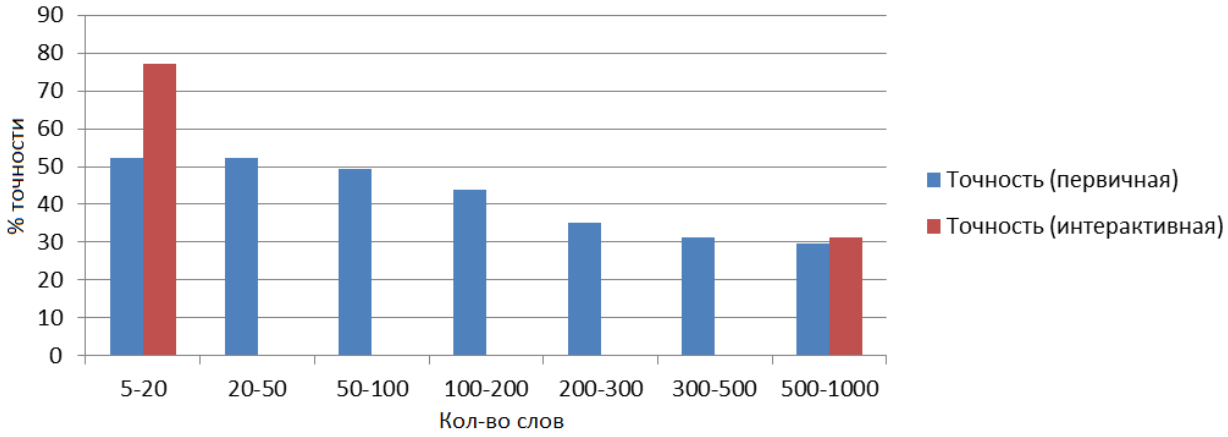


Рис. 4.46. Зависимость точности предлагаемого метода от длины текстов.

Частично, это можно объяснить тем, что параметр LSTM-слоя отвечающий за количество обрабатываемых слов в последовательности в данной модели равен 25. А также тем, что для больших текстов результирующий вектор значительно усредняется за счет большого числа различных мыслей и смыслов в тексте.

4.4. Решение практической задачи по кластеризации показателей системы стратегического планирования Российской Федерации

Для апробации алгоритма на актуальных задачах предприятия была выбрана задача кластеризации набора ключевых показателей эффективности (КПЭ, KPI) системы стратегического планирования Российской Федерации описанная в третьей главе. Документы стратегического планирования это открытые данные, каждый документ содержит набор КПЭ. В среднем КПЭ со-

держит одно предложение, состоящее из 5-15 слов. Количество показателей на момент начала работы приближалось к 500 000, и имелась лишь верхнеуровневая классификация документов по категориям единой межведомственной информационно – статистической системы (ЕМИСС) [122]. Первым шагом на пути к решению поставленной задач требовалось провести исследование структуры собранного набора данных. Результатом этого исследования явился трехуровневый классификатор документов стратегического планирования. Для этого обучена языковая модель методом описанным во второй главе данной диссертации. С помощью обученной нейросети были получены векторы признаков для каждого из показателей набора данных и проведена кластеризация предложенным методом. Таблица 4.6 содержит результаты первичной кластеризации тематики “Общее образование”. Общий объем набора данных для кластеризации составил 6 975 показателей (предварительно в тексте показателей были исправлены опечатки через специализированный сервис). В данном эксперименте не приводятся показатели точности кластеризации, т.к. не существовало единой версии правильного разбиения. Приведенные результаты кластеризации являлись основой для дальнейшей работы экспертов и выработки консолидированного решения по тому как должен выглядеть итоговый классификатор.

Таблица 4.6. Результаты первичной кластеризации показателей эффективности

Класс, определенный в результате кластеризации	Кол-во
Повышение квалификации персонала	135
Научные, творческие и спортивные мероприятия	389
Технологическая оснащённость	245
Расширение сети образовательных учреждений	60
Государственные стандарты	874
Численность персонала	271
Инклюзивное образование	238
Единый государственный экзамен	809
Заработная плата персонала	500

Образовательной инициативы «Наша новая школа»	36
Оснащенность образовательных учреждений	126
Расход бюджета на обучающихся	30
Доступность образовательных услуг	794
Аттестация	378
Здоровье обучающихся	70
Ремонт образовательных учреждений	143
Удовлетворённость образовательными услугами	150
Эффективность образовательной деятельности	31
Оснащённость образовательных учреждений	25
Центры прикладной квалификации	6
Безопасность	16
Эффективность заключения контрактов с персоналом	7
Физическая культура и спорт	52

Для анализа результатов кластеризации требуется привлечение эксперта по данной предметной области. При этом эксперт в случае несогласия с результатами кластеризации должен иметь возможность повлиять на эти результаты точечно, не меняя кардинально всей структуры результата. Для автоматизации и повышения эффективности работы эксперта использован метод, описанный во второй главе и реализованный в программном комплексе, описанном в третьей главе настоящей диссертации. Кластеры, сформированные в результате кластеризации всех 6 975 показателей, были доступны для анализа, в ходе которого эксперт точечно вносил корректировки в кластеры в виде дополнительных параметров кластеризации и запускал очередную итерацию алгоритма кластеризации.

Рассмотрим кластер №14, который при первичной кластеризации назван как “Аттестация”. При знакомстве с содержимым кластера эксперт отметил две принципиально разных группы показателей: аттестация преподавателей и аттестация учащихся (см. Таблица 4.7). Для изменения кластеризации первые три показателя из нового класса “Аттестация персонала” были помечены для выне-

сения из состава кластера. В результате образован новый кластер для аттестации персонала, а кластер №14 переименован в “Аттестацию учащихся”.

Таблица 4.7. Показатели из кластера "Аттестация"

“ДОЛЯ ПЕДАГОГИЧЕСКИХ РАБОТНИКОВ ОБЩЕОБРАЗОВАТЕЛЬНЫХ ОРГАНИЗАЦИЙ КОТОРЫМ ПРИ ПРОХОЖДЕНИИ АТТЕСТАЦИИ В СООТВЕТСТВУЮЩЕМ ГОДУ ПРИСВОЕНА ПЕРВАЯ ИЛИ ВЫСШАЯ КАТЕГОРИЯ”
“ДОЛЯ ПЕДАГОГИЧЕСКИХ РАБОТНИКОВ МУНИЦИПАЛЬНЫХ ОБЩЕОБРАЗОВАТЕЛЬНЫХ УЧРЕЖДЕНИЙ КОТОРЫМ ПРИ ПРОХОЖДЕНИИ АТТЕСТАЦИИ ПРИСВОЕНА ПЕРВАЯ ИЛИ ВЫСШАЯ КАТЕГОРИЯ”
“ДОЛЯ ПЕДАГОГИЧЕСКИХ РАБОТНИКОВ ОБЩЕОБРАЗОВАТЕЛЬНЫХ ОРГАНИЗАЦИЙ КОТОРЫМ ПРИ ПРОХОЖДЕНИИ АТТЕСТАЦИИ ПРИСВОЕНА ПЕРВАЯ ИЛИ ВЫСШАЯ КАТЕГОРИЯ”
“ДОЛЯ ПЕДАГОГИЧЕСКИХ РАБОТНИКОВ ОБЩЕОБРАЗОВАТЕЛЬНЫХ ОРГАНИЗАЦИЙ КОТОРЫМ ПРИ ПРОХОЖДЕНИИ АТТЕСТАЦИИ ПРИСВОЕНА ПЕРВАЯ ИЛИ ВЫСШАЯ КАТЕГОРИЯ”
“ДОЛЯ ПЕДАГОГИЧЕСКИХ РАБОТНИКОВ ОБЩЕОБРАЗОВАТЕЛЬНЫХ ОРГАНИЗАЦИЙ КОТОРЫМ ПРИ ПРОХОЖДЕНИИ АТТЕСТАЦИИ ПРИСВОЕНА ВЫСШАЯ ИЛИ ПЕРВАЯ КАТЕГОРИЯ”
“ДОЛЯ ПЕДАГОГИЧЕСКИХ РАБОТНИКОВ ОБЩЕОБРАЗОВАТЕЛЬНЫХ ОРГАНИЗАЦИЙ КОТОРЫМ ПРИ ПРОХОЖДЕНИИ АТТЕСТАЦИИ ПРИСВОЕНА ПЕРВАЯ ИЛИ ВЫСШАЯ КАТЕГОРИЯ”
“ДОЛЯ ПЕДАГОГИЧЕСКИХ РАБОТНИКОВ ОБЩЕОБРАЗОВАТЕЛЬНЫХ ОРГАНИЗАЦИЙ КОТОРЫМ ПРИ ПРОХОЖДЕНИИ АТТЕСТАЦИИ ПРИСВОЕНА ПЕРВАЯ ИЛИ ВЫСШАЯ КАТЕГОРИЯ”
“ДОЛЯ ПЕДАГОГИЧЕСКИХ РАБОТНИКОВ ОБЩЕОБРАЗОВАТЕЛЬНЫХ УЧРЕЖДЕНИЙ КОТОРЫМ ПРИ ПРОХОЖДЕНИИ АТТЕСТАЦИИ ПРИСВОЕНА ПЕРВАЯ И ВЫСШАЯ КАТЕГОРИИ”
“ДОЛЯ ПЕДАГОГИЧЕСКИХ РАБОТНИКОВ ОБЩЕОБРАЗОВАТЕЛЬНЫХ ОРГАНИЗАЦИЙ КОТОРЫМ ПРИ ПРОХОЖДЕНИИ АТТЕСТАЦИИ ПРИСВОЕНА ПЕРВАЯ ИЛИ ВЫСШАЯ КВАЛИФИКАЦИОННАЯ КАТЕГОРИЯ”
“ДОЛЯ ПЕДАГОГИЧЕСКИХ РАБОТНИКОВ ПРОШЕДШИХ АТТЕСТАЦИЮ С ПРИСВОЕНИЕМ ПЕРВОЙ КВАЛИФИКАЦИОННОЙ КАТЕГОРИИ ОТ ОБЩЕГО ЧИСЛА ПЕДАГОГИЧЕСКИХ РАБОТНИКОВ ОБЩЕОБРАЗОВАТЕЛЬНЫХ УЧРЕЖДЕНИЙ”
“СОХРАНЕНИЕ ДОЛИ ПЕДАГОГИЧЕСКИХ РАБОТНИКОВ ОБЩЕОБРАЗОВАТЕЛЬНЫХ ОРГАНИЗАЦИЙ КОТОРЫМ ПРИ ПРОХОЖДЕНИИ АТТЕСТАЦИИ ПРИСВОЕНА ПЕРВАЯ ИЛИ ВЫСШАЯ КАТЕГОРИЯ ЕЖЕГОДНО”
“ДОЛЯ ОБУЧАЮЩИХСЯ КЛАССОВ НЕ ПРОШЕДШИХ ГОСУДАРСТВЕННУЮ ИТОГОВУЮ АТТЕСТАЦИЮ В ФОРМЕ ЕГЭ В ОБЩЕЙ ЧИСЛЕННОСТИ ОБУЧАЮЩИХСЯ КЛАССОВ”
“ДОЛЯ ОБУЧАЮЩИХСЯ КЛАССОВ НЕ ПРОШЕДШИХ ГОСУДАРСТВЕННУЮ ИТОГОВУЮ АТТЕСТАЦИЮ В ФОРМЕ В ОБЩЕЙ ЧИСЛЕННОСТИ ОБУЧАЮЩИХСЯ КЛАССОВ МУНИЦИПАЛЬНЫХ

ОБЩЕОБРАЗОВАТЕЛЬНЫХ ОРГАНИЗАЦИЙ”
“ДОЛЯ ОБУЧАЮЩИХСЯ КЛАССОВ НЕ ПРОШЕДШИХ ГОСУДАРСТВЕННУЮ ИТОГОВУЮ АТТЕСТАЦИЮ В ОБЩЕЙ ЧИСЛЕННОСТИ ОБУЧАЮЩИХСЯ КЛАССОВ МУНИЦИПАЛЬНЫХ ОБЩЕОБРАЗОВАТЕЛЬНЫХ УЧРЕЖДЕНИЙ”
“ДОЛЯ ОБУЧАЮЩИХСЯ КЛАССОВ НЕ ПРОШЕДШИХ ГОСУДАРСТВЕННУЮ ИТОГОВУЮ АТТЕСТАЦИЮ В ФОРМЕ В ОБЩЕЙ ЧИСЛЕННОСТИ ОБУЧАЮЩИХСЯ КЛАССОВ МУНИЦИПАЛЬНЫХ ОБЩЕОБРАЗОВАТЕЛЬНЫХ ОРГАНИЗАЦИЙ”
“ДОЛЯ ВЫПУСКНИКОВ ОБЩЕОБРАЗОВАТЕЛЬНЫХ УЧРЕЖДЕНИЙ ОСВОИВШИХ ОСНОВНЫЕ ОБЩЕОБРАЗОВАТЕЛЬНЫЕ ПРОГРАММЫ СРЕДНЕГО ОБЩЕГО ОБРАЗОВАНИЯ СДАВШИХ ЕДИНЫЙ ГОСУДАРСТВЕННЫЙ ЭКЗАМЕН И ПОЛУЧИВШИХ АТТЕСТАТЫ”
“ДОЛЯ ОБУЧАЮЩИХСЯ МУНИЦИПАЛЬНЫХ ОБЩЕОБРАЗОВАТЕЛЬНЫХ УЧРЕЖДЕНИЙ ОСВАИВАЮЩИХ ОСНОВНЫЕ ОБРАЗОВАТЕЛЬНЫЕ ПРОГРАММЫ НАЧАЛЬНОГО ОБЩЕГО ОСНОВНОГО ОБЩЕГО И СРЕДНЕГО ПОЛНОГО ОБЩЕГО ОБРАЗОВАНИЯ В СООТВЕТСТВИИ С НОВЫМИ ФЕДЕРАЛЬНЫМИ ГОСУДАРСТВЕННЫМИ ОБРАЗОВАТЕЛЬНЫМИ СТАНДАРТАМИ НАЧАЛЬНОГО ОБЩЕГО ОСНОВНОГО ОБЩЕГО И СРЕДНЕГО ПОЛНОГО ОБЩЕГО ОБРАЗОВАНИЯ В ОБЩЕЙ ЧИСЛЕННОСТИ ОБУЧАЮЩИХСЯ МУНИЦИПАЛЬНЫХ ОБЩЕОБРАЗОВАТЕЛЬНЫХ УЧРЕЖДЕНИЙ”
“ЧИСЛЕННОСТЬ ОБУЧАЮЩИХСЯ ОБЩЕОБРАЗОВАТЕЛЬНЫХ ОРГАНИЗАЦИЙ КРАЯ ОСВАИВАЮЩИХ ОБРАЗОВАТЕЛЬНЫЕ ПРОГРАММЫ ОБЩЕГО ОБРАЗОВАНИЯ”
“ОБУЧАЮЩИХСЯ ПО ОБРАЗОВАТЕЛЬНЫМ ПРОГРАММАМ ОСНОВНОГО ОБЩЕГО ОБРАЗОВАНИЯ”
“УВЕЛИЧЕНИЕ ДОЛИ ВЫПУСКНИКОВ ОБЩЕОБРАЗОВАТЕЛЬНЫХ УЧРЕЖДЕНИЙ ОСВОИВШИХ ОСНОВНЫЕ ОБЩЕОБРАЗОВАТЕЛЬНЫЕ ПРОГРАММЫ ОСНОВНОГО ОБЩЕГО ОБРАЗОВАНИЯ СДАВШИХ ОСНОВНОЙ ГОСУДАРСТВЕННЫЙ ЭКЗАМЕН И ПОЛУЧИВШИХ АТТЕСТАТЫ”
“УДЕЛЬНЫЙ ВЕС ВЫПУСКНИКОВ КЛАССОВ ПОЛУЧИВШИХ АТТЕСТАТ О СРЕДНЕМ ОБЩЕМ ОБРАЗОВАНИИ В ОБЩЕМ ЧИСЛЕ ВЫПУСКНИКОВ КЛАССОВ”
“УДЕЛЬНЫЙ ВЕС ЧИСЛЕННОСТИ ВЫПУСКНИКОВ ОСВОИВШИХ ОБРАЗОВАТЕЛЬНЫЕ ПРОГРАММЫ СРЕДНЕГО ОБЩЕГО ОБРАЗОВАНИЯ ПОЛУЧИВШИХ КОЛИЧЕСТВО БАЛЛОВ ПО ЕГЭ НИЖЕ МИНИМАЛЬНОГО В ОБЩЕЙ ЧИСЛЕННОСТИ ВЫПУСКНИКОВ ОСВОИВШИХ ПРОГРАММЫ ОБЩЕГО ОБРАЗОВАНИЯ СДАВАВШИХ ЕГЭ”
“ЧИСЛО ОБУЧАЮЩИХСЯ ЗАВЕРШИВШИХ ОБУЧЕНИЕ ПО ОБЩЕОБРАЗОВАТЕЛЬНЫМ ПРОГРАММАМ ОСНОВНОГО ОБЩЕГО ОБРАЗОВАНИЯ ПОДЛЕЖАЩИХ ГОСУДАРСТВЕННОЙ ИТОГОВОЙ АТТЕСТАЦИИ”

В результате весь объем данных разложен на классы, а также выделены дополнительные классы, не выявленные на первом этапе кластеризации (см. Таблица 4.8).

Таблица 4.8. Дополнительные кластеры, выявленные в ходе интерактивной кластеризации.

Класс, определенный в результате кластеризации	Кол-во
Аттестация персонала	14
Научная и творческая деятельность учащихся	22
Образовательные программы	10
Поддержка одарённых детей	7
Продолжение обучения на следующей ступени образования	8
Углубленное изучение предметов	23
Лагерь	13
Инновационная деятельность	13
Квалификация персонала	57
Обеспеченность питанием	107

Аналогично была проведена кластеризация для всех 11 тематик и в общей сложности кластеризовано 598 615 показателей.

4.5. Оценка эффективности работы метода в проведенном эксперименте

Для оценки эффективности использования предложенного метода проведено сравнение с полностью ручной обработкой и обработкой с использованием библиотеки тематической кластеризации из пакета sklearn. В тематическом анализе используется скрытое размещение Дирихле (LDA, Latent Dirichlet allocation) в результате чего формируются тематики из групп ключевых слов и проводится соотнесение показателей с этими группами.

Замеры времени обработки проводились на кластеризации 100 показателей на 5 классов, с предположением, что в среднем автоматически алгоритм кластеризации способен определить достоверно 4 класса и 25% показателей попадают в “мусорный” кластер. При полностью ручной обработке показателей экспертом временные затраты на кластеризацию 100 показателей складываются из следующих видов работ: по 1 минуте на знакомство с каждыми 10-ю

показателями (используется термин “знакомство”, а не “прочтение”, т.к. эксперту требуется проанализировать смысл данного показателя); 30 минут на анализ информации, определение классов и критериев соотнесения к ним; 5 минут на разметку 100 показателей по определенным классам. В итоге ручная обработка 100 показателей занимает 45 минут. При этом эксперту требуется повторить операцию большое число раз, размечая новые показатели по ранее определенным классам, определение новых классов и возможно даже перегруппировку классов. Данную активность можно сравнить с интуитивной реализацией алгоритма агломеративной кластеризации. При большом числе объектов эксперту в памяти придется держать большое число классов и правил соотнесения, что на первом этапе приведет к росту числа технических ошибок соотнесения, а при тысячах объектов потребует от эксперта фиксации знаний на бумаге или в электронной таблице и написании обработчиков упрощающих соотнесения показателей с классами, т.е. только ручная обработка перестанет быть возможной.

При использовании для автоматизации работы по кластеризации метода тематической кластеризации временные затраты складываются из следующих видов работ: знакомство с тематиками определенными алгоритмом (прочтение наборов ключевых слов по тематикам), определение классов на основе полученных результатов и правил соотнесения – 15 минут; ручная корректировка 35-40% неверно соотнесенных показателей – 15-20 минут. Итого время, затрачиваемое на обработку с использованием вспомогательного инструмента в виде тематического анализа, равняется 30-35 минутам. Причины низкой эффективности методов тематического анализа при кластеризации коротких текстов были рассмотрены в первой главе. В случае со 100 показателями просмотр списков ключевых слов почти полностью эквивалентен прочтению всех показателей экспертом. При увеличении объемов выборки для кластеризации будут возникать проблемы аналогичные случаю с ручной обработкой, т.к. правила соотнесения с классами и правила корректировки находятся в голове эксперта и не формализованы.

В случае кластеризации предложенным методом затраты складываются из следующих видов работ: знакомство с результатами первичной кластеризации (просмотр 5-10% примеров из каждого класса) – 5 минут; анализ полученных классов - 5 минут; внесение 5-7 правил для корректировки результатов кластеризации. Итого время, затрачиваемое на кластеризацию показателей методом предлагаемым в данной диссертации составит 15-17 минут, что почти в три раза быстрее ручной обработки и в 2 раза быстрее метода с использованием тематического анализа. С учетом общих объемов кластеризуемых текстов, даже при линейной апроксимации экономия трудозатрат эксперта составила более 9 человеко-месяцев. При этом следует отметить, что при росте объемов выборки для кластеризации, временные затраты будут расти линейно, или даже суб-линейно, в то время как трудозатраты эксперта будут расти экспоненциально, т.к. возрастает число классов и вариантов распределения показателей между ними. Все введенные ограничения на каждой итерации кластеризации формализованы и учитываются на будущих этапах, таким образом, при росте объема выборки точность работы нейронной сети должна возрастать, что приведет к меньшему числу вводимых новых ограничений.

4.6. Скорость работы метода

В результате многократных прогонов сценариев кластеризации по последним двум экспериментам данной главы была получена статистика скорости работы метода на различных фазах. Время необходимое для дообучения языковой модели для корпусов Авито и Стратпланирование (0,5-1 миллионов текстов) на CPU составляет 15-30 суток, а на GPU (ресурс Google.Colab) - 12 часов.

Работа алгоритма кластеризации при первичном прогоне (варианты рабочих станций с CPU и GPU сравнимы) занимает 1-2 часа для выборки 10 000 текстов). Основное время занимает настройка части сети отвечающей за кластеризацию (обучение методом автоэнкодера).

Последующие прогоны на фазе интерактивной кластеризации занимают 2-5 минут.

4.7. Выводы по главе

В данной главе были представлены результаты ряда экспериментов, подтверждающие работоспособность и эффективность предлагаемых моделей и алгоритма. В частности была показана полнота предлагаемого алгоритма кластеризации предполагающая, что с помощью алгоритма кластеризации может быть получено произвольное разбиение исходного множества объектов на кластеры.

Пример с набором данных “Ирисы Фишера” и сравнение с основными алгоритмами кластеризации в эксперименте с объявлениями компании “Avito” показали преимущество предлагаемого алгоритма в плане точности результатов кластеризации. В то же время результаты даже нескольких шагов интерактивной кластеризации не уступают качеству алгоритмов классификации, что является важным результатом с учетом того, что современные наборы данных становятся большими, и полное обучение с учителем провести не представляется возможным в силу недоступности всех значений меток или высокой вычислительной сложности такого обучения.

В финале главы были представлены результаты и анализ эффективности работы предложенного алгоритма на наборе данных коротких текстов содержащих показатели эффективности системы стратегического планирования Российской Федерации. В данном эксперименте не представляется возможным оценить точность алгоритма, но снижение трудозатрат экспертов, оцениваемая более чем в 9 человеко-месяцев, позволяет сделать выводы об эффективности предложенного алгоритма для исследованных наборов данных.

ЗАКЛЮЧЕНИЕ

В результате выполнения работы получены следующие практические и научные результаты:

1. Проведено исследование моделей и методов машинного обучения для обработки текстов, и разработана архитектура искусственной нейронной сети реализующей кластеризацию на базе пространства признаков языковой модели русского языка.

2. Разработан метод обработки текстов для расширения словаря языковой модели на базе нейронной сети с использованием нечеткого иерархического классификатора, который позволяет повысить точность кластеризации в среднем на 10%.

3. Разработан метод для обработки обратной связи от эксперта, используемый для корректировки весовых коэффициентов нейронной сети, что позволяет проводить интерактивную кластеризацию наборов коротких текстов.

4. Сформулирован перечень этапов программы проведения испытаний метода нечеткой интерактивной кластеризации коротких текстов. Проведенные исследования по данной программе позволили установить границы применения предлагаемого метода. Метод наиболее эффективен для текстов с количеством слов от 10 до 100. В ходе проведенных исследований была достигнута средняя точность кластеризации 80%, при более низком числе дополнительных ограничений по сравнению с аналогичными методами. На разработанный программный модуль, использованный для проведения численных экспериментов получено свидетельство о государственной регистрации программы для ЭВМ № 2021615642.

5. Составлен алгоритм автоматизации работ по нечеткой интерактивной кластеризации коротких текстов в СППР “Федеральная Система Стратегического Планирования РФ (ФИС СП)”.

6. Проведена апробация разработанных модели, методов и алгоритма нечеткой интерактивной кластеризации коротких текстов в качестве элементов СППР “ФИС СП”. Внедрение алгоритма позволило решить задачу составления автоматического классификатора ключевых показателей эффективности документов стратегического планирования, при этом экономия трудозатрат эксперта оценивается более чем в 9 человеко-месяцев. Наличие автоматического классификатора позволяет снизить временные затраты экспертов при проведении процедуры проверки корректности классификации входящих (новых) документов.

7. Разработанный метод интерактивной кластеризации является универсальным и может быть применен для различных наборов коротких текстов. Предложенный метод может быть доработан для совместного использования с различными языковыми моделями и обобщен на случай совместной работы ряда экспертов.

Список сокращений и условных обозначений

API – прикладной программный интерфейс (Application Programming Interface)
BERT – архитектура языковой модели от компании Google (Bidirectional Encoder Representations from Transformers)
CNN – архитектура искусственной нейронной сети содержащей слои с операцией свертки (СНС, Convolutional Neural Network)
CPU – ядро процессора (Central Processing Unit)
DEC – алгоритм кластеризации на основе нейронной сети (Unsupervised Deep Embedding for Clustering Analysis)
ELMo – архитектура языковой модели предназначенная для получения сжатых векторных представлений (Embeddings from Language Models)
GPU – ядро графического процессора (Graphics Processing Unit)
IDE – программное окружения для разработки ПО (Integrated Development Environment)
LDA – Латентное размещение Дирихле (Latent Dirichlet allocation)
LSTM – архитектура искусственной нейронной сети с долгой краткосрочной памятью (Long short-term memory)
NER – задача обработки естественного языка, заключающаяся в распозновании именованных сущностей (Named Entity Recognition)
NLP – Обработка Естественного Языка (Natural Language Processing)
NLTK – один из ведущих пакетов программного обеспечения по обработке естественного языка
REST - архитектурный стиль взаимодействия компонентов распределённого приложения в сети (Representational State Transfer)
RNN - архитектура искусственной нейронной сети содержащей рекуррентные слои (РНС, Recurrent Neural Network)
RuBERT – языковая модель архитектуры BERT обученная на текстах на русском языке

SOM – самоорганизующиеся карты Кохонена (Kohonen self-organized maps)

SOTA – результат работы метода или решения демонстрирующий наилучшие результаты работы на момент исследования (state-of-the-art)

SQuAD – стандартный набор данных в области обработки естественного языка (The Stanford Question Answering Dataset)

t-SNE – алгоритм для сокращения размерности векторного пространства на основе стохастического разложения соседей (t-distributed Stochastic Neighbor Embedding)

UI – пользовательский интерфейс (User Interface)

ULMFiT – архитектура языковая модели из фреймворка FastAI (Universal Language Model Fine-tuning for Text Classification)

UNK – обозначение неизвестного токена в тексте (Unkown)

ГАС – Государственная Автоматизированная Система

КПЭ – ключевые показатели эффективности (KPI – Key Process Indicator)

МЭР – Министерство Экономического Развития

НИР – научно исследовательская работа

ОМСУ – орган местного самоуправления

ОС – операционная система

ПО – программное обеспечение

РОИВ – региональный орган исполнительной власти

РФ – Российская Федерация

СМЭВ – система межведомственного электронного взаимодействия

СП – стратегическое планирование

СППР – система поддержки принятия решений

ФИС – Федеральная Информационная Система

ФОИВ – федеральный орган исполнительной власти

СПИСОК ЛИТЕРАТУРЫ

1. Aggarwal C. C. Data Clustering Algorithms and Applications. / C. C. Aggarwal, C.K. Reddy // Chapman and Hall/CRC. – ISBN: 9781466558212. - 2014.
2. Alam M. A Review on Clustering of Web Search Result. / M. Alam, K. Sadaf // Advances in Computing and Information Technology. Advances in Intelligent Systems and Computing, Springer, Berlin, Heidelberg. / Meghanathan N., Nagamalai D., Chaki N. (eds) - 2013. – Vol. 177.
3. Aljalbout E. Clustering with Deep Learning: Taxonomy and New Methods. / E. Aljalbout, V. Golkov, Y. Siddiqui, M. Strobel, D. Cremers // arXiv:1801.07648, 2018.
4. Amigo E. A comparison of Extrinsic Clustering Evaluation Metrics based on Formal Constraints. / E. Amigo, J. Gonzalo, J.Verdejo // Departamento de Lenguajes y Sistemas Informaticos, UNED, Madrid, Spain. – 2009.
5. Amorim R. Feature Weighting for Clustering: Using K-Means and the Minkowski. // LAP Lambert Academic Publishing. – 2012.
6. Bae J. Interactive Clustering: A Comprehensive Review. / J. Bae, T. Helldin, M. Riveiro, S. Nowaczyk, M. Bouguella, G. Falkman // ACM Comput. Surv. -2020. - Vol. 53. - No. 1.
7. Bagherjeiran A. Adaptive clustering: obtaining better clusters using feedback and past experience. / A. Bagherjeiran, C. F. Eick, Chun-Sheng Chen, R. Vilalta // Fifth IEEE International Conference on Data Mining (ICDM'05), Houston, TX. – 2005. - P. 4. - DOI: 10.1109/ICDM.2005.17.
8. Baker C. F. The Berkeley FrameNet Project. / C.F. Baker, C.J. Fillmore, J.B. Lowe // Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics (ACL '98/COLING '98), Association for Computational Linguistics, USA. - 1998. – Vol. 1, - P. 86–90. - DOI: https://doi.org/10.3115/980845.980860.

9. Balcan M.F. Clustering with Interactive Feedback. / M.F. Balcan, A. Blum // Algorithmic Learning Theory, Lecture Notes in Computer Science, Springer, Berlin, Heidelberg. / Freund Y., Györfi L., Turán G., Zeugmann T. (eds). – Vol. 5254. - 2008.

10. Ball G.H. Isodata: a method of data analysis and pattern classification. / G.H. Ball, D.J. Hall // Stanford Research Institute, Menlo Park,United States, Office of Naval Re-search, Information Sciences Branch. - 1965.

11. Banerjee S. Clustering short texts using wikipedia. / S. Banerjee, K. Ramanathan, A. Gupta // SIGIR , ACM. / Wessel Kraaij; Arjen P. de Vries; Charles L. A. Clarke; Norbert Fuhr & Noriko Kando, ed. -2007. - P. 787-788.

12. Basu S. Assisting Users with Clustering Tasks by Combining Metric Learning and Classification. / S. Basu, D. Fisher, S.M. Drucker, H. Lu // Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence. - 2010.

13. Basu S. Constrained Clustering: Advances in Algorithms, Theory, and Applications. / S. Basu, I. Davidson, K. Wagstaff // CRC Press. - 2008.

14. Basu S. Semi-supervised Clustering by Seeding. / S. Basu, A. Banerjee, R. Mooney // In Proceedings of 19th International Conference on Machine Learning. - 2002.

15. Beltagy I. SciBERT: A Pretrained Language Model for Scientific Text / I. Beltagy, K. Lo, A. Cohan // Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Hong Kong, China : Association for Computational Linguistics. - 2019. - P. 3615-3620. - URL: https://www.aclweb.org/anthology/D19-1371.

16. Bengio Y. A neural probabilistic language model. / Y. Bengio, R. Ducharme, P. Vincent, C. Janvin // Learn, Res. 3 / J. Mach. – 2003. - P.1137–1155.

17. Blei D.M. Latent dirichlet allocation. / D.M. Blei, A.Y. Ng, M.I. Jordan // Learn. Res. 3. / J. Mach. – 2003. - P.993-1022.

18. Bradbury J. Quasi-Recurrent Neural Networks. / J. Bradbury, S. Merity, C. Xiong, R. Socher // ICLR. - 2017.

19. Burtsev M. DeepPavlov: Open-Source Library for Dialogue Systems. // Proceedings of ACL 2018, System Demonstrations. — 2018. —P. 122—127.

20. Campello R. Density-Based Clustering Based on Hierarchical Density Estimates. / R. Campello, D. Moulavi, J. Sander // Advances in Knowledge Discovery and Data Mining, Springer. – 2013.

21. Caruana R. Multitask Learning. // Learning to Learn, Springer, Boston, MA. / Thrun S., Pratt L. (eds). – 1998.

22. Chandrasekaran E. Fuzzy node fuzzy graph and its cluster analysis. / E. Chandrasekaran, N.Sathyaseelan. // International Journal of Engineering Research and Applications (IJERA). - 2012. - Vol. 2, Issue 3, May-Jun 2012. - P.733-738. - ISSN: 2248-9622.

23. Chen Y. Ant Spatial Clustering Based on Fuzzy IF-THEN Rule. / Y. Chen, M. Han, H. Zhu // Fuzzy Information and Engineering, Advances in Intelligent and Soft Computing Series. – 2010. – Vol. 78.

24. Chen Y. Ant Spatial Clustering Based on Fuzzy IF-THEN Rule. / Y. Chen, M. Han, H. Zhu // Fuzzy Information and Engineering, Advances in Intelligent and Soft Computing. – 2010. – Vol. 78. - P. 563-569.

25. Cohn D. Semi-supervised Clustering with User Feedback. / D. Cohn, R. Caruana, A. Mccallum // arXiv preprint. - 2008.

26. Collobert R. A Unified Architecture for Natural Language Processing: Deep Neural Networks with Multitask Learning. / R. Collobert, J. Weston // Proceedings of the 25th International Conference on Machine Learning, ACM, New York, NY, USA , - 2008. - P. 160-167.

27. Comaniciu D. Mean shift: A robust approach toward feature space analysis., / D. Comaniciu, P. Meer // IEEE Transactions on Pattern Analysis and Machine Intelligence. - 2002.

28. Conneau A. Supervised Learning of Universal Sentence Representations from Natural Language Inference Data. / A. Conneau, D. Kiela, H. Schwenk, L. Barrault, A. Bordes // Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, Copenhagen, Denmark. - 2017.

29. Dai A.M. Semi-supervised Sequence Learning. / A.M. Dai, Q.V. Le // Proceedings of the 28th International Conference on Neural Information Processing Systems (NIPS'15), MIT Press, Cambridge, MA, USA. – 2015. – Vol. 2. - P. 3079–3087. - URL: http://papers.nips.cc/paper/5949-semi-supervised-sequence-learning.pdf.

30. Dasgupta S. Which Clustering Do You Want? Inducing Your Ideal Clustering with Minimal Feedback. / S. Dasgupta, V. Ng // arXiv:1401.5389. - 2014. – URL: https://arxiv.org/abs/1401.5389.

31. Demiriz A. A Genetic Algorithm Approach for Semi-Supervised Clustering. / A. Demiriz, K.P. Bennett, M.J. Embrechts // International Journal of Smart Engineering System Design. - 2002. - Vol. 4.

32. Devlin J. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, / J. Devlin, M.W. Chang, K. Lee, K. Toutanova // arXiv preprint arXiv:1810.04805. - 2018.

33. Dizaji K.G. Deep Clustering via Joint Convolutional Autoencoder Embedding and Relative Entropy Minimization. / K.G. Dizaji, A. Herandi, C. Deng, W. Cai, H. Huang // IEEE International Conference on Computer Vision (ICCV), Venice. - 2017.

34. Dong Y. A hierarchical clustering algorithm based on fuzzy graph connectedness. / Y. Dong, Y. Zhuang, K. Chen, X. Tai. // Fuzzy Sets and Systems. - 2006. – Vol. 157, Issue 13. - P. 1760-1774. – ISSN: 0165-0114.

35. Dudarin P.V. A Technique to Pre-trained Neural Network Language Model Customization to Software Development Domain. / P.V. Dudarin, V.G. Tronin, K.V. Svyatov // Artificial Intelligence (RCAI 2019), Communications in Computer and Information Science, Springer, Cham. / Kuznetsov S., Panov A. (eds). – 2019. - Vol 1093.

36. Dudarin P.V. An Approach to Fuzzy Hierarchical Clustering of Short Text Fragments Based on Fuzzy Graph Clustering. / P.V. Dudarin, N.G. Yarushkina // Proceedings of the Second International Scientific Conference “Intelligent Information Technologies for Industry” (IITI’17), Advances in Intelligent Systems and Computing, Springer. Cham. - 2018. - Vol 679.

37. Ester M. A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. / M. Ester, H. P. Kriegel, J. Sander, X. Xu // Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining, Portland, OR, AAAI Press. – 1996. – P. 226–231.

38. Fatehi K. Improving semi-supervised constrained k-means clustering method using user feedback. / K. Fatehi, A. Bozorgi, M.S. Zahedi, E. Asgarian // Journal of Computing and Security. – 2014. - Vol 1, num. 4.

39. Frey B.J. Clustering by Passing Messages Between Data Points. / B.J. Frey, Delbert D. // Science Feb. - 2007.

40. Gabrilovich E. Wikipedia-based Semantic Interpretation for Natural Language Processing. / E. Gabrilovich, S. Markovitch // Journal of Artificial Intelligence Research (JAIR) 34. – 2009. – P. 443-498.

41. Gath I. Unsupervised Optimal Fuzzy Clustering. / I. Gath, A.B. Geva // IEEE Transactions on Pattern Analysis and Machine Intelligence. -1989. - Vol. 11, no. 7. - P. 773-781.

42. Graves A. Hybrid speech recognition with Deep Bidirectional LSTM. / A. Graves, N. Jaitly, M. Rahman. // ASRU. – 2013.

43. Greff K. Neural Expectation Maximization. / K. Greff, S. van Steenkiste, J. Schmidhuber // Advances in Neural Information Processing Systems 30. - 2017.

44. Han X. A novel machine learning approach to rank web forum posts. / X. Han, J. Ma, Y. Wu, C. Cui. // Soft Computing. – 2014. – Vol. 18, Issue 5. - P. 941–959.

45. Hastie T. The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Second Edition. / T. Hastie, R. Tibshirani, J. Friedman // Springer Series in Statistics book series. - 2009.

46. Hinton G. Distilling the knowledge in a neural network / G. Hinton, O. Vinyals, J. Dean // arXiv preprint arXiv:1503.02531. – 2015.

47. Hochreiter S. Long short-term memory. / S. Hochreiter, J. Schmidhuber // Neural computation 9 (8). – 1997. - P. 1735-1780.

48. Hoffer E. Deep Metric Learning Using Triplet Network. / E. Hoffer, N. Ailon // Similarity-Based Pattern Recognition, Lecture Notes in Computer Science, Springer, Cham. / Feragen A., Pelillo M., Loog M. (eds). -2015. - Vol 9370.

49. Hou D., Gu Y. An Efficient Successive Iteration Partial Cluster Algorithm for Large datasets. / D. Hou, Y. Gu // Fuzzy Information and Engineering, Advances in Intelligent and Soft Computing. -2010. - Vol 78. - P. 557-562.

50. Hovy E. OntoNotes: the 90% solution. / E. Hovy, M. Marcus, M. Palmer, L. Ramshaw, R. Weischedel // In Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers (NAACL-Short '06). Association for Computational Linguistics, USA. – P. 57–60.

51. Howard J. Fastai. // - 2021. URL: https://github.com/fastai/fastai.

52. Howard J. Universal Language Model Fine-tuning for Text Classification / J. Howard, S. Ruder // Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). — Melbourne, Australia : Association for Computational Linguistics, 07/2018. — P. 328—339. — URL: https://www.aclweb.org/anthology/P18-1031

53. Huang Y. Mixed-Iterative Clustering. // PhD thesis at Language Technologies Institute School of Computer Science Carnegie Mellon University Pittsburgh, PA 15213. - 2010.

54. Huang Y. Text clustering with extended user feedback. / Y. Huang, T.M. Mitchell // Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Seattle, Washington, USA. - 2006.

55. Hubert L. Comparing partitions. / L. Hubert, P. Arabie // Journal of Classification. – 1985. – Vol. 2, №1. - P. 193-218. DOI:10.1007/BF01908075.

56. Jain A.K. Algorithms for Clustering Data. / A.K. Jain, C.R. Dubes // Pretice Hall, Engelwood, N.J., 07632. - 1988.

57. Jain A.K. Data Clustering: 50 Years Beyond K-Means // Pattern Recognition Letters. -2009. – Vol. 31(8). – P. 651-666. - DOI: 10.1016/j.patrec.2009.09.011.

58. Jain A.K. Data Clustering: A Review. / A.K. Jain, M.N. Murty, P.J. Flynn // ACM Computing Surveys (CSUR), USD. -1999. – Vol.31, Issue 3. - P. 264-323.

59. Jolliffe, I. T. Principal Component Analysis. Springer, Verlag. – 1986. - P. 487. - DOI:10.1007/b98835. - ISBN 978-0-387-95442-4.

60. Joshi M. BERT for Coreference Resolution: Baselines and Analysis / M. Joshi, O. Levy, L. Zettlemoyer, D. Weld // Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Hong Kong, China : Association for Computational Linguistics. - 2019. – P. 5803–5808. – URL: https://www.aclweb.org/anthology/D19-1588.

61. Joshi M. SpanBERT: Improving Pre-training by Representing and Predicting Spans / M. Joshi // Transactions of the Association for Computational Linguistics. – 2020. – Vol. 8. – P. 64–77. – URL: https://transacl.org/ojs/ index.php/tacl/article/view/1853.

62. 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. – 2014. – Vol. 1.

63. Kannan A. Automated Response Suggestion for Email. / A. Kannan, K. Kurach, S. Ravi, T. Kaufmann, A. Tomkins, B. Miklos, G. Corrado, L. Lukacs, M. Ganea, P. Young, V. Ramavajjala, S. Reply. // KDD, arXiv:1606.04870. – 2016.

64. Kapil S. On K-means data clustering algorithm with genetic algorithm. / S. Kapil, M. Chawla, M.D. Ansari // Fourth International Conference on Parallel, Distributed and Grid Computing (PDGC), Waknaghat. - 2016.

65. Kneser R. Improved backing-off for M-gram language modeling. / R. Kneser, H. Ney // IEEE Computer Society. – 1995. - P. 181-184.

66. Kuratov Y. Adaptation of deep bidirectional multilingual transformers for russian language / Y. Kuratov, M. Arkhipov // Computational Linguistics and Intellectual Technologies. International Conference "Dialogue 2019" Proceedings. – 2019. – P. 333–339.

67. Kutuzov A. Texts in, meaning out: neural language models in semantic similarity task for Russian. / A. Kutuzov, I. Andreev // Proceedings of the Dialog 2015 Conference, Moscow, Russia. - 2015.

68. Lafferty J.D. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. / J.D. Lafferty, A. McCallum, F.C.N. Pereira // Proceedings of the Eighteenth International Conference on Machine Learning (ICML '01), Morgan Kaufmann Publishers Inc., San Francisco, CA, USA. – 2001. – P. 282–289.

69. Lample G. Neural Architectures for Named Entity Recognition / G. Lample, M. Ballesteros, S. Subramanian, K. Kawakami, C. Dyer // Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics, San Diego, California. – 2016. - P. 260–270.

70. Le Q. Distributed Representations of Sentences and Documents. / Q. Le, T. Mikolov // Proceedings of the 31st International Conference on Machine Learning, PMLR 32(2). – 2014. – P. 1188-1196.

71. Leela V. Comparative Study of Clustering Techniques in Iris Data Sets. / V. Leela, K. Sakthipriya, R. Manikandan // World Applied Sciences Journal 29 (Data Mining and Soft Computing Techniques), - 2014.

72. Li J. Chameleon based on clustering feature tree and its application in customer segmentation, / J. Li, K. Wang, L. Xu // Ann Oper Res. – 2009. - P. 168-225. – DOI:doi.org/10.1007/s10479-008-0368-4.

73. Li L. Deep Clustering with Gated Convolutional Networks. / L. Li, H. Kameoka // IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Calgary. - 2018.

74. Manning C.D. An Introduction to Information Retrieval. / C.D. Manning, P. Raghavan, H. Schütze // Cambridge University Press, Cambridge, England. – 2009.

75. Mansoori E.G. GACH: a grid based algorithm for hierarchical clustering of high-dimensional data. // Soft Computing. - 2014. – Vol. 18, Issue 5. – P. 905-922.

76. McCann B. Learned in Translation: Contextualized Word Vectors. / B. McCann, J. Bradbury, C. Xiong, R. Socher // NIPS / I. Guyon, U. von Luxburg, S. Bengio, H.M. Wallach, R.Fergus, S.V.N. Vishwanathan, R. Garnett (eds). - 2017. - P. 6297-6308.

77. McInnes L. HDBScan: Hierarchical density based clustering / L. McInnes, J. Healy, S. Astels // Journal of Open Source Software, The Open Journal. - 2017. –Vol. 2, num.11.

78. Meier B.B. Learning Neural Models for End-to-End Clustering. / B.B. Meier, I. Elezi, M. Amirian, O. Dürr, T. Stadelmann // Artificial Neural Networks in Pattern Recognition, Lecture Notes in Computer Science, Springer, Cham. / L. Pancioni, F. Schwenker, E. Trentin (eds.). -2018. - Vol 11081.

79. Mikolov T. Distributed representations of words and phrases and their compositionality. / T. Mikolov, I. Sutskever, K. Chen, G. Corrado, J. Dean, // Proceedings of the 26th International Conference on Neural Information Processing Systems, Lake Tahoe, Nevada. – 2013. - P. 3111-3119.

80. Mikolov T. Recurrent neural network based language model. / T. Mikolov, M. Karafiát, L. Burget, J. Černocký, S. Khudanpur // INTERSPEECH. -2010. - P. 1045-1048.

81. Modha, D.S. Feature Weighting in k-Means Clustering. / D.S. Modha, W.S. Spangler // Machine Learning. – 2003. – P. 52-217. DOI: doi.org/10.1023/A:1024016609528.

82. Nebu C.M. Semi-supervised clustering with soft labels. / C.M. Nebu, S. Joseph. // International Conference on Control Communication & Computing India (ICCC), Trivandrum. - 2015.

83. Nebu C.M., Joseph S. Semi-supervised clustering with soft labels. / C. M. Nebu, S. Joseph // International Conference on Control Communication & Computing India (ICCC), Trivandrum. -2015. - P. 612-616. - doi: 10.1109/ICCC.2015.7432969.

84. Nivre J. Universal Dependencies v1: A Multilingual Treebank Collection. / J. Nivre, M.C. de Marneffe, F. Ginter, Y. Goldberg, J. Hajič, C.D. Manning, R. McDonald, S. Petrov, S. Pyysalo, N. Silveira, R. Tsarfaty, D. Zeman // Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16), Portorož, Slovenia. -2016. - P. 1659–1666.

85. Novák V. A general methodology for managerial decision making using intelligent techniques. / V. Novák, I. Perfilieva, N.G. Jarushkina // Chapter Recent Advances in Decision Making, Series Studies in Computational Intelligence. - 2009. - Vol., 222. - P. 103-120.

86. Pedregosa F. Scikit-learn: Machine Learning in Python. / F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, É. Duchesnay // Journal of Machine Learning Research. – 2011. - Vol. 12.

87. Pedrycz W. Algorithms of fuzzy clustering with partial supervision. // Pattern Recognition Letters. – 1985. - Vol 3.

88. Peters M. Deep contextualized word representations. / M. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, L. Zettlemoyer // Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, New Orleans, Louisiana. - 2018. – Vol. 1. arXiv:1802.05365.

89. Ramachandran P. Unsupervised Pretraining for Sequence to Sequence Learning. / P. Ramachandran, P. Liu, Q. Le // Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, Copenhagen, Denmark. - 2017.

90. Raymond T.Y. Fuzzy relation, fuzzy graphs and their applications to clustering analysis. / T.Y. Raymond, S.Y. Bang // Fuzzy Sets and their Applications to Cogni-

tive and Decision Processes, Academic Press. – 1975. - P. 125-149. – ISBN: 9780127752600.

91. Rokach L. Clustering Methods. / L. Rokach, O. Maimon // Data Mining and Knowledge Discovery Handbook, Springer, Boston, MA. / O. Maimon, L. Rokach (eds). - 2005.

92. Rosenfeld A. Fuzzy graphs. // Fuzzy Sets and Their Applications to Cognitive and Decision Processes, Academic Press, New York. / L.A. Zadeh, K.S. Fu, K. Tanaka, M. Shimura (eds.). – 1975. - P. 77–95.

93. Rousseeuw P.J. Silhouettes: a Graphical Aid to the Interpretation and Validation of Cluster Analysis, Computational and Applied Mathematics. – 1987. - P. 53–65. – DOI: 10.1016/0377-0427(87)90125-7.

94. Ruder S. A Survey Of Cross-lingual Word Embedding Models. / S. Ruder, I. Vulić, A. Søgaard // Journal of Artificial Intelligence Research. – 2019.

95. Ruspini E.H. A new approach to clustering. // Inform. and Control. – 1969. – Vol. 15(1). – P. 22–32.

96. Sameena K. Clustering Using Strong Arcs in Fuzzy Graphs. // Gen. Math. Notes. - 2015. - Vol. 30. - P. 60-68. – ISSN: 2219-7184.

97. Sandeep Narayan K.R. Connectivity in a Fuzzy Graph and its Complement. / K.R. Sandeep Narayan, M.S. Sunitha // Gen. Math. Notes. - 2012. - Vol. 9, No. 1, March 2012. -P.38-43. – ISSN: 2219-7184.

98. Shavrina T. To the methodology of corpus construction for machine learning: “Taiga” syntax tree corpus and parser. / T. Shavrina, O. Shapovalova // КОРПУСНАЯ ЛИНГВИСТИКА. - 2017. - P. 78-84.

99. Suresh T. LSTM Model for Semantic clustering of user-generated content using AI Geared to wearable Device. / T. Suresh, K.T. Meena Abarna // Semanticscholar.org Corpus ID: 212585860. - 2017.

100. Sutskever I. On the importance of initialization and momentum in deep learning. / I. Sutskever, J. Martens, G. Dahl, G. Hinton // Proceedings of the 30th International Conference on Machine Learning, PMLR 28(3). – 2013. – P. 1139-1147.

101. Toldova S. Coreference Resolution for Russian: The Impact of Semantic Features / S. Toldova, I. Maxim // Computational Linguistics and Intellectual Technologies. International Conference" Dialogue 2017" Proceedings. – 2017. – P. 339–349.
102. Torra V. Fuzzy c-means for fuzzy hierarchical clustering. // Proc. FUZZ-IEEE. – 2005. - P. 646-651.
103. Vaswani A. Attention is All you Need. // Advances in Neural Information Processing Systems 30, Curran Associates, Inc. – 2017. - P. 5998-6008.
104. Vincent D. Fast unfolding of communities in large networks. / D. Vincent J.L. Guillaume, R. Lambiotte, E. Lefebvre. // J. Stat. Mech. - 2008.
105. Wang A. Bert has a mouth, and it must speak: Bert as a markov random field language model / A. Wang, K. Cho // arXiv preprint arXiv:1902.04094. – 2019.
106. Wang Z. Semi-supervised Clustering for Short Text via Deep Representation Learning / Z. Wang, H. Mi, A. Ittycheriah // Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning, Association for Computational Linguistics, Berlin, Germany. - 2016.
107. Winkler R. Problems of Fuzzy c-Means Clustering and Similar Algorithms with High Dimensional Data Sets. / R. Winkler, F. Klawonn, R. Kruse // Challenges at the Interface of Data Analysis, Computer Science and Optimization, Studies in Classification, Data Analysis, and Knowledge Organization. Springer, Berlin, Heidelberg. / W. Gaul, A. Geyer-Schulz, L. Schmidt-Thieme, J. Kunze (eds). – 2012.
108. Xie J. Unsupervised deep embedding for clustering analysis / J. Xie, R. Girshick, A. Farhadi // ICML'16: Proceedings of the 33rd International Conference on International Conference on Machine Learning. - 2002.
109. Xu J. Self-Taught Convolutional Neural Networks for Short Text Clustering / J. Xu, B. Xu, P. Wang, S. Zheng, G. Tian, J. Zhao // IEEE Neural Networks. – 2017. – Vol. 88.
110. Yang B. Towards K-means-friendly spaces: Simultaneous deep learning and clustering. / B. Yang, X. Fu, N.D. Sidiropoulos, M. Hong // Proceedings of the 34th International Conference on Machine Learning. – 2017. – Vol. 70.

111. Yang C. I Know You'll Be Back: Interpretable New User Clustering and Churn Prediction on a Mobile Social Application. / C. Yang, X. Shi, L. Jie, J. Han // The 24th ACM SIGKDD International Conference. - 2018.

112. Yang J. Joint Unsupervised Learning of Deep Representations and Image Clusters. / J. Yang, D. Parikh, D. Batra // IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV. - 2016.

113. Zhang J. A hybrid clustering algorithm based on PSO with dynamic crossover. / J. Zhang, Y. Wang, J. Feng // Soft Computing. - 2014. – Vol. 18, Issue 5. – P. 961–979.

114. Zhang T. BIRCH: an efficient data clustering method for very large databases. / T. Zhang, R. Ramakrishnan, M. Livny // Proceedings of the 1996 ACM SIGMOD international conference on Management of data (SIGMOD '96). - 1996, - P. 103–114. - DOI:10.1145/233269.233324.

115. Воронцов К.В. Вероятностное тематическое моделирование: теория, модели, алгоритм и проект BigARTM. / М: МФТИ, Федеральный исследовательский центр “Информатика и управление”. - 2021.

116. Гречачин В. А. К вопросу о токенизации текста. // Международный научно-исследовательский журнал. - 2016. - № 6 (48) Часть 4. - С. 25−27.

117. Дударин П.В. Алгоритм построения иерархического классификатора коротких текстовых фрагментов на основе кластеризации нечеткого графа. / П.В. Дударин, Н.Г. Ярушкина // Радиотехника, Москва, - 2017. - № 6.

118. Дударин П.В. Методика и алгоритм кластеризации объектов экономической аналитики. / П.В. Дударин, А.П. Пинков, Н.Г. Ярушкина // Автоматизация процессов управления, НПО Марс. - 2017. - № 1.

119. Дударин П.В. Подход к оценке трудоемкости задач в процессе разработки программного обеспечения на основе нейронных сетей. / П.В. Дударин, В.Г. Тронин, К.В. Святов, В.А. Белов, Р.А. Шакуров // Автоматизация процессов управления, НПО Марс. – 2019. - № 3.

120. Дударин П.В. Подход к трансформации кластерного дерева признаков в векторное пространство признаков. / П.В. Дударин, Н.Г. Ярушкина // Радиотехника, Москва. – 2018. - № 6.

121. Официальный сайт Министерства экономического развития Российской Федерации // - URL: http://economy.gov.ru/minec/activity/sections/strategicPlanning/ (дата обращения: 23.02.2021)

122. Официальный сайт Федеральной службы государственной статистики // URL: http://www.gks.ru/wps/wcm/connect/rosstat_main/rosstat/ru/statistics/databases/emiss/ (дата обращения: 02.05.2017)

123. Павлов А.Н. Методы обработки экспертной информации: Учебно-методическое пособие. / А.Н. Павлов, Б.В. Соколов // СПб.: ГУАП. - 2005.

124. Сибирёв И.В. Индексы оценки результатов кластеризации // Нечеткие системы, мягкие вычисления и интеллектуальные технологии, Труды VII всероссийской научно-практической конференции, Санкт-Петербург. – 2017. - том 1.

125. Славнов К.А. Анализ социальных графов. // - 2015. - URL: http://www.machinelearning.ru/wiki/images/6/60/2015_417_SlavnovKA.pdf (дата обращения: 02.05.2017)

126. Федеральный закон “О стратегическом планировании в Российской Федерации” № 172-ФЗ от 28.07.2014 г. // - URL: http://pravo.gov.ru/proxy/ips/?docbody=&nd=102354386 (дата обращения: 02.05.2018)

127. Шелехова Н.В. Информационные технологии в аналитическом контроле качества алкогольной продукции. / Н.В. Шелехова, В.А. Поляков, Е.М. Серба, Т.М. Шелехова, О.В. Веселовская, Л.И. Скворцова // Пищевая промышленность, Москва. – 2018. - № 8.

128. Шелехова Н.В. Управление технологическими процессами производства алкогольной продукции с применением информационных технологий. / Н.В.

Шелехова, Л.В. Римарева // Хранение и переработка сельхозсырья, Пищевая промышленность, Москва. – 2017. № 3.

ПРИЛОЖЕНИЕ 1. Акт и справки о внедрении результатов диссертационной работы

IBS УМНЫЙ ВЫБОР МЕНЯЮЩИХСЯ ТЕХНОЛОГИЙ

ООО «ИБС Экспертиза»
ОГРН 1067761849704, ИНН/КПП 7713606622/771301001
Россия, 127434, Москва, Дмитровское шоссе, дом 9Б, этаж 5, помещение XIII, комната 6
телефон/факс: +7 (495) 967 80 80
ibs@ibs.ru, www.ibs.ru

18.02.2021 № 0324-09

Справка
о внедрении результатов
кандидатской диссертационной работы

Выдана для предъявления в диссертационном совете Д 212.277.04 Федерального государственного бюджетного образовательного учреждения высшего образования «Ульяновский государственный технический университет» г. Ульяновск.

Результаты диссертационной работы Дударина Павла Владимировича "Исследование и разработка моделей и методов нечеткой кластеризации коротких текстов", представленной на соискание ученой степени кандидата технических наук, а именно: модель нечеткой кластеризации коротких текстов, методы расширения словаря языковой модели и корректировки весов нейронной сети для учета обратной связи эксперта в интерактивной кластеризации, а также программная реализация метода нечеткой интерактивной кластеризации на языке Python были использованы в рамках НИР в интересах Министерства экономического развития РФ по теме: «Разработка рекомендаций по совершенствованию информационного обеспечения участников стратегического планирования в части осуществления мониторинга и контроля реализации документов стратегического планирования с использованием федеральной информационной системы стратегического планирования (ФИС СП)», шифр темы 0101-01-17.

Внедрение позволило провести анализ набора данных ключевых показателей эффективности и подготовить предложения для формирования классификатора показателей федеральной информационной системы стратегического планирования (ФИС СП).

Директор проекта: Л.А. Либкин

Генеральный директор: Г.О. Кочаров

ИБС Экспертиза

IBS
УМНЫЙ ВЫБОР МЕНЯЮЩИХСЯ ТЕХНОЛОГИЙ

ООО «ИБС Экспертиза»
ОГРН 1067761849704, ИНН/КПП 7713606622/771301001
Россия, 127434, Москва, Дмитровское шоссе, дом 9Б,
этаж 5, помещение XIII, комната 6
телефон/факс: +7 (495) 967 80 80
ibs@ibs.ru, www.ibs.ru

18.02.2021 № 0323-09

Справка
о внедрении результатов
кандидатской диссертационной работы

Выдана для предъявления в диссертационном совете Д 212.277.04 Федерального государственного бюджетного образовательного учреждения высшего образования «Ульяновский государственный технический университет» г. Ульяновск.

Результаты диссертационной работы Дударина Павла Владимировича "Исследование и разработка моделей и методов нечеткой кластеризации коротких текстов", представленной на соискание ученой степени кандидата технических наук, а именно: модель нечеткой кластеризации коротких текстов, методы расширения словаря языковой модели и корректировки весов нейронной сети для учета обратной связи эксперта в интерактивной кластеризации, а также программная реализация метода нечеткой интерактивной кластеризации на языке Python были использованы в рамках НИР в интересах Министерства экономического развития РФ по теме: «Разработка методического обеспечения интеллектуальной системы проверки уведомления об утверждении (одобрении) документа стратегического планирования или внесении в него изменений при ведении федерального государственного реестра документов стратегического планирования Федеральной информационной системы стратегического планирования (ФИС СП)», шифр темы 0103-01-18.

Внедрение результатов работы позволило повысить эффективности и уменьшить нагрузку на участников Стратегического планирования при работе в федеральной информационной системе стратегического планирования (ФИС СП).

Директор проекта: Л.А. Либкин

Генеральный директор: Г.О. Кочаров

«ИБС Экспертиза» ИНН 7713606622 ОГРН 1067761849704 МОСКВА

IBS

УМНЫЙ ВЫБОР МЕНЯЮЩИХСЯ ТЕХНОЛОГИЙ

ООО «ИБС Экспертиза»
ОГРН 1067761849704, ИНН/ КПП 7713606622 / 771301001
Россия, 127434, Москва, Дмитровское шоссе, дом 9Б, этаж 5, помещение XIII, комната 6
телефон/факс: +7 (495) 957 80 80
ibs@ibs.ru, www.ibs.ru

УТВЕРЖДАЮ
Генеральный директор
ООО "ИБС Экспертиза"
Кочаров Г. О.

«ИБС Экспертиза»

Акт

о внедрении результатов

кандидатской диссертационной работы

Комиссия в составе:

Председатель – **Александрова Елена Владимировна,** директор отделения собственных платформ,

члены комиссии:

Эйделанд Павел Вадимович – директор дивизиона разработки и тестирования

Иванова Екатерина Михайловна – начальник отдела развития продуктов и решений,

составили настоящий акт о том, что результаты диссертационной работы **Дударина Павла Владимировича** "Исследование и разработка моделей и методов нечеткой кластеризации коротких текстов", представленной на соискание ученой степени кандидата технических наук,
а именно: архитектура искусственной нейронной сети и алгоритм нечеткой кластеризации коротких текстов, методы расширения словаря языковой модели и корректировки весов нейронной сети для учета обратной связи эксперта в интерактивной кластеризации, а также программная реализация метода нечеткой интерактивной кластеризации на языке Python внедрены в системе **Планета.Аналитика 4.0** компании **ООО "ИБС Экспертиза".**

Внедрение позволило заменить стандартный модуль кластеризации текстов на базе метрики TF-IDF и алгоритма k-means более производительным методом в части работы с наборами коротких текстов.

Председатель комиссии: Александрова Е.В.

Члены комиссии: Иванова Е.М.

Эйделанд П.В.

ПРИЛОЖЕНИЕ 2. Свидетельство о государственной регистрации программы для ЭВМ

РОССИЙСКАЯ ФЕДЕРАЦИЯ

СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2021615642

Программная система кластеризации коротких текстов

Правообладатель: *федеральное государственное бюджетное образовательное учреждение высшего образования «Ульяновский государственный технический университет» (RU)*

Авторы: *Ярушкина Надежда Глебовна (RU), Дударин Павел Владимирович (RU)*

Заявка № **2021614670**
Дата поступления **02 апреля 2021 г.**
Дата государственной регистрации
в Реестре программ для ЭВМ ***09 апреля 2021 г.***

Руководитель Федеральной службы по интеллектуальной собственности

Г.П. Ивлиев

ПРИЛОЖЕНИЕ 3. Результаты кластеризации КПЭ СП

Общее образование	
Тематическая группа	Кол-во
Повышение квалификации персонала	167
Научные, творческие и спортивные мероприятия	480
Технологическая оснащённость	302
Расширение сети образовательных учреждений	74
Государственные стандарты	1 079
Численность персонала	334
Инклюзивное образование	294
Единый государственный экзамен	998
Заработная плата персонала	617
Образовательной инициативы «Наша новая школа»	44
Оснащенность образовательных учреждений	155
Расход бюджета на обучающихся	37
Доступность образовательных услуг	980
Аттестация учащихся	466
Здоровье обучающихся	86
Ремонт образовательных учреждений	176
Удовлетворённость образовательными услугами	185
Эффективность образовательной деятельности	38
Оснащённость образовательных учреждений	31
Центры прикладной квалификации	7
Безопасность	20

Эффективность заключения контрактов с персоналом	9
Физическая культура и спорт	64
Аттестация персонала	14
Научная и творческая деятельность учащихся	22
Образовательные программы	12
Поддержка одарённых детей	9
Продолжение обучения на следующей ступени образования	10
Углубленное изучение предметов	28
Лагерь	16
Инновационная деятельность	16
Квалификация персонала	70
Обеспеченность питанием	132