МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «ДОНСКОЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ» (ДГТУ)

Институт опережающих технологий «Школа Икс»

Директор ИОТ «Школа Икс»
_____ П.В. Герасин
подпись

«___» _____ 2024 г.

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА
бакалаврская работа

Тема: «ИЗВЛЕЧЕНИЕ СЕМАНТИЧЕСКОГО ЯДРА РУССКОЯЗЫЧНОГО ТЕКСТА С ПОМОЩЬЮ АЛГОРИТМОВ МАШИННОГО ОБУЧЕНИЯ»

Направление подготовки 09.03.03 Прикладная информатика

Направленность Информационные технологии и системный анализ

Обозначение ВКР 09.03.03.220000.000 Группа ХИТ 41

Обучающийся _____ А.А. Ли
подпись, дата

Руководитель ВКР _____ доцент, к.т.н. А.Ф. Лысенко
подпись, дата

Консультанты по разделам:
«Экономическое обоснование работы» _____ доцент, к.э.н. О.И. Гузенко
подпись, дата

«Безопасность и экологичность работы» _____ доцент, к.э.н. Е.В. Котлярова
подпись, дата

Нормоконтроль _____ доцент, к.т.н. М.В. Чавычалов
подпись, дата

Ростов-на-Дону

2024

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «ДОНСКОЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ» (ДГТУ)

Институт опережающих технологий «Школа Икс»

Директор ИОТ «Школа Икс»
_____ П.В. Герасин
подпись

«___» _____ 2024 г.

ЗАДАНИЕ

на выполнение выпускной квалификационной работы

Тема «ИЗВЛЕЧЕНИЕ СЕМАНТИЧЕСКОГО ЯДРА РУССКОЯЗЫЧНОГО ТЕКСТА С ПОМОЩЬЮ АЛГОРИТМОВ МАШИННОГО ОБУЧЕНИЯ»

Обучающийся Ли Александр Альбертович

Обозначение ВКР 09.03.03.220000.000 Группа ХИТ 41

Тема утверждена приказом по ДГТУ от «19» марта 2024 г. № 1317-ЛС-О

Срок представления ВКР к защите 19 июня 2024 г.

Исходные данные для выполнения выпускной квалификационной работы:

- действующие законодательные акты РФ;
- план внедрения цифрового следа в образовательный процесс;
- документация по использованию инструментов информационных технологий;
- данные о тарификации технологических инструментов на их веб-сайтах;
- энциклопедии, государственные ресурсы для утверждения терминологии.

Содержание выпускной квалификационной работы

Введение:
Выпускная квалификационная работа (далее – ВКР) представляет собой модель машинного обучения, в обучении которой использовались данные компании-заказчика для определения коммуникативной цели реплик участников групповых работ. Данные необходимы для фасилитации образовательного процесса в ходе пост-деятельностной рефлексии.

Наименование и краткое содержание разделов:

1. Аналитическая часть

В качестве первого этапа ВКР послужило приведение терминологии для обоснования сущности проекта и основы для его проектирования. Кроме того, был проведен анализ ситуации заказа, потенциальные клиента и позиция инструмента на рынке.

2. Концепция

В ходе исследования были инициализированы входные данные, выявлен возможный пайплайн работы инструмента и его внедрение в образовательный процесс. Также был проведен брифинг на тему используемых концепций, таких как машинное обучение и задача классификации.

3. Разработка ML-решения

В ходе применения различных инструментов кодирования токенов и классификации и их последующей оценки был сформирован технологический стек работы.

4. Экспериментальное внедрение работы

На предоставленных компанией-заказчиком материалах была проведена качественная оценка работы выбранного стека, продемонстрированы результаты, зафиксирована инструкция к деплою инструмента.

5. Экономическое обоснование работы

Иллюстрация плана производства и организации деятельности, а также финансовой карты расходов и доходов при внедрении инструмента.

6. Безопасность и экологичность работы

В данной главе были суммированы мероприятия по обеспечении безопасности при проектировании и эксплуатации разрабатываемого инструмента, а также проведены замеры для оценки приемлемости условий организации работы.

Заключение:
В ходе ВКР были выполнены все поставленные задачи: был создан цифровой инструмент, предоставляющий материал для рефлексии по завершении онлайн групповой работы. В планах масштабируемости системы рассматривается увеличение датасета посредством сбора цифрового следа, увеличение количества классов-коммуникативных целей, дополнительное внедрение LLM-моделей для более связных выходных данных.

Перечень графического материала:
1. Размеченный датасет, QR-код на Google Colab, аспекты работы моделей.
2. Схема пайплайна работы и пространство векторов Word2Vec.
3. Графики оценки F1-score модели Word2Vec с классификатором SVC.

Руководитель ВКР _____ доцент А.Ф. Лысенко
подпись, дата

Задание принял к исполнению _____ А.А. Ли
подпись, дата

Аннотация

Выпускная квалификационная работа (ВКР) посвящена теме «Извлечение семантического ядра русскоязычного текста с помощью алгоритмов машинного обучения». Работа является проектом по заказу цифровой образовательной компании ООО «Лаборатория Мысли», фокусирующейся на групповой деятельности. Работа внедряется в процесс групповой работы, в частности бизнес-игры «Скифская лестница», предоставляя участникам материал для пост-деятельностной рефлексии. Для считывания семантики высказываний были задействованы собранные для обучения уникальные данные компании, существующие инструменты для транскрибации и диаризации речи из аудиофайла и плотные векторные представления Word2Vec, учитывающие контекст реплик.

Структура ВКР состоит из введения, основного текста, заключения, приложений, перечня информационных ресурсов и графической части. Эти данные представляют артефакты комплексного подхода к решению задачи, все задействованные инструменты, а также векторное пространство на обученных данных. Содержательная часть ВКР содержит 52 страницы основной работы, среди которых 14 рисунок, 2 таблицы и 3 листа приложений. Перечень использованных информационных ресурсов содержит 14 позиций. К работе приложены 3 листа графического материала.

Annotation

The final qualification work (FQW) is dedicated to the topic "Extracting the Semantic Core of Russian Text using Machine Learning Algorithms". The work is a project commissioned by the digital educational company LLC "Laboratoriya Mysli", focusing on group activities. The work is implemented in the group work process, particularly in the business game "Scythian Staircase", providing participants with material for post-activity reflection. To detect the semantics of utterances, unique data was collected for training, existing tools for transcribing and diarizing speech from an audio file were utilized, as well as Word2Vec embeddings, considering the context of utterances employed.

The structure of the FQW consists of an introduction, main text, conclusion, appendices, a list of references, and a graphic part. These data represent the artifacts of a comprehensive approach to solving the problem, all the involved tools, as well as the vector space on the trained data. The substantive part of the FQW contains 52 pages of the main work, including 14 figures, 2 tables, and 3 pages of appendices. The list of used information resources contains 14 items. 3 sheets of graphical material are attached to the work.

Содержание

Введение

В современной реальности ученик не зависит от учителя или места обучения благодаря цифровым коммуникационным технологиям. Он может выбирать, где и чему учиться, в какой среде развиваться, и каким образом включаться в деятельность. В этой новой системе образования успех зависит не только от того, насколько обучение адаптировано к текущему социально-экономическому укладу, но и от способности человека адаптироваться, осваивать новые виды деятельности и приобретать профессиональные навыки.

Цифровой след, внедренный в контекст образовательного процесса, предоставляет новое аналитическое поле. Данное поле становится объектом последующей глубинной работы, направленной на фиксацию прогресса каждого участника. Если исключить интеграцию цифрового следа в процессы онлайн групповых работ, то отслеживание активного участия, открытости к коммуникации, релевантности произнесенного, и общего вклада в деятельность становится неизмеримым на индивидуальном уровне ввиду отсутствия неких количественных параметров. Цифровой след же, напротив, предоставляет возможность обернуть возникающие процессы в статистические данные. Таким образом, цифровой след представляет собой уникальный источник информации, который может быть использован для более глубокого понимания эффективности работы и оценки индивидуального вклада каждого участника в проектную деятельность и его личностного прогресса.

Целью данной дипломной работы является создание инструмента на базе цифрового следа для анализа групповой работы участников образовательных программ и предоставления им материала для саморефлексии по завершении групповых взаимодействий. В основе решения лежит обработка записей групповых работ с использованием методов машинного обучения для последующего анализа человеком. В рамках дипломной работы планируется разработка сервиса на основе интеграции сервисов обработки естественного языка, который принимает запись групповой работы и предоставляет краткую

отчетность об основных событиях, возникших в контексте данного группового взаимодействия. Для анализа необходима запись сессии групповой работы с приемлемым качеством звука. Запись проходит через транскрибацию и разделение на высказывания отдельных участников (диаризацию), а затем проходит количественный и семантический анализ. В результате пользователь получает сводку о ключевых действиях, произошедших во время работы, и, таким образом, почву для рефлексии.

Планируется использование нескольких готовых моделей для анализа, находящихся в открытом доступе, и написание своего инструмента для категоризации высказываний участников.

Данный проект разрабатывается по запросу цифровой образовательной платформы ООО «Лаборатория Мысли» (далее — EntSpace). Его применение предполагается в контексте образовательных программ с использованием групповых проектных работ. При этом результат может использоваться не только для создания цифрового артефакта, но и в качестве материала для рефлексии своего движения участником образовательного процесса. Таким образом, потенциальными пользователями сервиса являются модераторы, руководители проектной работы или программы и лица, ответственные за объективацию непродуктовых результатов программы.

Таким образом, потенциал применения инструмента не ограничен отдельной компанией. Целевая система может применяться в целом ряде контекстов, в которых стоит задача организации личной рефлексии по итогам группового действия. Для этой цели в рамках данной работы мы сосредоточимся на первом этапе рефлексии (после предъявления целей) – фиксации способа действия участника.

1 Аналитическая часть

1.1 Понятия основных концепций в рамках проекта

1.1.1 Цифровой след

Проект непосредственно связан с понятием цифрового следа — «данных об образовательной, профессиональной или иной деятельности человека, представленные в электронной форме» [1]. Цифровой след служит различным целям. Например, объективации освоения тех или иных компетенций, предоставления обратной связи обучающимся, построения траектории движения студентов по программе, внешней коммуникации или оценивания результативности программы.

Результат несложно объективировать, когда обучение сопровождается тестами и формальным оцениванием, но когда оно построено вокруг групповой проектной работы и других групповых коммуникативных форматов, сделать это становится сложно. Пленарные заседания и общие дискуссии позволяют зафиксировать продуктовый результат групп и сделать некоторые выводы относительно образовательной позиции учащегося, но этого недостаточно, чтобы зафиксировать другого рода результаты – например, формирование социальных связей, изменение в представлениях, проявлении тех или иных компетенций. В то же время, процесс работы в группах предоставляет обширный материал для прогрессивного оценивания по данным показателям.

Основная цель использования цифрового следа в образовании — это сбор, анализ и использование цифровых данных о деятельности учащихся для совершенствования образовательных процессов. Цифровой след позволяет получить различную информацию о предпочтениях, успехах, стиле обучения и взаимодействии студентов с материалами, платформами и инструментами. Главная цель заключается в использовании этих данных для персонализации образовательного опыта, оптимизации обучения, создания

индивидуализированных образовательных программ и оценке успехов студентов.

Смещая фокус на непосредственно групповые работы, стоит отметить, что цифровой след дополняет их, предоставляя ценные данные о взаимодействии участников внутри команды. Он позволяет анализировать степень активности каждого участника, его вклад в обсуждения, и эффективность коммуникации. Эта информация может быть ценным инструментом для оптимизации командных процессов, выявления сильных сторон участников, а также для улучшения эффективности совместной работы в проекте.

1.1.2 Рефлексия

После анализа запроса к системе от компании-заказчика был сделан вывод, что конечная проблема, которую мы закрываем – организация индивидуальной рефлексии, а именно After Action Review (AAR) по итогам групповой работы или группового действия в целом [2]. Данный подход и его включение в процессы групповых взаимодействий поддерживает культуру непрерывного обучения наряду со стремлением находить инновационные практики для улучшения внутригрупповых процессов. На рисунке 1 кратко суммированы преимущества внедрения AAR.

Для рефлексии в данном контексте необходимо следующее:

— зафиксировать результат, который человек хотел достичь,

— зафиксировать результат, который получилось достичь,

— зафиксировать способ действия (стратегию), который привел к тому, что не получилось, и положить гипотезу того, каким образом его нужно изменить,

— зафиксировать способ действия, который привел к успеху.

Сбором и обработкой цифрового следа мы целимся в третий этап, фиксируя то, как человек ведет себя в группе.

Рисунок 1 – Преимущества AAR подхода

1.1.3 Семантическое ядро

Семантика — раздел лингвистики, изучающий смысловое значение единиц языка [3]. Семантическое ядро — цифровой инструмент, использующий смысл слов для преобразования их в единицы, находящих свое применение в различных сферах деятельности, таких как маркетинг, сбор цифрового следа, аналитика статистики запросов, и другое.

Важность семантического ядра в рамках проекта имеет неотъемлемую роль в связи с контекстуальностью произнесенных людьми реплик. Таким образом, анализировать слова в изолированном порядке будет неподходящим методом, поскольку они не всегда репрезентуют верный контекст, присущий разговорной речи в групповой деятельности. Например, рассматриваются две следующие реплики:

— «Да, давайте три [очка поставим], чтобы больше ресурсов осталось»;

— «Давай, я согласна!».

Очевидно, что две эти реплики имеют абсолютно разные цели, но они обе содержат слово «давай». С помощью семантического ядра удастся уловить разницу в коннотации реплик при их распределении.

Итак, сбор цифрового следа введен для формирования семантического ядра поведения участников групповых взаимодействий (являющееся уникальным для данной деятельности). Далее данные проходят обработку с помощью алгоритмов машинного обучения и предоставляются участникам как материал для рефлексии.

1.1.4 Предложение

В рамках проекта определение «предложения» играет важнейшую роль, поскольку на основе понимания слова строится этап разработки. Как единица синтаксиса, предложение — «это совокупность слов или слово, грамматически оформленная с точки зрения времени и реальности/ирреальности, интонационно завершенная и выражающая сообщение, вопрос или побуждение к действию»[4]. При формировании материала для дальнейшей работы будет учитываться данное определение слова «предложение», учитывая его коммуникативную функцию, которая, в частности, побуждает к действию. Именно в данном контексте употребление реплик класса «предложение» в групповой работе играет важнейшую роль: они напрямую влияют на динамику команды и процесс принятия решений.

1.2 Анализ ситуации заказа

Сервис разрабатывается по запросу цифровой образовательной платформы EntSpace, которая проводит синхронные онлайн программы, включающие в себя, в том числе, режим проектирования в малых группах. Платформа движется к цифровизации и автоматизации процессов и развитию learning analytics. В рамках проекта по сбору цифрового следа было предложено

написать сервис для сбора аналитики по вовлеченности и динамике участников в ходе проектной работы. На рисунке 2 показана схема того, как устроен сбор цифрового следа в ситуации заказа сейчас.

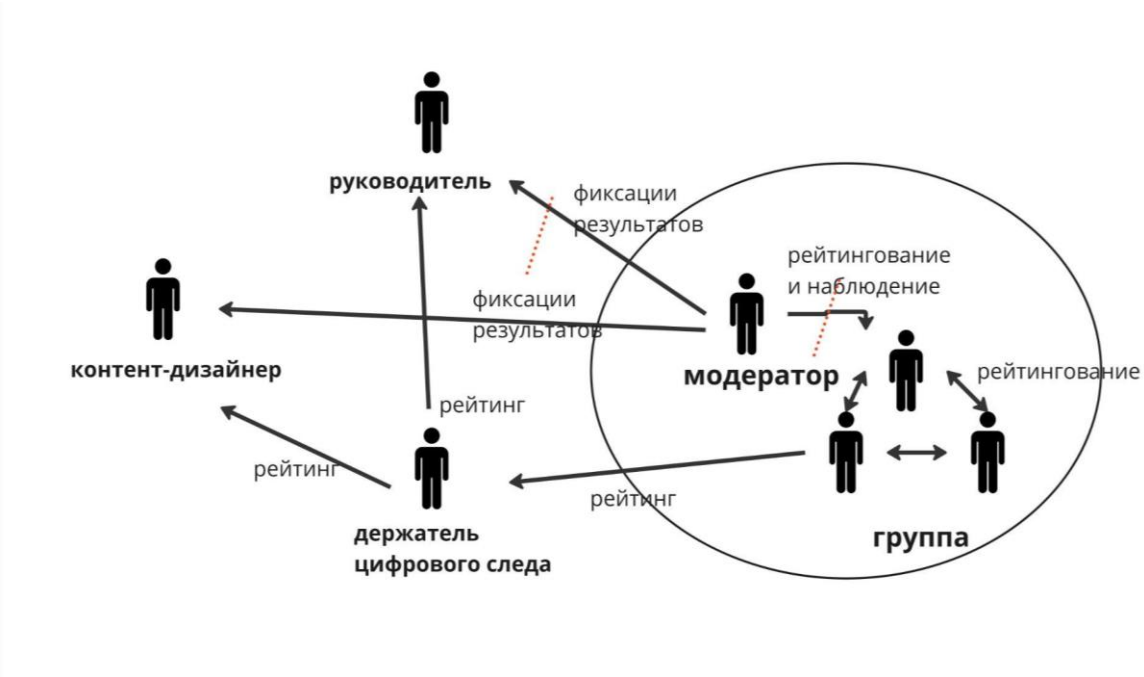


Рисунок 2 – Схема СРТ сбора цифрового следа

Цифровой след формируется на основе рейтинга, а в процессе объективации результатов участников участвуют модератор, руководитель (руководитель проектной работы, продюсер, или ведущий программы) и контент-дизайнер, который фиксирует результаты программы в форме цифровых артефактов для внутренней или внешней коммуникации, аналитики и предоставления участникам в качестве обратной связи.

При этом модератор — единственный человек, который напрямую видит участников во время групповых работ и может охарактеризовать или оценить их работу, и это становится материалом, с которым работают контент-дизайнер и руководитель. КД использует его для создания цифровых артефактов, а руководитель — для оценки эффективности программы, корректировки работы с участниками и отбора людей на те или иные позиции.

1.2.1 Проблема

Разрывы в этой системе в том, что, во-первых, модератор сфокусирован на движении группы по замыслу, нежели на фиксации результатов отдельных участников и обращает внимание лишь на ограниченный спектр результатов, в то время как другие позиции не могут напрямую следить за ходом групповой работы. Во-вторых, еще меньшая часть того, что замечает модератор, оказывается передана следующим позициям. Чтобы преодолеть второй разрыв, создаются различного рода формы и метрики для заполнения модераторами, но без решения первой проблемы сложно достичь аналитики должного уровня.

1.3 Гипотеза решения

Гипотеза нашего решения состоит в том, что этот разрыв преодолевается созданием цифрового инструмента для обработки записей групповых работ и предоставления сводки по ключевым вербальным событиям (в частности, инициатива при предъявлении некоего предложения) внутри данной работы. Таким образом, как показано на рисунке 3, контент-дизайнер и руководитель способны напрямую работать с результатом анализа групповых работ после обработки держателем цифрового следа без необходимости пересматривать многочасовые записи и создавать дополнительную когнитивную нагрузку модераторам. Более того, каждому участнику не нужно будет самостоятельно восстанавливать ментальную картинку всех произошедших взаимодействий в рамках групповой работы, искажая или упуская важные моменты, поскольку сводка будет предоставлена по факту возникновения взаимодействия. Важным аспектом касательно рефлексии является работа человеческой памяти, которая не способна фиксировать события с детальной точностью, либо формируя ложные воспоминания пост-фактум [5].

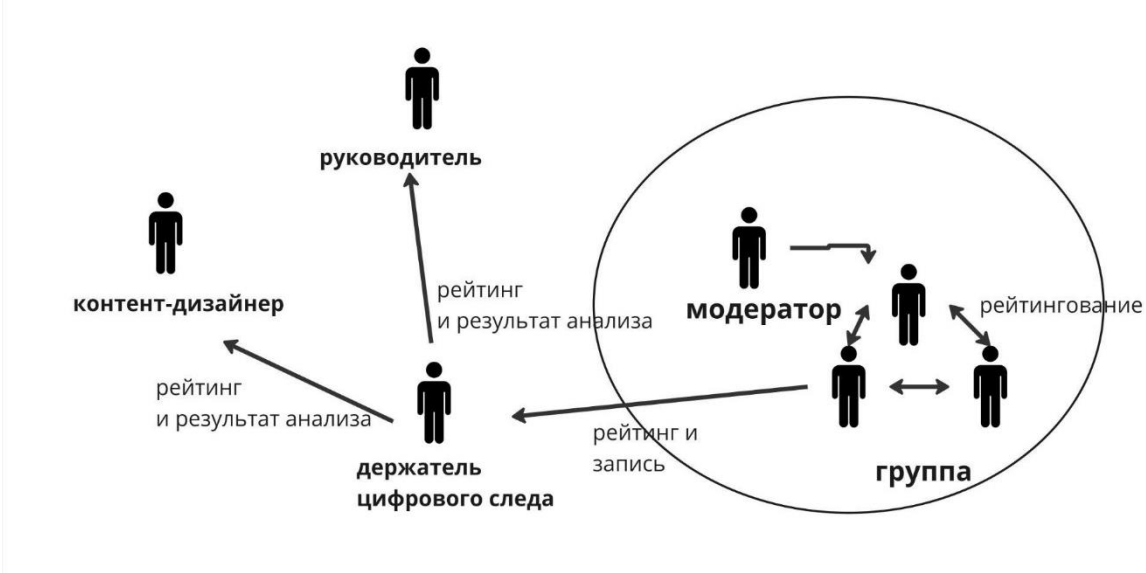


Рисунок 3 – CPT при внедрении инструмента

Главный вызов, с которым предстоит работать, состоит в том, насколько ценна будет предоставляемая информация участникам и насколько это может улучшить процесс рефлексии. Важно учитывать индивидуальную способность к анализу собственных действий в ретроспективе и фиксировать определенные точки движения или роста. Для смягчения проблемы предполагается, что участник не будет оставлен с информацией наедине. Ключевым игроком данного процесса будет являться модератор, направляющий мыслительный процесс в нужное русло — таким образом, предоставляя материал для работы. Для целостного анализа индивидуального вклада в групповую работу необходимо оценивать более сложные аспекты, такие как частота коммуникации, относящейся или не относящейся к общей тематике работы, склонность к задаванию вопросов или критике высказываний других, введение в дискуссию новых понятий. Если вы часто обсуждаете проблемы на работе и погоду с животными, уникальность и частота данных высказываний будет высокой, но не будет означать вашу вовлеченность и вклад в проект. Таким образом, сводка будет включать в себя только релевантную информацию — предложения, возникающие в рамках группы. Кто что предложил, в каком соотношении — чтобы также имелась целостная картинка по всей групповой работе. На данном

этапе в работу вступает семантический анализ — алгоритм, определяющий, является ли данное высказывание предложением или нет.

1.3.1 Анализ рынка

Внедрение в СРТ цифровых образовательных процессов начало набирать обороты особенно в последнее время — с развитием технологий искусственного интеллекта, хорошо проявляющее себя в рамках NLP (обработке естественного языка). Было выделено несколько перспективных технологий, применение которых переплетается с идеей проекта и его аспектами:

— финская технологическая компания Valamis, занимающаяся разработкой цифровых решений для образования в крупных организациях, предоставляет сервис для анализа образовательной траектории и результатов сотрудников на основе трекинга их активности на различных площадках. В результате для каждого ученика собираются утверждения в формате «Актор – Действие – Объект – Результат», по которым возможно не только отслеживать траекторию сотрудников, но и исследовать эффективность обучения и формировать оптимальную образовательную траекторию [6];

— Vosaic. Данный инструмент предлагает загрузку записи некой работы (необязательно групповой) и предоставляет пространство для передачи обратной связи на определенные таймстампы. Имеется вариация расширения возможностей с помощью ИИ. Проблема решения заключается в том, что работа производится непосредственно с записью, и оно рассчитано на иные формы работ: публичные выступления, презентация — для более качественной обратной связи. Внедрение ИИ не предусматривает вычленение семантического ядра из загруженной записи [7];

Помимо активных игроков на рынке образовательных цифровых продуктов также имеются инструменты, применение которых может затрагивать поставленную гипотезу. Например, Voyant Tools — это инструмент для анализа текста, который позволяет визуализировать и изучать текстовые данные. Он

может использоваться для облегчения анализа текста, выявления ключевых слов, определения частоты слов и создания визуальных диаграмм, чтобы помочь исследователям понять основные тенденции и паттерны в тексте. Voyant Tools фокусируется на анализе и визуализации текстовых данных, поэтому он может помочь в обработке текстов, которые представляют собой часть цифрового следа. Например, если у вас есть текстовые данные из онлайн-источников или социальных медиа, Voyant Tools поможет вам анализировать содержание этих текстов, выявлять ключевые слова, определять частоту их употребления и визуализировать различные паттерны. Тем не менее, инструмент является универсальным решением, который в том числе затрагивает поставленную нами гипотезу; чаще всего он используется для изучения языков. Таким образом, проблема данного решения заключается в отсутствии индивидуального подхода и сильной обобщенности результатов [8].

1.4 Стейкхолдеры и ЦА

В контексте заказа существуют несколько типов конечных пользователей:

— контент-дизайнер образовательной программы. Позиция контент-дизайнера предусматривает, во-первых, объективацию результатов и создание материалов для предоставления участникам или внешней аудитории. Контент-дизайнер обязан обладать компетенцией и интересом в образовании или предпринимательстве, но не обязательно в дизайне и программировании. Это обуславливает необходимость в простоте визуализации результатов анализа;

— держатель цифрового следа. В текущей ситуации занимается процессом сбора и обработки рейтинга участников и чаще всего совмещается с другими позициями;

— руководитель проектной работы. Важным шагом по завершении программы является технологическая рефлексии, для которой результат анализа может

послужить материалом. Поэтому необходимо предусмотреть предоставление данных не только но участникам, но и по группам, в которых они находились;

— продюсер. Для ситуации заказа важно, что в ходе групповой работы происходит отбор талантливых студентов на позиции в компанию. Интерес продюсера в этой ситуации в том, чтобы метрики давали возможность сделать выводы о способностях и потенциальных позициях, которые бы подошли кандидатам;

— модератор. Результат анализа может использоваться модератором для корректировки своей стратегии в отношении вовлечения отдельных участников;

— участник программы. Важно соблюдать этические принципы в отношении личных данных и избегать дискомфорта, который может быть связан с ощущением постоянной оценки и присутствия внешнего наблюдателя.

При этом лицом, принимающим решения, являются руководители образовательных программ с использованием групповой работы. Одним из оснований для принятия решений является то, что при использовании для анализа оффлайн групповых работ отдельной задачей является обеспечение качественной аудиозаписи. Поэтому мы предполагаем, что в первую очередь наше решение может быть интересно для руководителей онлайн образовательных программ.

2 Концепция

2.1 Входные данные

Обучение модели и проверка гипотезы планируется на материале игр «Скифская лестница», проводимых EntSpace. Формат предусматривает короткие такты групповой работы, в ходе которых принимается решение относительно ставки в игре. Это позволяет за короткий срок отметить, насколько активны были участники, предлагая свои идеи [9].

Часть аналитики поступаемых данных требует следующий порядок действий:

— сбор уникального датасета (в частности, «цифровой след» участников игр и адаптированная в текстовый формат аудиозапись взаимодействия). Разметка данных происходит посредством распределения реплик по категориям «предложение» и «другое». Определение категории реплики базируется на определении «предложения» (раздел 1.1.4);
— обработка датасета посредством современных библиотек (приведение в токены);
— применение различных методов кодирования токенов;
— использование алгоритмов классификации на основе закодированных токенов, оценка моделей и выявление метода с самым качественным результатом;
— сохранение моделей и применение в деятельности.

2.2 Пайплан работы системы

Пайплайн внедрения инструмента в процесс представлен на рисунке 4.

Система состоит из нескольких компонентов:

— блок транскрибации и диаризации, преобразующий аудио в текстовый файл с разделением реплик по разным спикерам вида «Speaker 1: реплика»;

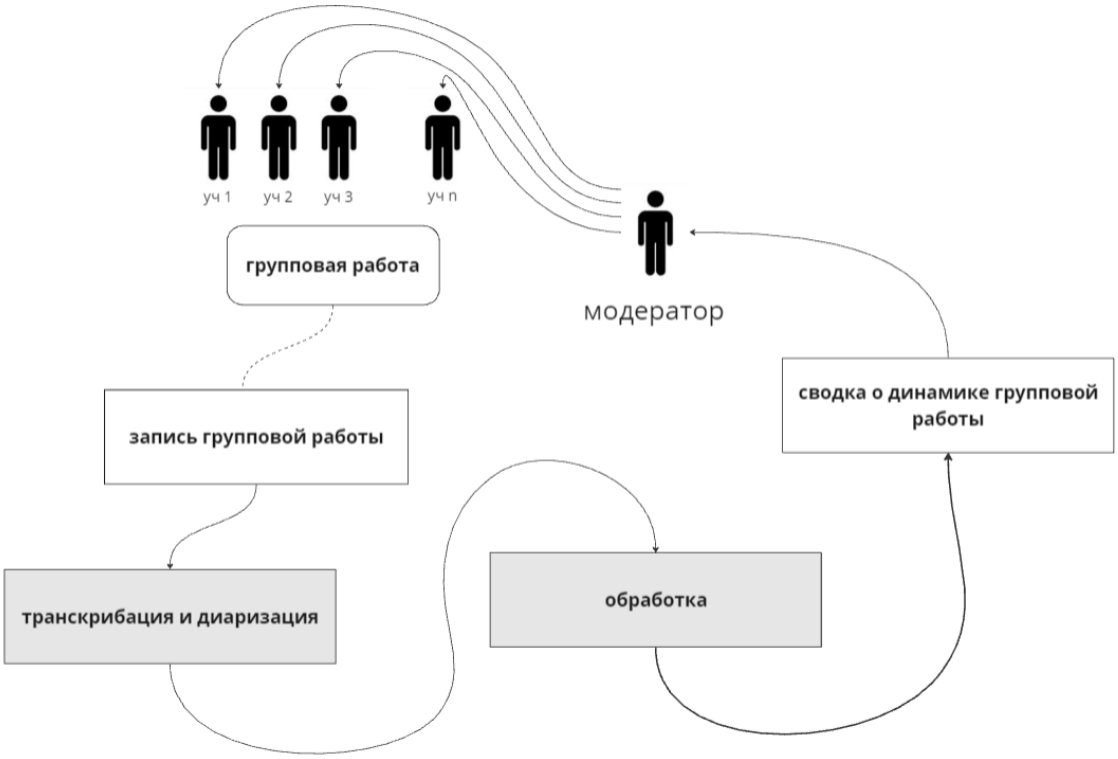


Рисунок 4 – Пайплайн работы проекта

— обработка текстового файла, классификация реплик по категории «предложение», предоставление сводки модератору;
— предоставление результата сводки участникам, проведение рефлексии.

2.3 Блок аналитики

2.3.1 Машинное обучение

Машинное обучение — область искусственного интеллекта, занимающаяся разработкой методов, которые позволяют компьютерам обучаться на поступаемых данных и улучшать свою работу на основе опыта решения множества сходных задач. Построение данных методов базируется на различных отраслях математики: математическая статистика, численные методы, теория вероятности, методы оптимизации, и другие. Область можно описать следующим образом: «Раздел машинного обучения, с одной стороны, образовался в результате разделения науки о нейросетях на методы обучения

сетей и виды топологий их архитектуры, с другой стороны — вобрал в себя методы математической статистики» [10].

Способы обработки данных в машинном обучении следующие:

— обучение с учителем, где нейросеть получает заранее отмеченный человеком набор данных (датасет), где заранее отмечено, что эти данные обозначают. Данный способ полезен для нейросетей, решающих задачу классификации — где необходимо считать некоторый объем данных и распределить его по категориям. Например, выявление негативного или положительного отзыва; определение сообщения как спама;

— обучение без учителя, где нейросеть получает объем неразмеченных данных и пытается самостоятельно найти связи и общие признаки. Данный способ полезен для алгоритмов, решающих задачу кластеризации и нахождения ассоциаций; последнее относится к рекомендательной системе, где нейросеть может строить связи на основе одних объектов и распространять их на другие, похожие;

— обучение с подкреплением, где получаемые нейросетью данные обрабатываются случайным образом, которые впоследствии модифицируются на основе предоставленных алгоритму критериев. Требуется большое количество итераций для получения желаемого результата.

Преимущественным способом обработки данных в машинном обучении в данном проекте рассматривается обучение с учителем. Для отслеживания инициативности участников групповых работ и определения их реплик как «предложение», требуется предоставить алгоритму уникальные размеченные данные с групповых работ, на основе которых нейросеть сможет распределять реплики на новых данных. Кластеризация неуместна ввиду четко поставленной задачи, где присутствуют определенные классы; по этой же причине неуместно обучение с подкреплением.

Процесс создания модели машинного обучения включает в себя несколько шагов (рисунок А.1, приложение А). Автор источника отметил, что это процесс прогнозирующей модели [11]. Тем не менее, алгоритм применим и к процессу подготовки модели классификации:

— вопрос: как часто проявляется инициатива каждого участника во время групповых работ?;
— сбор данных и создание признаков: транскрибированные записи четырех групповых работ, где вручную реплики были отмечены как «предложения», фиксируя их коммуникативную цель. Остальные реплики отмечены как «другое»;
— очистка данных и нормализация данных: происходит с помощью токенизации и фильтрации изначального датасета;
— выборка данных: нейросеть после изучения признаков и классов получает на входе учебные и тестовые данные;
— создание модели и оценка модели: алгоритм, обученный на учебных и тестовых данных, выдает процент точности предсказания классов признакам;
— развертывание модели: интеграция обученной модели в рабочие процессы.

2.3.2 Задача классификации

Задача классификации в целом заключается в том, что при наличии объекта (текста, реплики) нужно определить, к какому классу относится этот объект. Классы должны быть заданы заранее и являться дискретными.

Типы задач классификации текста выделяются следующие:

— бинарная классификация (binary classification). В данном типе имеются лишь два класса объектов, где каждый объект может принадлежать только одному классу;
— многоклассовая классификация (multiclass classification). В данном типе имеется несколько классов, где объект также принадлежит одному классу;

— многозначная классификация (multilabel classification). В данном типе имеется несколько классов объектов, и каждый объект может примерять несколько классов.

В рамках проекта в классификаторе имеются две метки (класса): «предложение» и «другое». Задачей классификации будет обучиться на размеченном датасете и научиться прогнозировать класс поступаемой реплики в ходе применения модели — способом обработки данных будет обучение с учителем. На рисунке 5 представлена схема перехода к многоклассовой классификации проекта при его масштабируемости и сбора большего количества данных, что предоставит больше материала для обработки впоследствии.

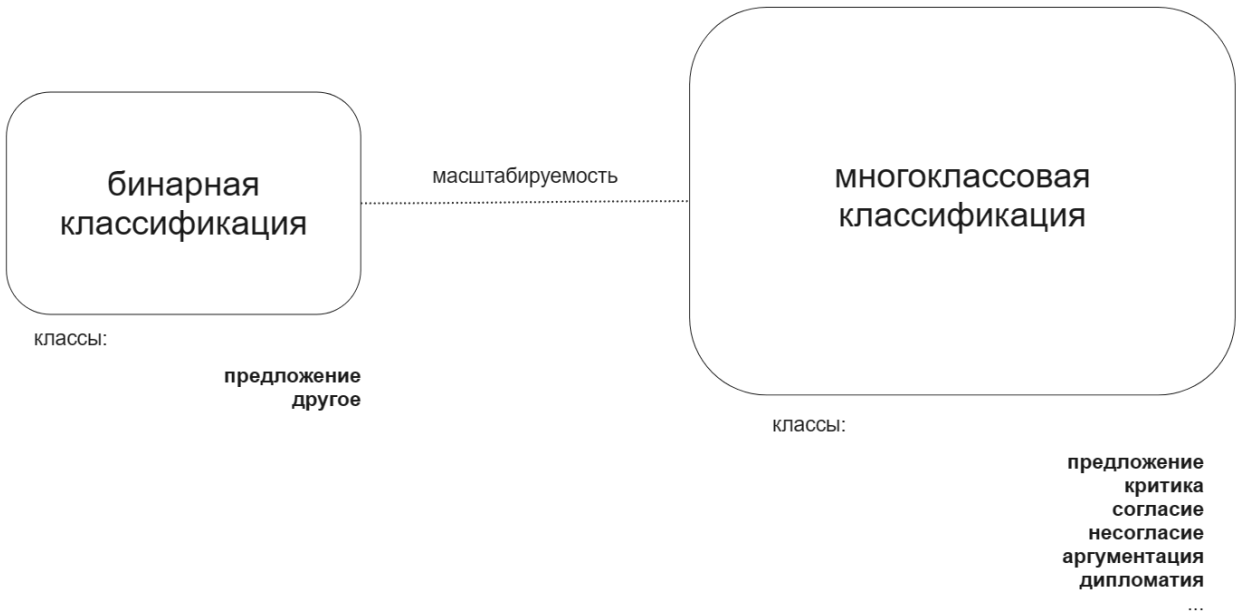


Рисунок 5 – Позиция блока в рамках масштабируемости

2.3.3 Оценка качества классификации текста

Для оценки алгоритмов используются следующие метрики:

— кросс-валидация. Данная метрика помогает оценить устойчивость модели к различным подвыборкам данных и уменьшить эффект переобучения. Она разбивает данные на тренировочную и тестовую выборки и повторяет процесс обучения с разными комбинациями набора данных;

— доля правильных ответов (accuracy). Эта метрика показывает, насколько достоверно модель распределяет реплики по категориям (другое и предложение);

— точность (precision). Данная метрика фокусируется на доле положительных верных предсказаний (в нашем случае, если реплика является «предложением» — то это «положительный» исход). Чем выше показатель, тем меньше ложноположительных срабатываний при работе алгоритма. При этом, метрика не учитывает ситуации, когда фактические предложения не были отнесены к классу «предложение». Работает по распределению в формуле (2.1):

$$Precision = \frac{TP}{TP + FP}, \quad (2.1)$$

где TP – True Positive;

FP – False Positive (обозначения представлены на рисунке 6);

— полнота (recall). Данная метрика учитывает все принадлежащие положительному классу объекты, учитывая, были или не были они отнесены к классу «предложение». Расчет метрики приведен в формуле (2.2):

$$Recall = \frac{TP}{TP + FN}, \quad (2.2)$$

где FN – False Negative;

— f-score. Данная метрика является средним гармоническим значением между precision и recall. Расчет в формуле (2.3):

$$F1\ score = \frac{2PR}{P + R}, \quad (2.3)$$

где P – Precision;

R – Recall.

В задаче классификации реплик к категории «предложение» метрика precision является приоритетной, поскольку корректно распределенные реплики важнее, чем те реплики, что не отнеслись к классу «предложение». Причиной этого является тот факт, что при предоставлении сводки участнику групповой работы уже будет иметься список произнесенных им предложений, и отсутствие некоторых из них несильно повлияет на исход его работы с материалом.

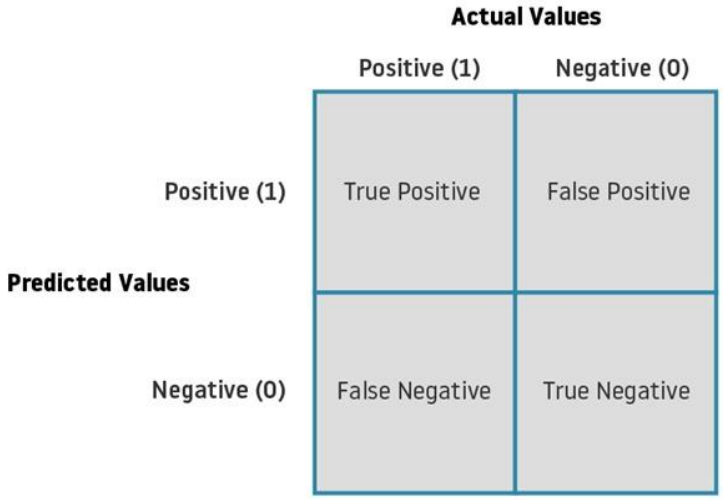


Рисунок 6 – Распределение возможных ошибок в задаче классификации

Тем не менее, если реплике был ложно присвоен класс «предложение», тогда участник не будет уверен в том, с чем ему стоит работать. Таким образом, в рамках проекта точность > полнота, но ввиду малого количества обучаемых данных, метрики по отдельности будут исключены из оценки качества моделей; вместо этого будет внедряться вычисление F1-score на каждом шаге кросс-валидации и анализ среднего значения и стандартного отклонения F1-score по всем шагам. Описанный подход полезен следующим образом:

— при работе с небольшим объемом данных способ является самым информативным;
— f1-score дает сбалансированную оценку модели, базируясь на гармоническом среднем между точностью и полнотой;
— кросс-валидация помогает использовать имеющиеся данные максимально эффективно и предотвратить переобучение [12].

3 Разработка ML-решения

Задача блока машинного обучения — извлечение семантического ядра русскоязычного текста. Однако следует отметить, что некоторые применяемые технологические инструменты не фокусировались на анализе смысловой нагрузки; вместо этого они применялись для выполнения операций, основанных на математических моделях и статистических вычислениях — что нельзя отнести к семантике. Тем не менее, в рамках разработческой части продукта будут представлены как алгоритмы начального уровня, так и инструменты, представляющие собой продвинутые разработки в данной области, чтобы продемонстрировать разницу в работе алгоритмов и необходимости учета семантики при работе с репликами. Детали приведены на листе 1 графической части.

3.1 Сбор и разметка датасета

На рисунке 7 представлен один из размеченных файлов датасета. Четыре файла были использованы в качестве датасета для обучения модели.

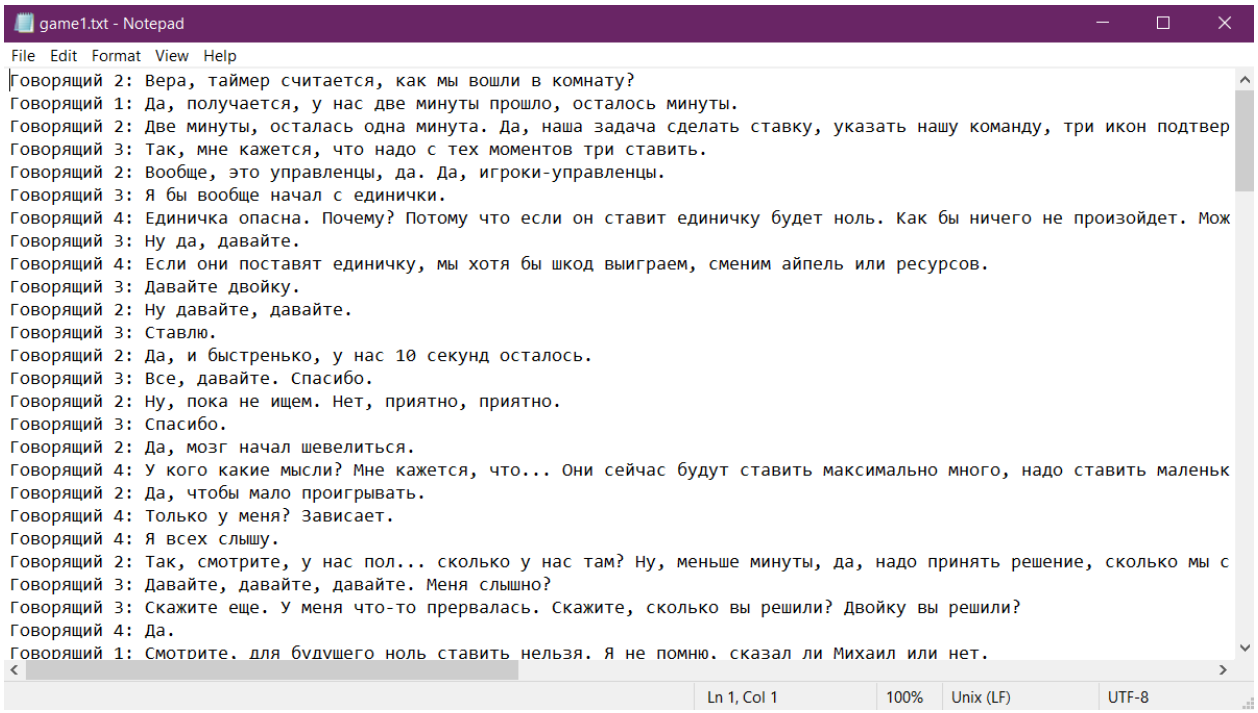


Рисунок 7 – Один из обрабатываемых файлов с транскрибированной речью

3.2 Токенизация

Токенизация — приведение слов в токены. Токены — образованные из различных словоформ единицы, используемые в алгоритмах машинного обучения. Они необходимы для подготовки данных к векторизации, их структурирования, уменьшения размерности и, как следствие, повышения качества модели. Это происходит путем избавления от ненужных либо неиспользуемых элементов, таких как знаков препинания, стоп-слов, различных форм одного и того же слова. Для дальнейших моделей будет использоваться библиотека pymorphy3, являющаяся морфологическим анализатором для русского языка. Первоначальный файл с данными для обработки располагается на рисунке 8.

	text	category
0	Давайте двойку.	предложение
1	Ставлю.	предложение
2	Нам надо пять поставить.	предложение
3	Ребят, 2 надо ставить.	предложение
4	2 надо ставить, у нас ресурсов нет.	предложение
...	...	...
258	Так, ну вы правила поняли?	другое
259	Я поняла правила. Нам нужно, чтобы у нас остал...	другое
260	То есть у нас сейчас получается будет два раун...	другое
261	Смотрим их, да, и наблюдаем за стратегией других.	другое
262	Так, а мне, Варвара, мне надо каждый раз отпра...	другое

263 rows × 2 columns

Рисунок 8 – Первичный вид датасета

Каждую строку столбца *text* необходимо токенизировать, приведя к нормальной форме. Для этого необходимо установить и импортировать библиотеки. Далее, инициализируем функцию, которая в дальнейшем будет использоваться для быстрого препроцессинга текста. Затем необходимо подготовить объекты, содержащие в себе стоп-слова, знаки пунктуации (которые

оказались уникальны для датасета) и морфологический анализатор pymorphy3 (листинг 3.1).

Листинг 3.1 – Функция препроцессинга текста

```
def preprocess(text, stop_words, punctuation_marks, morph):
    tokens = word_tokenize(text.lower())
    preprocessed_text = []
    for token in tokens:
        if token not in punctuation_marks:
           lemma = morph.parse(token)[0].normal_form
           if lemma not in stop_words:
              preprocessed_text.append(lemma)
    return preprocessed_text
```

3.3 Логистическая регрессия

Логистическая регрессия — статистическая модель, которая используется для предсказания вероятности появления некоторого события Y путем его сравнения с логистической кривой (сигмоидой). Результатом выступает число от 0 до 1, выдающее вероятность принадлежности события к заданным классам (как правило, бинарным).

В рамках проекта, классы следующие: {предложение: 1, другое: 0}. Таким образом, если вероятность предсказания модели будет выше либо равна 0.5, то реплика будет принадлежать категории «предложение»; в ином случае — «другое».

3.3.1 Мешок слов (bag of words)

Мешок слов — метод кодирования, который позволяет представить предложение (либо целый документ) в виде чисел по частоте их упоминания. Данный способ полезен, когда результат зависит от наличия или отсутствия определенных слов. Вектор, формирующийся данным методом, фиксирует количество упоминания того или иного слова.

Для проверки работы модели логистической регрессии в связке с мешком слов требуется привести все токенизированные реплики в последовательности, где за каждым словом будет скрываться число. Числа для слов выбираются в соответствии с частотой, с которой слово встречается в тексте; это представлено на рисунке 9.

	text	category	preprocessed_text	sequences
0	Давайте двойку.	1	[давать, двойка]	[2, 32]
1	Ставлю.	1	[ставить]	[4]
2	Нам надо пять поставить.	1	[пять, поставить]	[44, 3]
3	Ребят, 2 надо ставить.	1	[ребята, ставить]	[114, 4]
4	2 надо ставить, у нас ресурсов нет.	1	[ставить, ресурс]	[4, 27]
...	...	...	...	...
258	Так, ну вы правила поняли?	0	[правило, понять]	[200, 17]
259	Я поняла правила. Нам нужно, чтобы у нас остал...	0	[понять, правило, нужно, остаться, максимальны...	[17, 200, 22, 6, 201, 113, 201, 27, 175, 113, ...
260	То есть у нас сейчас получается будет два раун...	0	[получаться, раунд]	[28, 56]
261	Смотрим их, да, и наблюдаем за стратегией других.	0	[смотреть, наблюдать, стратегия]	[12, 409, 33]
262	Так, а мне, Варвара, мне надо каждый раз отпра...	0	[варвар, каждый, отправить, ещё, ответ, следую...	[197, 78, 16, 42]

Рисунок 9 – Частота встречаемых в датасете слов

Перед кодированием слов нужно также взять во внимание необходимость фиксированной длины поступаемых векторов и возможность поступления неизвестных слов, тех, которые не были включены в обучение модели.

— код заполнитель: 0;

— код для неизвестного слова: 1.

Таким образом, в словаре нумерация по словам начнется с 2. Далее заполняем константы, подготавливая данные к обучению и создаем словари.

Словарь «index_to_word», работающий обратным образом, полезен для возможности воспроизвести слово по номеру кода. Например, видя значение 2, мы сможем распознать, что за ним скрыто слово «давать». Далее преобразуем список предобработанных слов в список кодов с помощью функции «text_to_sequence», обновляем файл с датасетом, кодируем категориальные данные классов в цифровые и заменяем их в датасете. Результат применения описанных методов виден на рисунке 10.

```
words.most_common(10)
```

```
[('давать', 47),
 ('поставить', 43),
 ('ставить', 28),
 ('мочь', 27),
 ('остаться', 20),
 ('всё', 20),
 ('команда', 19),
 ('это', 18),
 ('выиграть', 12),
 ('просто', 12)]
```

Рисунок 10 – Датасет со столбцом закодированных слов и классов

Данные готовы для обучения. Инициируем функцию «vectorize_sequences», которая будет создавать векторы из списка кодов слов (листинг 3.2).

Листинг 3.2 – Функция, создающая векторы из мешка слов

```
def vectorize_sequences(sequences, dimension=len(words)):
    results = np.zeros((len(sequences), dimension))
    for i, sequence in enumerate(sequences):
        for index in sequence:
            results[i, index] += 1.
    return results
```

Инициализируем модель машинного обучения, проверяем точность модели с помощью метода оценки F1-score в сочетании с кросс-валидацией. Точность модели оказалась 0,66 (+/- 0,24).

3.4 Плотные векторные представления слов (word embeddings)

Векторные представления слов — полезный инструмент преобразования слов с учетом сохранения семантики. Это происходит за счет того, что каждое слово преобразуется в массив чисел с плавающей точкой и располагается в пространстве, где алгоритм может отследить расположение информационных фрагментов относительно исследуемого элемента.

3.4.1 Navec

Библиотека Navec является инструментом для работы с естественными языками, предназначенным для формирования плотных векторных представлений слов и поддержки задач обработки естественного языка. Библиотека часто используется для анализа текстов на русском языке, она обеспечивает предобученные модели и помогает векторизовать слова для последующего их использования в моделях машинного обучения и глубокого обучения. Это мощный инструмент для извлечения представлений слов, которые могут улучшить точность и производительность различных NLP-задач, к примеру, в задачах классификации текста, кластеризации или извлечения информации.

Для применения данной библиотеки на наших данных нужно также пройти все стандартные шаги, начиная с установки и импорта библиотек, определения констант, обработки загружаемого датасета. Касательно констант, для векторного представления необходимо иметь одинаковое количество векторов для каждого элемента. Таким образом, ввиду небольшого размера датасета, максимальную длину укажем 50, а размер вектора, используемый для представления слов, 300. Константу для инициализации генератора случайных чисел укажем 42. Далее необходимо установить предварительно обученные на русском языке векторы Navec.

Для векторизации текста введем функцию и применим ее на токенах датасета, чтобы получить векторы на каждый элемент, как представлено на рисунке 11. Токены формируются функцией «preprocess», инициализированной ранее. Кроме того, столбец с категориями высказываний также приводится к цифровой форме (листинг В.1).

Разделяем данные на вектора и категории, затем отдаем на анализ кросс-валидации, которая автоматически занимается распределением на тренировочную и тестовые выборки; далее выводим оценку модели. Точность модели оказалась 0,63 (+/- 0,19).

	text	category	preprocessed_text	embeddings
0	Давайте двойку.	1	[давать, двойка]	[[-0.20394225, -0.5428032, 0.17268088, -0.5472...
1	Ставлю.	1	[ставить]	[[-0.56923324, -0.3954833, -0.06762549, 0.0721...
2	Нам надо пять поставить.	1	[пять, поставить]	[[0.1683296, -0.4029474, -0.67731494, 0.000809...
3	Ребят, 2 надо ставить.	1	[ребята, 2, ставить]	[[0.56820583, 0.046246283, -0.28193244, 0.5412...
4	2 надо ставить, у нас ресурсов нет.	1	[2, ставить, ресурс]	[[0.21431214, 0.37028718, 0.13679631, -0.18653...
...	...	...	...	...
258	Так, ну вы правила поняли?	0	[правило, понять]	[[-0.5377986, -0.6289255, -0.39644632, 0.04223...
259	Я поняла правила. Нам нужно, чтобы у нас остал...	0	[понять, правило, нужно, остаться, максимальны...	[[-0.25260425, 0.04839928, -0.2842658, -0.0750...
260	То есть у нас сейчас получается будет два раун...	0	[получаться, раунд]	[[0.010760385, 0.26762047, 0.40114763, 0.14934...
261	Смотрим их, да, и наблюдаем за стратегией других.	0	[смотреть, наблюдать, стратегия]	[[0.36390617, -0.5678413, -0.24011786, 0.23606...

Рисунок 11 – Датасет со столбцом плотных векторов

3.4.2 Word2Vec

Word2Vec — дистрибутивно-семантическая модель, разработанная Google. В ее основе в качестве контекста используется «скользящее окно», которое считывает целевое слово и слова, идущие до и после него. Один из подходов обучения модели — это Continuous Bag of Words. Идея подхода такова, что все слова, внесенные в контекст, имеют такие вектора, которые в сумме дают вектор целевого слова, которое находится в «центре окна» и которое нужно предсказать. Второй подход — Skip-gram, работающий противоположным образом: на входе предоставляется слово, а слова контекста предсказываются на основе векторных представлений.

Для начала работы с Word2Vec, ее нужно импортировать и обучить на своих данных. «Sentences» — токенизированные тексты датасета. Параметр «min_count» репрезентует частоту встречаемости слова в документе: в данном случае, слово должно встречаться минимум 5 раз. Иначе можно столкнуться с выбросами и редкими словами, семантику которых невозможно выявить из-за малого количества данных. «Vector_size» указывает размер вектора.

Наедине, предобученная модель Word2Vec может только предоставить векторные представления некоторых слов, а также наиболее близкие к ним слова

семантически. Для корректной работы алгоритма в рамках проекта модели необходимо добавить классификатор, который на основе заданных векторных представлений будет распределять реплики по категориям.

Переходим к использованию модели на наших данных. Необходимо добавить новый столбец в датафрейм, где будут храниться вектора Word2Vec, используемые в обучении для классификатора (листинг 3.3).

Листинг 3.2 – Функция, добавляющая обученные векторы Word2Vec в датафрейм

```
def document_vector(word2vec_model, doc):
    doc = [word for word in doc if word in word2vec_model.wv]
    if len(doc) == 0:
       return np.zeros(word2vec_model.vector_size,
dtype=np.float32)

    return np.mean(word2vec_model.wv[doc], axis=0)

    daf['doc_vector'] = daf['preprocessed_text'].apply(lambda doc:
document_vector(model, doc))
```

Функция «document_vector» используется для получения векторного представления целого документа на основе предобученной модели Word2Vec. Word2Vec отображает каждое слово в векторном пространстве, где каждому слову соответствует вектор определенной размерности. Так, слова с похожими смыслами или часто встречающиеся в одинаковых контекстах находятся близко друг к другу в этом пространстве.

Тем не менее, Word2Vec не предоставляет напрямую вектора для фраз или целых документов, а только для отдельных слов. Для того, чтобы получить представление для больших текстов используется вычисление среднего вектора из векторов всех слов, встречающихся в документе.

Функция делает следующее: она берет список слов документа, проверяет, есть ли они в модели (т.е. были ли они учтены при обучении модели Word2Vec), а затем вычисляет средний вектор из векторов слов, обнаруженных в модели. Если документ не содержит слов, которые есть в модели, то функция возвращает

нулевой вектор. Данное преобразование объединяет семантику всех слов документа в одно векторное пространство, что необходимо для классификации.

В качестве алгоритмов классификации далее будут рассматриваться метод опорных векторов.

3.5 Метод опорных векторов

Предобработанные данные берутся с шага 3.2. В добавление к ним импортируем классификатор вместе с функциями оценки работы модели и разделяем выборку на векторы и классы, запускаем код. Результат работы модели: 0,64 (+/- 0,29).

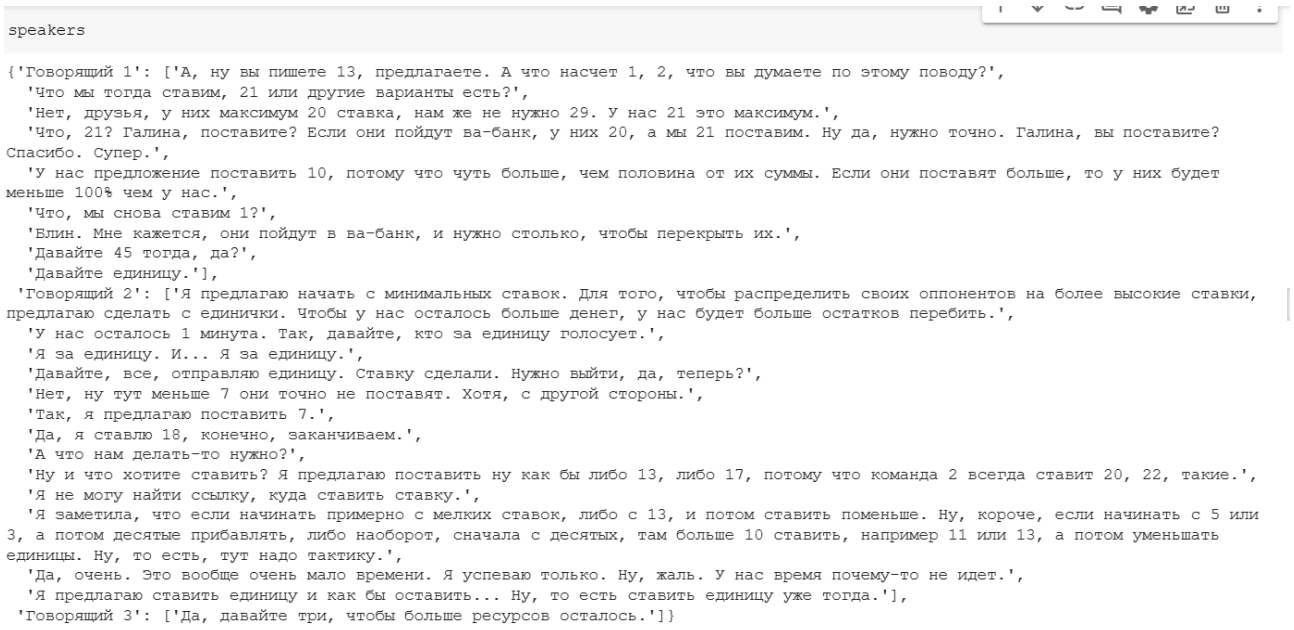
Среди всех моделей, оценка не превышала 66% с учетом дисперсии. Метод оценки с помощью кросс-валидации представлен на листе 3 графической части. Для качественной оценки попытаемся внедрить использование моделей на аутентичном наборе данных: из последней проведенной групповой сессии — и, пользуясь определением «предложения», решим, насколько качественно работают модели.

4 Экспериментальное внедрение работы

Текст пятой групповой работы использовался для проверки работоспособности модели и адекватности оценки поступаемых данных.

4.1 Мешок слов

Последним этапом в применении модели будет ее качественная оценка на неразмеченном наборе данных — последней транскрибированной игре. Она содержит 140 линий реплик; всего говорящих, принимающих участие, трое. Для этого создаем список говорящих в виде словаря, обрабатываем текст с помощью ранее отмеченных функций, и добавляем реплику в словарь только в том случае, сли модель распознала ее как принадлежащей классу «предложение» (листинг В.2). Результат работы данного алгоритма представлен на рисунке 12.



```
speakers

{'Говорящий 1': ['А, ну вы пишете 13, предлагаете. А что насчет 1, 2, что вы думаете по этому поводу?',
  'Что мы тогда ставим, 21 или другие варианты есть?',
  'Нет, друзья, у них максимум 20 ставка, нам же не нужно 29. У нас 21 это максимум.',
  'Что, 21? Галина, поставите? Если они пойдут ва-банк, у них 20, а мы 21 поставим. Ну да, нужно точно. Галина, вы поставите?
Спасибо. Супер.',
  'У нас предложение поставить 10, потому что чуть больше, чем половина от их суммы. Если они поставят больше, то у них будет
меньше 100% чем у нас.',
  'Что, мы снова ставим 1?',
  'Блин. Мне кажется, они пойдут в ва-банк, и нужно столько, чтобы перекрыть их.',
  'Давайте 45 тогда, да?',
  'Давайте единицу.'],
 'Говорящий 2': ['Я предлагаю начать с минимальных ставок. Для того, чтобы распределить своих оппонентов на более высокие ставки,
предлагаю сделать с единички. Чтобы у нас осталось больше денег, у нас будет больше остатков перебить.',
  'У нас осталось 1 минута. Так, давайте, кто за единицу голосует.',
  'Я за единицу. И... Я за единицу.',
  'Давайте, все, отправляю единицу. Ставку сделали. Нужно выйти, да, теперь?',
  'Нет, ну тут меньше 7 они точно не поставят. Хотя, с другой стороны.',
  'Так, я предлагаю поставить 7.',
  'Да, я ставлю 18, конечно, заканчиваем.',
  'А что нам делать-то нужно?',
  'Ну и что хотите ставить? Я предлагаю поставить ну как бы либо 13, либо 17, потому что команда 2 всегда ставит 20, 22, такие.',
  'Я не могу найти ссылку, куда ставить ставку.',
  'Я заметила, что если начинать примерно с мелких ставок, либо с 13, и потом ставить поменьше. Ну, короче, если начинать с 5 или
3, а потом десятые прибавлять, либо наоборот, сначала с десятых, там больше 10 ставить, например 11 или 13, а потом уменьшать
единицы. Ну, то есть, тут надо тактику.',
  'Да, очень. Это вообще очень мало времени. Я успеваю только. Ну, жаль. У нас время почему-то не идет.',
  'Я предлагаю ставить единицу и как бы оставить... Ну, то есть ставить единицу уже тогда.'],
 'Говорящий 3': ['Да, давайте три, чтобы больше ресурсов осталось.']}
```

Рисунок 12 – Логистическая регрессия + мешок слов: результат работы

Большое количество реплик не имеют никакого отношения к категории «предложение». Примеры:

— «нет, друзья, у них максимум 20 ставка, нам же не нужно 29. У нас 21 это максимум»;

— «у нас предложение поставить 10, потому что чуть больше, чем половина от их суммы. Если они поставят больше, то у них будет меньше 100% чем у нас»;

— «нет, ну тут меньше 7 они точно не поставят. Хотя, с другой стороны».

4.2 Плотные векторные представления слов

4.2.1 Navec

Алгоритм для вывода такой же, как в 4.1; однако, используется не функция «vectorize_vector», а обученные на наших данных векторы navec, трансформированные в двумерный массив с размерностью 50. Работа алгоритма на тех же данных представлена на рисунке 13.

Данный результат является более осмысленным, нежели просто скалькулированным, тем не менее некоторые фразы также семантически выбиваются из ранее зафиксированного определения предложения, нацеленного на побуждение к некоему действию:

— «а вот тоже я отправила сообщение тут же»;

— «давайте, все, отправляю единицу. Ставку сделали. Нужно выйти, да, теперь?»;

— «я не могу найти ссылку, куда ставить ставку».

```
[124] speakers

{'Говорящий 1': ['А вот тоже я отправила сообщение тут же.',
  'А, ну вы пишете 13, предлагаете. А что насчет 1, 2, что вы думаете по этому поводу?',
  'Что мы тогда ставим, 21 или другие варианты есть?',
  'У нас предложение поставить 10, потому что чуть больше, чем половина от их суммы. Если они поставят больше, то у них будет меньше 100% чем у нас.',
  'Мы можем поставить чуть больше половины. Просто тогда будет вероятность больше, что мы...',
  'Да, но тут как бы не рискнешь и не победишь, наверное. Четвертая рисковая, окей. Ну у меня зафиксировано, что они начали с 4. Правда, четвертая, по-моему.',
  'Ну что, три, как договаривались? Пятые вроде начали в прошлый раз с 11.',
  'Давайте 45 тогда, да?',
  'Давайте единицу.'],
 'Говорящий 2': ['У нас осталось 1 минута. Так, давайте, кто за единицу голосует.',
  'Давайте, все, отправляю единицу. Ставку сделали. Нужно выйти, да, теперь?',
  'Но я предлагаю 7. Почему? Потому что даже если они поставят чуть больше 8, у них в остатке останется 11, а у нас тут чуть больше монет 21.',
  'Если они перетянут, допустим, если они поставят 8, и  них в остатке останется 11, а у нас 21, то у нас будет преимущество на следующие шаги, у нас останется чуть больше монет. Просто если они сейчас поставят 8, мы поставим 10, и мы перетянем камень, у нас останется 18.',
  'Так, я предлагаю поставить 7.',
  'Да, я ставлю 18, конечно, заканчиваем.',
  'А что нам делать-то нужно?',
  'Ну и что хотите ставить? Я предлагаю поставить ну как бы либо 13, либо 17, потому что команда 2 всегда ставит 20, 22, такие.',
  'Я не могу найти ссылку, куда ставить ставку.',
  'Ну все, я уже тройку отправила. Ладно, согласна с вами. Надо выиграть.',
  'У меня 45 и только ссылочка, пришлите еще раз кого-то.',
  'Я не успела ставку сделать, верните, пожалуйста.',
  'Я предлагаю ставить единицу и как бы оставить... Ну, то есть ставить единицу уже тогда.'],
 'Говорящий 3': ['Предлагаю поставить полностью ставку. Сколько осталось. И мы их выиграем. Они не поставят полностью всю.']}
```

Рисунок 13 – Логистическая регрессия + Navec: результат работы

4.2.2 Word2Vec

Трансформируем транскрибированный текст, используем обученные на наших данных векторы Word2Vec, которые затем применяем к классификатору SVC.

На рисунке 14 представлены данные, которые вывели максимально приближенный к реальности список предложений, подходящие к заданному нами определению. Некоторые высказывания, безусловно, могут не влиять напрямую на ход действий групповой работы, но реплика все равно не перестает являться предложением.

```
[185] speakers

{'Говорящий 1': ['А, ну вы пишете 13, предлагаете. А что насчет 1, 2, что вы думаете по этому поводу?',
  'Что мы тогда ставим, 21 или другие варианты есть?',
  'Что, 21? Галина, поставите? Если они пойдут ва-банк, у них 20, а мы 21 поставим. Ну да, нужно точно. Галина, вы поставите? Спасибо. Супер.',
  'Что, мы снова ставим 1?',
  'Ну, посмотрим. Будет смешно, если не поставить четыре.',
  'Блин. Мне кажется, они пойдут в ва-банк, и нужно столько, чтобы перекрыть их.',
  'Давайте 45 тогда, да?',
  'Давайте единицу.'],
 'Говорящий 2': ['Я предлагаю начать с минимальных ставок. Для того, чтобы распределить своих оппонентов на более высокие ставки, предлагаю сделать с единички. Чтобы у нас осталось больше денег, у нас будет больше остатков перебить.',
  'Так, я предлагаю поставить 7.',
  'Ну и что хотите ставить? Я предлагаю поставить ну как бы либо 13, либо 17, потому что команда 2 всегда ставит 20, 22, такие.',
  'Я заметила, что если начинать примерно с мелких ставок, либо с 13, и потом ставить поменьше. Ну, короче, если начинать с 5 или 3, а потом десятые прибавлять, либо наоборот, сначала с десятых, там больше 10 ставить, например 11 или 13, а потом уменьшать единицы. Ну, то есть, тут надо тактику.',
  'Я предлагаю ставить единицу и как бы оставить... Ну, то есть ставить единицу уже тогда.'],
 'Говорящий 3': ['Да, давайте три, чтобы больше ресурсов осталось.']}
```

Рисунок 14 – Метод опорных векторов + Word2Vec: результат работы

При внедрении обученных алгоритмов на аутентичных данных можно сделать вывод, что лучшие результаты показала модель, использующая векторы Word2Vec в связке с классификатором SVC. Несмотря на довольно средний балл при использовании методов оценки качества модели (ввиду малого количества данных), алгоритм машинного обучения показал наилучший результат, учитывая семантику высказываний при их классификации. Пространство векторов для данной работы представлен на листе 2 графической части.

4.3 Деплой ML-приложения

4.3.1 Технологический стек и инструментарий

В рамках проекта используется следующие инструменты:

— python выбран из-за высокой читаемости кода, богатой экосистемы библиотек для машинного обучения и NLP (таких как Scikit-learn и gensim);

— Word2Vec и SVC — инструменты для обработки текста и классификации. Их выбор обоснован малым размером датасета, для которого Word2Vec может предоставить удобное векторное пространство в рамках всех обученных данных, а SVC отлично справляется с задачей классификации при наличии двух классов и хорошо работает даже с небольшими датасетами;

— Flask — легковесный и гибкий веб-фреймворк для Python. Его простота и возможность быстро развертывать веб-приложения являются ключевыми факторами выбора в контексте разработки машинного обучения.

4.3.2 Описание методики деплоя

Написанное приложение удобно использовать локально, без необходимости развертывания: нужно лишь загрузить транскрипт групповой работы, открыть его и запустить функцию обработки. Тем не менее, для демонстрации работы сервиса как приложения использовался фреймворк Flask. Он позволил быстро и удобно получать и обрабатывать POST-запросы.

Для деплоя приложения машинного обучения использовался сервис Amvera. Его преимущество состоит в том, что за сервисом стоит отечественная организация, для использования которой не возникло препятствий, а также простота в сборке и развертывании. Алгоритм был следующий:

— написать конфигурационный файл «amvera.yaml» или сгенерировать на сервисе;

— создать файл «requirements.txt» со всеми зависимостями проекта;

— воспользоваться выделенным репозиторием или привязать к сервису свой репозиторий;
— загрузить файлы «amvera.yaml», «requirements.txt», «app.py» через веб-интерфейс;
— запустить сборку, а затем развертывание приложения.

Обученный классификатор SVC и подготовленные вектора Word2Vec использовались для основных вычислительных задач в процессе машинного обучения. Эти файлы располагаются в папке в репозитории, которые были скопированы на сервер при развертывании. Приложение включает их при обработке текста.

5 Экономическое обоснование работы

5.1 Описание работы

Работа предполагает внедрение вспомогательного инструмента в систему пост-деятельностной рефлексии с помощью алгоритмов машинного обучения. Данная рефлексия позволяет отследить свое поведение в рамках групповой деятельности, а также то, как менялась динамика взаимодействия после высказываний участников. Инструмент предоставляет сводку высказываний, произнесенных тем или иным спикером, которые входят в категорию «предложение». Технологическое решение является заказом цифровой образовательной платформы EntSpace.

5.2 План производства и организация деятельности

Организация деятельности включает в себя несколько этапов:

— определение требований и исходных материалов;
— выбор методов транскрибации и диаризации видео/аудиозаписей;
— оценка существующих предварительно обученных моделей Word2Vec и их пригодность для текущей задачи;
— постановка экспериментов для начальной оценки производительности стека технологий на текстовой форме групповой работы;
— создание прототипа классификатора SVC, включая функции предобработки данных;
— внедрение базового бэкенда на Flask для тестирования всей системы в целом;
— проведение тестирования и итераций для улучшения точности прототипа.

После подготовки приложения к выполнению базовой функции были проведены шаги для оптимизации и улучшения обрабатываемых данных:

— улучшение точности и производительности SVC с помощью тонкой настройки и валидации;

— интеграция всего функционала в фреймворк Flask с учетом лучших практик безопасности и производительности.

По завершении оптимизации готовый функционал был развернут на сервере.

5.3 Финансовый план

Для расчета финансового плана учтем расходную карту в ходе формирования и эксплуатации проекта в течение трех месяцев после разработки.

Основные пункты расходов перечислены в таблице 1. Далее будут адресованы основные моменты формирования расходной карты.

Учет заработной платы работника должен включать в себя вычет НДФЛ в виде 13% (для резидентов РФ). Отчисления в социальный фонд покрываются работодателем, а именно:

а) 22% на пенсионное обеспечение,

б) 5,1% на медицинское обеспечение,

в) 2,9% на социальное обеспечение по нетрудоспособности и материнству,

г) 5% (условно) на социальное обеспечение от несчастных случаев;

Средняя заработная плата младшего ML-инженера в России составляет 80.000 руб. в месяц (960.000 руб. в год). Для подсчета расходов необходимо учесть формулу (5.1) и выполнить ее пять раз, учитывая отчисления и налоги:

$$Sum = S \cdot \frac{T\%}{100\%}\ , \qquad (5.1)$$

$$Sum = \frac{960000 * 22\% + 960000 * 5{,}1\% + 960000 * 2{,}9\% + 960000 * 5\% + 960000 * 13\%}{100\%}$$

где Sum – сумма расходов с учетом налоговых и социальных отчислений;

S – заработная плата ML-инженера за год;

T – проценты отчислений в социальный фонд.

Итого, выходит 460.800 рублей в год (38400 руб. в месяц). К данным отчислениям необходимо будет прибавить заработную плату сотруднику.

Для использования устройства для обучения модели машинного обучения учитываем амортизацию. Стоимость ПК: 80.000 руб. Допустим, что средний срок службы ПК – 5 лет (60 месяцев). Для расчета издержек по износу оборудования полную стоимость делим на срок службы: 80.000/60 ≈ 1333.3 руб/мес.

В ходе обучения использовался сервис для транскрибации и диаризации групповых работ AssemblyAI. В рамках обучения было использовано около 30 часов аудиозаписей для создания датасетов и тестирования работы. Тарификация сервиса следующая: первые 100 часов обработки – 0 руб/м; после – 0,12 долларов в час ≈ 10,85 руб/ч (по курсу 02.06.2024). Промежуток, за который эти часы использовались, – 67 дней (≈ 2 месяца). Таким образом, на эксплуатацию в ближайшие 3 месяца потребуется 0 рублей, поскольку будет потрачено около 45 часов, что в сумме дает около 75 часов обработки файлов. Это входит в лимит бесплатного тарифа. Кроме того, помимо использования API данного сервиса, на его веб-странице имеется раздел песочницы, позволяющий загружать видео- и аудиофайлы без ограничений, который по окончании обработки также предоставляет текстовый диаризованный транскрипт групповой работы, который впоследствии пригоден для обработки разрабатываемым инструментом.

Далее необходимо описать положительные эффекты от внедрения данного инструмента. Модератору необходимо проводить онлайн групповые сессии, а затем пост-деятельностную рефлексию. Допустим, на подготовку к проведении качественной рефлексии модератор затрачивает время для прослушивания записи работы с целью транскрибации; деятельностные групповые работы проходят раз в неделю, и каждая длится по 2 часа. В среднем, транскрибация часовой записи занимает 4 часа. Таким образом, подготовка к одной рефлексии может занимать около 8 часов в неделю или 32 часа в месяц. При внедрении инструмента, который включает в себя автоматическую транскрибацию текста и выделение важных для рефлексии реплик, пост-обработка групповой сессии

занимает максимум 10 минут (в зависимости от размера загружаемого аудио- или видеофайла вместо 4 часов (240 минут). Допустим, что почасовая ставка модератора – 500 руб. Таким образом, помимо прямых обязанностей работника, компания должна была сокращать пул задач модератора, чтобы включить в нее часы для обработки данных. Потери на данной задаче доходят до 16.000 руб. в месяц и сокращение задач сотрудника. За 3 месяца деятельности экономия будет 48.000 руб., что является месячной заработной платой модератора. Кроме того, участники, покупающие программу, платят за качественную работу, и при проведении качественной рефлексии с большей вероятностью продолжат покупать цифровые продукты организации. Минимальная стоимость участника в программе составляет 75.000 руб. Данная сумма умножается на число принимающих участие – минимум 3 человека. Итого, выгода после внедрения продукта за 3 месяца эксплуатации: $(75.000 \cdot 3 + 16.000) \cdot 3 = 723.000$ руб.

Таблица 1 – Основные пункты расходов при производстве

Единица	Расходы	
	Стоимость единицы, рублей в месяц	Итого за 3 месяца, рублей в месяц
Зарплата начинающего ML-инженера	80.000	240.000
Налоговые и социальные отчисления	38.400	115.200
Интернет	500	1.500
Хостинг на сервисе Amvera	1.450	4.350
ПК для работы: Thunderobot 911 Plus Pro SD	1.333	4.000
AssemblyAI	0	0
Сумма:		365.050

Итого, на поддержание и эксплуатацию инструмента за 3 месяца потребуется 365.050 рублей. Подразумевается, что датасет будет увеличиваться посредством включения большего количества проведенных групповых сессий.

Расходы покрываются компанией-заказчиком. Кроме того, из ранее зафиксированных расчетов, компания уйдет в плюс на 357.950 рублей ввиду условного привлечения большего количества людей посредством предоставления качественного материала и сокращением часов работы модератора.

6 Безопасность и экологичность работы

6.1 Оценка технологической безопасности проектируемой технологии

Проект является цифровым решением, собирающим цифровой след участников и обрабатывающим его при использовании различных библиотек для работы с данными. В ядре проекта предусматривается ведение записи онлайн групповой работы и последующая обработка произнесенных участниками реплик. Таким образом, важным моментом является зафиксировать риски, касаемые безопасности решения и его использования в сети Интернет, экологическую составляющую проекта и профилактику здоровьесберегающих мер поведения.

Современная рабочая среда требует от пользователей длительного времени взаимодействия с электронными устройствами, что значительно увеличивает риски для здоровья, такие как зрительное утомление, мышечно-скелетные нарушения и психологическое напряжение. Поскольку разрабатываемое решение является цифровым, все прямые и косвенные стейкхолдеры оказываются подвержены ранее отмеченным рискам. Далее в главе будут подробно рассматриваться мероприятия для обеспечения безопасности при разработке и использовании продукта.

6.1.1 Разработка мероприятий по охране труда при создании ПО вычислительной техники и автоматизированных систем

Как было отмечено ранее, вопрос безопасности жизнедеятельности при разработке цифрового продукта проявляется в различных сферах. Вследствие, при оценке безопасности важно рассмотреть следующий список превентивных мероприятий:

— первостепенно следует оценить рабочие условия с учетом проектных работ и использования компьютерной техники, исключив факторы, способные оказать неблагоприятное воздействие на здоровье участников;

— в процессе реализации проекта важно обеспечить достаточное освещение рабочих мест, комфортные уровни шума, температуры и приемлемого коэффициента воздушных потоков, а также влажности в помещении;
— стоит выполнять систематические перерывы для снижения нагрузки на органы зрения и опорно-двигательный аппарат.

Оценка категории пожароопасности помещения, в котором разрабатывается указанное ПО, следует провести согласно СП 12.13130.2009, учитывая, что деятельность связана с использованием оборудования класса В (рабочее место с ПК). Необходимо обеспечить наличие достаточного количества средств пожаротушения, а также четко проработать план эвакуации и проведение инструктажей по пожарной безопасности.

Требования к организации рабочего места включают в себя следующее:
— обеспечение достаточного пространства для работы;
— регулируемую мебель, которая позволит поддерживать правильную осанку во время работы;
— оборудование для предотвращения перегрева компьютерной техники.

Для комфортной работы рабочее пространства разработчика должно удовлетворять следующим условиям:
— необходимо обеспечить достаточно свободное пространство под столом, чтобы можно было удобно разместить ноги, не подгибая их;
— рабочий стол для разработчика должен быть такой высоты, чтобы быть удобным для работы и при необходимости использования подлокотников;
— поверхность стола должна быть такая, чтобы исключить появление бликов в поле зрения программиста;
— оптимальная высота рабочей поверхности стола должна находиться в пределах 680-760 миллиметров, а поверхность для клавиатуры лучше всего расположить на уровне около 650 миллиметров [13].

Согласно санитарно-гигиеническим требованиям (СаНПиН), необходимо также поддерживать определенный микроклимат в помещении, обеспечивая

свежий воздух и избегая перепадов температур, а также определенный уровень электромагнитного излучения.

6.2 Оценка экологической безопасности при создании ПО автоматизированных систем

Разработка и использование программного обеспечения, в том числе веб-приложений, оказывает определенное влияние на окружающую среду. Это влияние проявляется на этапах разработки и дальнейшего использования ПО.

6.2.1 Разработка мероприятий по охране окружающей среды при создании ПО вычислительной техники или автоматизированных систем

Для обеспечения безопасного процесса разработки ПО необходимо учесть следующие аспекты:

— расход природных ресурсов. Для создания серверного оборудования и других компонентов используются значительные объемы натуральных ресурсов, включая металлы, которые могут быть дефицитными. Добывающие и производственные работы часто сопровождаются экологическим ущербом. Чтобы минимизировать такое воздействие, следует продлевать жизнь оборудованию, использовать его эффективно, а также применять переработанные материалы и правильно утилизировать устаревшую аппаратуру;

— энергопотребление и теплоотдача. Создание ПО требует ресурсоемких вычислительных операций, что вызывает увеличение потребления энергии и тепловыделения. Вследствие этого повышается нагрузка на системы охлаждения и, как следствие, возможен рост выбросов парниковых газов. Чтобы этого избежать, нужно оптимизировать работу программного обеспечения путем осознанного подхода к тестированию приложений и запуску скриптов, а также использовать энергоэффективное оборудование и технологии виртуализации;

— электромагнитные излучения. Значительное воздействие на экологию оказывает электромагнитное излучение от сервисных устройств, сетевого оборудования и других электронных компонентов. Чтобы его минимизировать, применяют экранированные провода и специальные корпуса, а также осуществляют контроль уровня излучения и аккуратно планируют расположение устройств, чтобы электромагнитные поля не пересекались;

— воздействие IT-сектора на углекислый след. Суммарный выпуск парниковых газов, связанный с производством и использованием IT-устройств и ПО, формирует углеродный след. Сюда входят выбросы, возникающие на всех этапах жизненного цикла оборудования. Снизить углеродный след можно разработкой ПО, которое потребляет меньше ресурсов, применением энергоемкой аппаратуры, использованием облачных технологий с высокой энергоэффективностью и переходом на возобновляемые источники энергии. Также применяются инструменты для мониторинга и оценки углеродного следа, которые помогают городским сообществам снизить выбросы углерода.

Для сокращения углеродного следа предприятиям стоит сосредоточиться на оптимизации хранения и обработки данных. В первую очередь рекомендуется регулярно проводить аудит данных, чтобы определить «темные данные» – информацию, которая постоянно хранится, но на самом деле не используется. Это поможет устранить избыточное хранение данных и сэкономить ресурсы.

Далее, компаниям следует стремиться к переходу на зеленые технологии в центрах обработки данных, такие как использование энергии из возобновляемых источников и внедрение более эффективных систем охлаждения, которые снижают общее энергопотребление.

Еще один важный шаг — оптимизация инфраструктуры для обработки больших данных с применением более эффективных алгоритмов, позволяющих выполнять необходимые операции быстрее и с меньшим энергопотреблением.

Итак, стимулирование разработки и применения программного обеспечения, снижающего необходимость постоянной обработки и хранения данных, может оказаться действенным способом уменьшения воздействия на окружающую среду. Внедрение упомянутых мер не только способствует сокращению углеродного следа, но также может привести к снижению операционных расходов и повышению общей эффективности обработки данных. Выше обозначенные меры являются важными этапами в качественном обеспечении экологической безопасности разрабатываемого проекта [14].

6.3 Расчетная часть

Оценив важные меры для организации безопасного рабочего места разработчику, необходимым является выполнить самостоятельные замеры условий окружающей среды и соотнести с допустимыми значениями. Выполняются замеры инсоляции (прибор Эко-Е), а также электромагнитного и радиационного излучений в рабочем помещении (прибор СОЭКС) для оценки пригодности рабочего пространства.

Прибор Эко-Е предназначен для измерения уровня освещенности (люксметр) и инсоляции. Он использует фотодетектор, который преобразует свет в электрический сигнал. Этот сигнал обрабатывается и отображается на дисплее в виде показателя освещенности в люксах. Для измерения инсоляции прибор устанавливается в точке замера, ориентируется в направлении источника света и производится считывание показателей при различных условиях освещения (естественном и комбинированном).

Прибор СОЭКС (нитратометр, экотестер, дозиметр) измеряет уровни электромагнитного и радиационного излучений. Он оснащен датчиками для регистрации радиационного фона и электромагнитных полей. Датчики фиксируют излучение, преобразуют его в электрические сигналы, которые обрабатываются и отображаются на экране прибора. Для замера

электромагнитного и радиационного излучений прибор помещается в точку замера, и считываются показатели на экране.

Показатели должны не превышать допустимые уровни. Согласно СанПиН 2.2.1/2.1.1.1278-03, минимально допустимый уровень освещенности на рабочем месте может варьироваться от 300 до 500 люкс. Касаемо электромагнитного излучения, в соответствии с СанПиН 2.1.2.1002-00 «Допустимые уровни электромагнитного излучения радиочастотного диапазона в жилых помещениях», предельно допустимые уровни магнитных полей промышленной частоты (50 Гц) в рабочей зоне не должны превышать 10 мкТл. Еще одним из условий приемлемого содержания рабочего места является предельно допустимый уровень радиационного фона для рабочих помещений, который, согласно СанПиН 2.6.1.2523-09, составляет 0,2 мкЗв/ч.

Результаты замеров в четырех точках рабочего пространства представлены в таблице 2.

Таблица 2 – Замеры инсоляции и электромагнитного и радиационного излучений

№ замера	Естественное освещение, люкс	Комбинированное освещение, люкс	Электромагнитное излучение, мкТ	Радиоактивное излучение, мкЗв/ч
1	300	450	6,1	0,18
2	350	440	0,5	0,13
3	320	460	1,2	0,13
4	420	500	0,8	0,16

Во всех точках значения освещенности (естественной и комбинированной) выше минимального уровня. Данные показатели соответствуют нормам для большинства типов работ, включая работу за компьютером. Касательно электромагнитного излучения, уровни в точках замера варьируются от 0.8 до 6.1мкТ, что ниже предельно допустимого уровня. Показатели радиационного

излучения составляют 0,1 мкЗв/ч, что также ниже допустимого предела. Таким образом, все замеры соответствуют нормам.

Во всех точках замера все показатели попадают в допустимый диапазон, что говорит об обеспечении безопасного рабочего места при разработке ML-инструмента.

В ходе выпускной квалификационной работы был проведен анализ важных мер безопасности и экологической устойчивости при разработке инструмента машинного обучения. Основное внимание уделено созданию безопасного рабочего пространства для разработки проекта, эффективному использованию природных ресурсов, снижению энергопотребления, минимизации электромагнитного излучения, а также уменьшению углеродного следа, связанного с IT-сектором. Дополнительным параметром было проведение замеров инсоляции и электромагнитного и радиационного излучений в рабочем пространстве, полностью удовлетворяющим допустимым нормам. Безопасность рабочего места гарантирует защиту как здоровья разработчика, так и окружающей среды.

Заключение

Выпускная квалификационная работа на тему «Извлечение семантического ядра русскоязычного текста с помощью алгоритмов машинного обучения» была выполнена в полном объеме, а именно:

— сбор и разметка датасета по окончании деятельностно-игровых сессий «Скифская лестница», проводимых компанией EntSpace;
— токенизация текста и подготовка токенов к векторизации;
— тестирование различных методов кодирования токенов и финальная остановка на плотных векторах Word2Vec;
— тестирование классификаторов и остановка на методе опорных векторов SVC;
— применение модели на практике по окончании групповой сессии и оценка работы проекта;
— деплой ML-решения.

Среди недостатков решения выделяются следующие моменты:

— малый набор данных для обучения, ограничивающий выбор инструментария;
— малое количество классов для рефлексии участниками, что предоставляет неполную картину деятельности по ее завершении;
— контекстуальность решения. Данный инструмент натренирован и нацелен на групповую деятельностную игру «Скифская лестница», проводимую EntSpace. Для универсальности решения может быть предусмотрено дальнейшее развитие проекта в сторону предоставления более качественного материала для рефлексии по окончании именно данного типа групповой работы, либо сделан пивот в сторону групповой работы как таковой.

Для повышения качества работы системы может рассмотреться увеличение как датасета, так и классов (например, рисунок 5). Кроме того, внедрение LLM-моделей (таких как ChatGPT от OpenAI, или GigaChat от Сбербанка) может донести данные по завершении групповой работы в «человеческом» языке, более простым для восприятия и, как следствие, анализа.

Более того, внедрение веб-интерфейса, связывающего все используемые сервисы, также бы упростило работу эксплуатации инструмента модератором.

Тем не менее, нынешнее состояние ML-решения позволяет на ранних этапах зафиксировать участнику ключевые этапы групповой деятельности, проанализировать свои действия, соотнести их с ходом самой работы и отрефлексировать итоги.

Пост-деятельностная рефлексия является абстрактным, но эффективным инструментом личностного развития. Данный метод позволяет оценить свои действия со стороны по завершении деятельности, в чем использование спроектированного проекта оказывается полезным вспомогательным инструментом, сохраняющим ресурсы модераторов групповых работ и помогающим вспомнить ключевые моменты, которые повлияли на динамику взаимодействия.

Перечень использованных информационных ресурсов

1. Стандарт цифрового следа : [сайт]. — URL: https://standard.2035.university/v1.0.2. (дата обращения: 10.01.2024). — Текст : электронный.

2. Cronin Gerard, Andrews Steven. After action reviews: A new model for learning / Emergency nurse: the journal of the RCN Accident and Emergency Nursing Association. — 2009. — Vol. 17. — P. 32-35. — DOI: 10.7748/en2009.06.17.3.32.c7090.

3. Partee B. Semantics / The MIT Encyclopedia of the Cognitive Sciences / eds. R. A. Wilson, F. C. Keil. — Cambridge, MA: The MIT Press, 1999. — Pp. 739–742.

4. Русский толковый словарь РАН : [сайт] / под ред. В. В. Лопатина. — М.: Справочно-информационный интернет-портал ГРАМОТА.РУ, 2005. — URL: http://www.gramota.ru. (дата обращения: 04.06.2024). — Текст : электронный.

5. Reyna V. A New Intuitionism: Meaning, Memory, and Development in Fuzzy-Trace Theory / Judgment and decision making. — 2012. — Vol. 7. — DOI: 10.1017/S1930297500002291.

6. Valamis : [сайт]. — URL: https://www.valamis.com/ (дата обращения: 04.06.2024). — Текст : электронный.

7. Vosaic : [сайт]. —URL: https://vosaic.com/ (дата обращения: 04.06.2024). — Текст : электронный.

8. Voyant Tools : [сайт]. — URL: https://voyant-tools.org/ (дата обращения: 04.06.2024). — Текст : электронный.

9. Бизнес-игра «Скифская лестница» : [сайт] — URL: https://ru.entspace.com/programs/humantech-praktikum-po-igrotehnike/biznes-igra-skifskaya-lestnica (дата обращения: 17.05.2024). — Текст : электронный.

10. Булыгин И. (науч. рук. Сухаренко Д.В.) Машинные алгоритмы анализа социальных сетей / Сборник тезисов докладов конгресса молодых ученых. Электронное издание. — СПб: Университет ИТМО, 2020. — Режим доступа: https://kmu.itmo.ru/digests/article/5167.

11. Харрисон М. Машинное обучение: карманный справочник. Краткое руководство по методам структурированного машинного обучения на Python / пер. с англ. — СПб.: ООО «Диалектика», 2020. — 320 с.: ил. — ISBN 978-5-907203-17-4.

12. Миколов Т., Чен К., Коррадо Г., Дин Дж. Эффективная оценка представлений слов в векторном пространстве / arXiv:1301.3781. — 2013. — DOI: 10.48550/arXiv.1301.3781.

13. Петровец В. М., Винокуров А. А. Эргономические требования к проектированию рабочего места программиста. — 2022.

14. Лиханова Е. Как сократить цифровой углеродный след организации / Лиханова Е. : [сайт] / RB.RU. – Режим доступа: https://rb.ru/story/dark-data-decarbonisation/ (дата обращения: 04.06.2024). — Текст : электронный

Приложение А

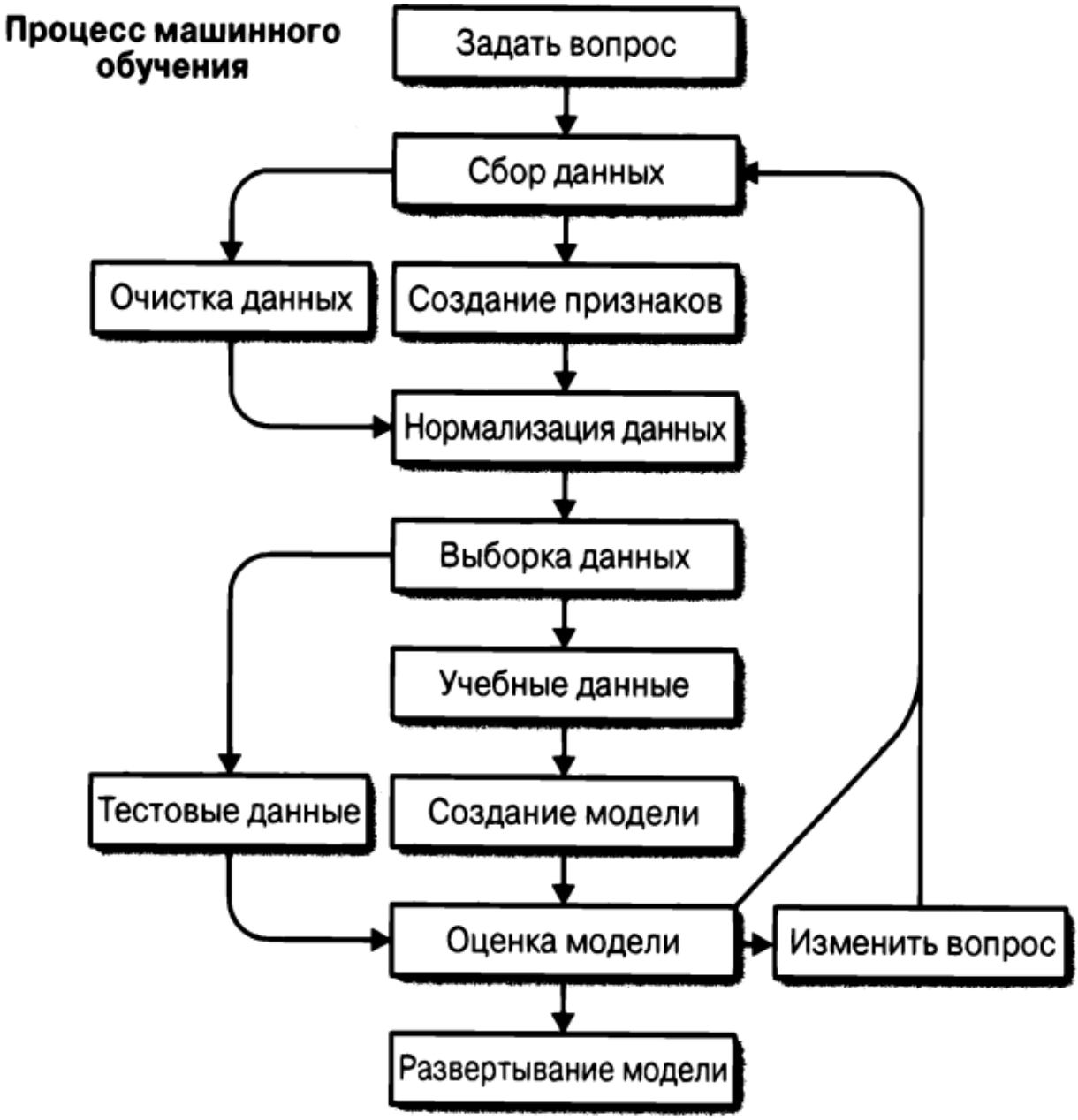


Рисунок А.1 – Общий рабочий процесс машинного обучения

Приложение В

Листинг В.1 – Функция «Векторизация токенов» с помощью библиотеки Navec (для языка Python 3.10)

```
def vectorize_text(txt, navec, max_phrase_len):
    unk = navec['<unk>']
    text_embeddings = []
    for token in txt:
        embedding = navec.get(token, unk)
        text_embeddings.append(embedding)
    # Дополняем или обрезаем реплики для фиксированной длины
max_review_len
    l = len(text_embeddings)
    if l > max_phrase_len:
        text_embeddings = text_embeddings[:max_phrase_len]
    else:
        text_embeddings.extend([navec['<pad>']] * (max_phrase_len
- l))
    return text_embeddings

dataframe['embeddings'] = dataframe.apply(lambda row:
vectorize_text(row['preprocessed_text'], navec, max_phrase_len),
axis=1)
```

Листинг В.2 – Программа формирования ответа по окончании анализа

```
speakers = dict()
test = ['140 строк реплик']
# отделяем спикера от его реплики посредством нахождения знака
двоеточия
for i in test:
    speaker, utterance = i.split(': ')
    speaker = speaker.strip() # 'Speaker A'
    utterance = utterance.strip() # '*text*'

    # приводим реплику в нормальный вид и векторизуем

    utterance_preprocessed = preprocess(utterance, stop_words,
punctuation_marks, morph)
    utterance_seq = text_to_sequence(utterance_preprocessed,
word_to_index)
    utterance_bow = vectorize_sequences([utterance_seq],
max_words)

    # предсказываем категорию высказывания

    prediction = lr.predict(utterance_bow)

    # добавляем спикера в словарь (если его еще нет)

    if speaker not in speakers:
      speakers[speaker] = []

    # если классификатор распознал реплику как предложение,
присваиваем ее спикеру

    if prediction:
      speakers[speaker].append(utterance)
```