

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

**«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИМЕНИ Н. Г. ЧЕРНЫШЕВСКОГО»**

Кафедра математической кибернетики и компьютерных наук

**СОЗДАНИЕ ЧАТ-БОТА ДЛЯ ИНФОРМАЦИОННОЙ ПОДДЕРЖКИ
СТУДЕНТОВ СГУ С ИСПОЛЬЗОВАНИЕМ НЕЙРОННОЙ СЕТИ**

АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ

Студентки 4 курса 411 группы
направления 02.03.02 — Фундаментальная информатика и информационные
технологии
факультета КНиИТ
Гворовой Ксении Владимировны

Научный руководитель

к. ф.-м. н., доцент

Ю. Н. Кондратова

Заведующий кафедрой

к. ф.-м. н., доцент

С. В. Миронов

Саратов 2024

ВВЕДЕНИЕ

В настоящее время информационные технологии активно развиваются и внедряются во все сферы нашей жизни. Одной из таких сфер является образование. Студенты сегодня ожидают быстрого и удобного доступа к информации, которая поможет им в учебе и повысит их эффективность. Однако, поиск нужной информации может быть трудоемким и затратным процессом.

Саратовский государственный университет является ведущим вузом Саратова, который реализует множество программ высшего образования. Поток студентов и абитуриентов, которые нуждаются в информационной поддержке со стороны вуза, постоянно растет. Эту поддержку обеспечивает сайт `sgu.ru`, но иногда найти нужную информацию бывает крайне сложно. Вследствие чего приходится искать контакты представителей отдельных подразделений ВУЗа и связываться с ними лично, чтобы узнать интересующую информацию. Это в свою очередь увеличивает время ожидания ответа, а также нагрузку на сотрудников университета. Использование чат-ботов позволяет многократно сократить время ожидания, а также оградить сотрудников от многочисленных вопросов на одну и ту же тему, что способствует повышению коэффициента полезного действия работников и серьезной экономии времени пользователей.

В связи с этим, создание чат-бота для информационной поддержки студентов становится актуальной задачей. Чат-бот позволит оптимизировать деятельность сотрудников университета, вынужденных затрачивать большое количество времени на трансляцию информации, содержащуюся в открытых источниках.

Целью данной работы является создание чат-бота на языке программирования Python для информационной поддержки студентов и абитуриентов Саратовского государственного университета имени Н. Г. Чернышевского, который способен упрощать поиск необходимой информации об оплате обучения, получении стипендии, графиках учебного процесса, военной кафедре и т.д.

Для достижения поставленной цели необходимо решить следующие задачи:

1. Изучить основы чат-ботов и их применение в информационной поддержке студентов.
2. Проанализировать требования к функциональности чат-бота для студентов

3. Разработать и реализовать алгоритмы обработки запросов
4. Протестировать, оптимизировать и поддерживать чат-бота для информационной поддержки студентов.

Актуальность исследования обусловлена необходимостью обеспечения эффективной и удобной коммуникации между студентами и университетом. В современном мире, где технологии играют все более важную роль, использование чат-бота может значительно улучшить доступность и оперативность предоставления информации студентам. Бот сможет отвечать на часто задаваемые вопросы, предоставлять актуальные новости и объявления, помогать в организации учебного процесса и т.д. Такой инструмент позволит снизить нагрузку на сотрудников университета, сократить время ожидания ответов и повысить качество обслуживания студентов.

Структура и объём работы. Для решения поставленных задач выполнена выпускная квалификационная работа, включающая в себя введение, 2 основные главы, заключение, список использованных источников из 21 наименования и 7 приложений. Работа изложена на 71 странице, содержит 20 рисунков.

Первая глава имеет название «Обзор чат-ботов и методов их создания» и содержит основную теоретическую информацию о технологиях и подходах, которые использовались при разработке чат-бота.

Вторая глава имеет название «Разработка чат-бота», данная глава содержит подробное описание процесса выполнения работы.

Выпускная квалификационная работа заканчивается заключением, списком использованных источников, а также приложениями с кодом А-Ж.

1 Основное содержание работы

1.1 Обзор чат-ботов и методов их создания

1.1.1 Определение чат-ботов и история их развития

Чат-боты и голосовые помощники стали неотъемлемой частью современного цифрового взаимодействия, обеспечивая мгновенную поддержку и автоматизацию задач в различных областях, от клиентского обслуживания до образования. Их развитие благодаря искусственному интеллекту и машинному обучению позволяет создавать ботов, способных вести сложные диалоги и понимать контекст запросов. История чат-ботов началась с теста Тьюринга в 1950 году, ставшего отправной точкой для разработки технологий имитации человеческого общения. Проекты такие, как ELIZA, PARRY и A.L.I.C.E., заложили основы для последующих достижений в области симуляции разговоров. Современные голосовые помощники, такие как Siri и Alexa, успешно интегрированы в повседневные устройства и используют машинное обучение для выполнения команд и поиска информации. В России популярны голосовые помощники Алиса и Маруся, которые активно развиваются, интегрируясь в экосистемы своих компаний-разработчиков и упрощая повседневные задачи пользователей. Таким образом, развитие чат-ботов и голосовых помощников отражает значительный прогресс в области искусственного интеллекта и обработки естественного языка, улучшая качество цифрового взаимодействия и повседневной жизни людей.

1.1.2 Классификация чат-ботов

Чат-боты имеют два основных типа классификации: бизнес-классификацию, которая включает разговорные боты, ассистентов и QA-боты, и классификацию по техническим характеристикам, к которой относятся боты на основе бизнес-правил, боты на основе искусственного интеллекта и гибридные боты. Разговорные боты имитируют неструктурированное общение, в то время как ассистенты выполняют конкретные задачи, извлекая информацию из ответов пользователей. QA-боты заменяют разделы FAQ, предоставляя ответы на вопросы пользователей. Боты на основе бизнес-правил используют заранее определённые сценарии и варианты ответов, тогда как боты на основе искусственного интеллекта используют нейронные сети и обработку естественного языка для формирования диалога на основе обучающих данных. Гибридные боты комбинируют элементы обоих типов, что делает их особенно эффективными

в коммерческих приложениях.

1.1.3 Чат-боты в мессенджерах

Мессенджеры стали неотъемлемой частью современной жизни, предоставляя удобные средства коммуникации, включая текстовые сообщения, медиафайлы, аудио- и видеозвонки, и групповые чаты. Их популярность обусловлена доступностью, интуитивно понятным интерфейсом, возможностью синхронизации данных между устройствами, а также защитой конфиденциальности через шифрование сообщений.

С начала 2010-х годов мессенджеры стали платформой для развития чат-ботов, которые добавили новую функциональность, включая автоматизированные сервисы, такие как получение информации, заказ товаров и общение с персональными ассистентами. Telegram, самый популярный мессенджер в России, благодаря своей мощной функциональности, высокой скорости работы, надежности, мощному API и высоким стандартам безопасности, идеально подходит для создания чат-ботов.

Чат-боты были первоначально внедрены банками для улучшения обслуживания клиентов и снижения издержек. Позже они нашли применение в IT и телекоммуникационных компаниях, а также в сфере обслуживания, улучшая взаимодействие с клиентами и автоматизируя процессы. В последние годы чат-боты также активно внедряются в образование для поддержки студентов, предоставления информации и оптимизации учебного процесса.

1.1.4 Использование чат-ботов в сфере образования

Использование чат-ботов в образовательных учреждениях становится актуальным для решения информационных задач. Чат-боты помогают студентам быстро получать ответы на часто задаваемые вопросы, что разгружает сотрудников и экономит время. Они могут быть интегрированы в сайты, приложения или социальные сети, обеспечивая круглосуточный доступ к информации, быстрое реагирование, малое потребление трафика и автоматизацию рутинной работы. Примеры успешного применения включают помощников в СПбГУ, КузГТУ и ВШЭ, которые помогают абитуриентам и студентам с информацией о поступлении, расписании и других вопросах, повышая эффективность взаимодействия с университетом.

1.1.5 Методы и подходы к созданию чат-ботов

Разработка чат-ботов осуществляется с использованием различных подходов, которые зависят от их классификации. Основными методами являются детерминированные системы и интеллектуальные системы, основанные на машинном обучении и обработке естественного языка (NLP).

Детерминированные чат-боты работают по predetermined правилам и сценариям, не обладая способностью к самообучению. Их разработка включает создание сценариев и логических деревьев диалога, где каждое решение ведёт к определённому ответу или следующему шагу. Важным этапом является составление базы ответов на возможные запросы пользователей и определение ключевых слов и фраз для активации сценариев.

Интеллектуальные чат-боты используют технологии NLP и машинное обучение. NLP помогает понять и генерировать текст на естественном языке через процессы токенизации, поиска сущностей, синтаксического и семантического анализа. Машинное обучение позволяет обучать чат-ботов на больших наборах данных, применяя методы классификации, кластеризации и ранжирования.

1.1.6 Применения нейронных сетей для создания чат-ботов

Использование нейронных сетей в разработке чат-ботов значительно улучшает их адаптивность и интеллектуальные способности. Нейронные сети, такие как рекуррентные нейронные сети (RNN), включая их усовершенствованные версии LSTM и GRU, а также современные модели трансформеров (например, BERT и GPT), позволяют чат-ботам лучше понимать естественный язык и генерировать осмысленные ответы. Разработка таких чат-ботов начинается с тщательного сбора и подготовки данных, выбора подходящей архитектуры нейронной сети и обучения модели с использованием фреймворков глубокого обучения. После обучения модель оценивается и интегрируется в систему чат-бота. Завершающим этапом является постоянный мониторинг и обновление чат-бота для поддержания его актуальности и эффективности.

1.1.7 Обзор нейронной сети Bert

BERT (Bidirectional Encoder Representations from Transformers) – это модель для обработки естественного языка, основанная на двунаправленной архитектуре трансформеров. Она предназначена для выполнения различных задач NLP, таких как определение тональности текста, системы вопрос-ответ,

классификация текстов и анализ текста. BERT обучается на задачах моделирования языковых масок и предсказания следующего предложения, используя корпуса текстов из Wikipedia и BookCorpus. Эта модель позволяет разрабатывать эффективные AI-программы для обработки естественного языка, включая создание чат-ботов, автоматический перевод и текстовый анализ. Google выпустил BERT с открытым кодом на платформе TensorFlow, предоставив несколько многоязычных моделей. Для обработки текста BERT сначала разбивает его на токены, соответствующие словарю модели, и предсказывает контекст и связь между предложениями.

1.2 Разработка чат-бота

1.2.1 Анализ требований

Для разработки чат-бота для информационной поддержки студентов важен этап сбора требований, начавшийся с анализа потребностей студентов и сотрудников. Опросы выявили основные запросы, такие как информация о справках, оплате обучения, новостях университета, военной подготовке, пересдачах, расписании занятий, стипендиях, графике обучения, общежитии и расположении корпусов. Эти данные определили функциональность будущего чат-бота, который будет предоставлять информацию, направлять на дополнительные ресурсы и облегчать взаимодействие с сотрудниками университета и студентов.

1.2.2 Обоснование и выбор инструментов создания чат-бота

На сегодняшний день существует множество языков программирования для создания чат-ботов, таких как Java, Python и PHP, каждый из которых имеет свои преимущества и недостатки. Python выделяется своей простотой синтаксиса, широким набором библиотек и универсальностью, позволяя легко автоматизировать различные задачи. Для разработки телеграм-ботов на Python часто используются библиотеки aiogram, requests для HTTP-запросов, bs4 для парсинга HTML/XML, а также для работы с данными используются библиотеки numpy, pandas, seaborn, matplotlib, sklearn, re, pytorch, transformers, datasets и spaCy.

1.2.3 Регистрация чат-бота

Разработка чат-бота для Telegram включает два этапа: создание оболочки и программирование функционала. Первый этап начинается с регистрации

чат-бота в Telegram через «TelegramBot», который выдает уникальный токен и URL-адрес доступа. После настройки профиля чат-бота, можно приступить к написанию программной части на Python, используя необходимые библиотеки.

1.2.4 Проектирование чат-бота

В разработке чат-бота ключевые этапы включают настройку логирования на уровне INFO для отслеживания важных сообщений, ошибок и предупреждений, необходимых для отладки и мониторинга, и создание объекта MemoryStorage для хранения состояний диалогов с пользователями, обеспечивая сохранение прогресса и контекста общения. Бот инициализируется с уникальным токеном для взаимодействия с Telegram API, позволяя отправлять и получать сообщения. Для обработки сообщений создается объект Dispatcher, связанный с MemoryStorage, что упрощает распределение запросов и управление ими, дополнительно настраиваясь с LoggingMiddleware для подробного логирования обновлений и действий бота, что важно для отслеживания и исправления ошибок. Инициализируется переменная selected_faculty_link для хранения ссылок на выбранные пользователем факультеты, оптимизируя доступ к информации, и переменные tokenizer и model для работы с машинным обучением и обработки текста. Наконец, для удобства пользователей настраивается клавиатура с кнопкой «/start», упрощающая начальное взаимодействие с ботом. Когда пользователь взаимодействует с Telegram-ботом впервые или не знает всех доступных команд, клавиатура с кнопкой «/start» предоставляет удобный способ отправить команду. Функция parse_faculties парсит веб-сайт, извлекая и сокращая названия факультетов в словарь с соответствующими ссылками, а create_faculties_keyboard создает инлайн-клавиатуру, где каждая кнопка представляет факультеты с возможностью переключения страниц. get_confirm_keyboard генерирует клавиатуру с двумя кнопками "Да" и "Нет". Обработчик callback-запросов aiogram управляет выбором факультетов и навигацией по страницам. handle_confirmation обрабатывает ответы «Да» или «Нет» от пользователя, предоставляя возможность завершить или переслать диалог с ботом.

1.2.5 Проектирование и обучение нейронной сети

Первым этапом обучения нейронной сети является сбор и подготовка данных. Необходимо собрать достаточное количество репрезентативных дан-

ных для обучения модели. Для разработки чат-бота был выбран набор данных с Kaggle, после чего данные были обработаны: текст приведен к нижнему регистру, удалены цифры и знаки препинания.

Для анализа данных была выполнена визуализация распределения классов, выявлен дисбаланс. Созданы словари для соотнесения числовых меток с текстовыми описаниями классов. После проверки уникальных значений меток в данных были удалены строки с отсутствующими метками.

Затем определены вспомогательные функции для преобразования данных в формат, поддерживаемый библиотекой transformers, и вычисления весов классов для учета дисбаланса. Данные токенизированы с помощью BERT, созданы датасеты для обучения и тестирования. Реализован класс CustomTrainer для специфического вычисления потерь.

Модель обучалась с использованием предварительно обученного токенизатора и модели BERT для многоязычной классификации. После обучения выведен отчет о классификации, включая точность, полноту, F1-меру и матрицу ошибок. Визуализированы потери во время обучения, что помогло оценить процесс обучения и предотвратить переобучение.

Исследование метрик показало высокую точность классификации в целом, но выявлены классы с менее точными прогнозами, особенно в случае несбалансированных данных.

1.2.6 Общий алгоритм работы и тестирование нейронной сети

При классификации текста сначала он проходит этап предварительной обработки, включающий приведение к нижнему регистру, удаление цифр, знаков препинания и специальных символов, а также очистку от лишних пробелов. Затем текст токенизируется с помощью предварительно загруженного токенизатора BERT и подается на вход модели. Модель BERT работает в режиме оценки, генерируя вероятности для каждого класса с использованием функции софтмакс на ее выходах. На основе этих вероятностей выбирается класс с наибольшей вероятностью, который присваивается тексту как предсказанный. Таким образом, результатом работы модели являются вероятности принадлежности текста к каждому классу и метка предсказанного класса.

Нейронная сеть демонстрирует хорошие результаты с точностью 88 процентов (accuracy), что указывает на ее способность правильно классифицировать примерно 9 из 10 случаев. Среднее значение метрик precision (0.87), recall

(0.80) и F1-score (0.82) по всем классам (macro avg) показывает сбалансированность модели в распознавании различных классов, с небольшим улучшением в точности по сравнению с полнотой. Взвешенное среднее precision, recall и F1-score также составляет 0.88, что подчеркивает эффективность модели на несбалансированных данных.

1.2.7 Создание функции для поиска расписания по группам и преподавателям

Была создана функция `extract_entities` для извлечения именованных сущностей и номеров групп из текста. Функция использует библиотеку `sraCy` для обработки текста и извлечения именованных сущностей. Также применяются регулярные выражения для поиска трехзначных номеров групп. Результатом работы функции является список, содержащий фамилии преподавателей и номера групп.

Для получения расписания группы и преподавателя с веб-страницы университета реализованы функции `get_group_schedule_link`, `get_teacher_page_url` и `get_teacher_schedule_links`. Они используют библиотеку `BeautifulSoup` для парсинга HTML страницы, находят ссылки с номером группы или фамилией преподавателя и возвращают соответствующие URL-адреса.

Для Telegram-бота определен обработчик команды `«/start»` и класс `Form` для управления состояниями взаимодействия с пользователем. Включается состояние `confirm_response` для хранения ответа пользователя на запрос подтверждения.

Также реализованы функции `send_welcome`, `send_response` для отправки сообщений пользователю и обработки их ответов. Функции проверяют наличие загруженной модели для предсказаний и используют predefined ответы в случае необходимости.

1.2.8 Тестирование чат-бота

Процесс тестирования чат-бота в Telegram включал проверку функциональности, начиная с инициирования бота пользователем через кнопку "Старт до получения ответов на различные запросы. Пользователь мог задавать вопросы о расположении корпусов, оплате обучения, графиках учебного процесса, новостях университета, военной кафедре, получении справок, общежитии, расписании группы и пересдачах. Бот предоставлял текстовые ответы с соответству-

ющими ссылками и поддерживал взаимодействие с пользователем кнопками «Да» или «Нет» для подтверждения или опровержения предложенной информации. Тестирование подтвердило корректность работы бота и его способность предоставлять актуальные данные пользователю.

ЗАКЛЮЧЕНИЕ

В данной дипломной работе был рассмотрен процесс создания чат-бота для информационной поддержки студентов с применением нейронных сетей. В ходе выполнения работы были изучены основы проектирования чат-ботов, рассмотрены различные библиотеки и рассмотрены нейронные сети наиболее подходящие для чат-ботов. Также проанализирован теоретический материал, содержащий основные принципы разработки чат-бота и поиска информации по сайту, изучено влияние чат-ботов на образование и их возможности для улучшения информационной поддержки студентов.

Основные результаты и выводы работы позволяют сделать следующие утверждения:

Нейронные сети демонстрируют высокую эффективность в создании чат-ботов, способных обрабатывать естественный язык и предоставлять точные и информативные ответы на запросы студентов.

Созданный чат-бот успешно интегрируется в существующие информационные системы учебных заведений и обеспечивает непрерывную поддержку студентов в режиме реального времени.

Результаты тестирования показывают, что использование чат-бота значительно сокращает время получения необходимой информации студентами и улучшает их общий опыт взаимодействия с учебным процессом.

Дальнейшие исследования и разработки могут быть направлены на расширение функциональности чат-бота, персонализацию ответов и улучшение алгоритмов обучения нейронных сетей на основе обратной связи пользователей.

В процессе выполнения работы был создан чат-бот со следующими возможностями:

- Предоставление информации о графике учебного процесса
- Предоставление информации об оплате обучения
- Предоставление информации о прохождении военной кафедры
- Предоставление информации о получении стипендии
- Предоставление информации о пересдачах
- Предоставление информации о расположении корпусов
- Предоставление информации о новостях факультетов и университета
- Предоставление информации об общежитии

- Предоставление информации о получении справок
 - Предоставление информации о расписании по группам и преподавателям
- Таким образом, поставленные задачи были решены и цель достигнута.