

РАЗРАБОТКА МНОГОФУНКЦИОНАЛЬНОГО TELEGRAM-БОТА КАК ЕДИНОГО ШЛЮЗА К РАЗНООБРАЗНЫМ НЕЙРОСЕТЯМ

DEVELOPMENT OF A MULTIFUNCTIONAL TELEGRAM BOT AS A UNIFIED GATEWAY TO DIVERSE NEURAL NETWORKS

**D. Pavlyuk
I. Yakovlev**

Summary. This article is dedicated to the development of a multifunctional Telegram bot that acts as a unified gateway to various neural network technologies. The primary objective of the research is to address the pressing problems of fragmentation and geographical restrictions faced by users of AI services. The project aims to create a centralized platform that integrates both official APIs from leading companies (OpenAI, Google, Anthropic) and numerous third-party providers under a single, intuitive interface, making advanced technologies accessible to a wide audience. To achieve this goal, a modular asynchronous architecture was implemented in Python, utilizing the Aiogram 3 framework for Telegram API interaction and an SQLite database for data storage. The system is designed for deployment in Docker containers, enabling it to function effectively as a proxy gateway. As a result, a bot with extensive multimodal capabilities was developed: it supports text dialogues, image generation, editing, and analysis, audio transcription, and text extraction from documents. Special attention is given to the project's key outcome: the creation of a full-featured administrative interface directly within Telegram. Unlike counterparts with static configurations, this solution allows an administrator to dynamically add and remove AI models, manage user access, and send broadcasts without editing the source code or restarting the application. This flexibility and scalability are the main competitive advantages, making the bot a powerful and easily manageable tool.

Keywords: Telegram bot, neural networks, API integration, unified gateway, multimodality, Aiogram, Python, Docker.

Павлюк Даниил Артемович

Казанский национальный исследовательский
технический университет им. А.Н. Туполева-КАИ
liketoeatpizza19@gmail.com

Яковлев Игорь Владимирович

Казанский национальный исследовательский
технический университет им. А.Н. Туполева-КАИ
iakovleviv@gmail.com

Аннотация. Статья посвящена теме разработки многофункционального Telegram-бота, выступающего в роли единого шлюза к разнообразным нейросетевым технологиям. Основная цель исследования — решение актуальных проблем раздробленности и географических ограничений, с которыми сталкиваются пользователи AI-сервисов. Проект направлен на создание централизованной платформы, которая объединяет под одним интуитивно понятным интерфейсом как официальные API от ведущих компаний (OpenAI, Google, Anthropic), так и множество сторонних провайдеров, делая передовые технологии доступными для широкой аудитории. Для достижения этой цели была применена модульная асинхронная архитектура на языке Python с фреймворком Aiogram 3 для взаимодействия с Telegram API и базой данных SQLite для хранения данных. Система спроектирована для развертывания в контейнерах Docker, что позволяет ей эффективно функционировать как прокси-шлюз. В результате был реализован бот, обладающий широкими мультимодальными возможностями: он поддерживает текстовые диалоги, генерацию, редактирование и анализ изображений, транскрипцию аудио, а также извлечение текста из документов. Особое внимание уделяется ключевому результату проекта — созданию полнофункционального административного интерфейса внутри Telegram. В отличие от аналогов со статической конфигурацией, данное решение позволяет администратору динамически, без редактирования исходного кода и перезапуска приложения, добавлять и удалять AI-модели, управлять доступом пользователей и выполнять рассылки. Эта гибкость и масштабируемость являются главным конкурентным преимуществом, превращая бота в мощный и легко администрируемый инструмент.

Ключевые слова: telegram-бот, нейронные сети, интеграция API, единый шлюз, мультимодальность, Aiogram, Python, Docker.

Введение

Сегодня нейронные сети используются повсюду, в результате чего появилось множество их разных и узкоспециализированных видов. Крупные компании, такие как Google, OpenAI и Anthropic, а также множество независимых разработчиков, открывают доступ к своим нейросетям через API [4, с. 397]. Каждая такая модель, будь то большая языковая модель, генератор изображений или система синтеза речи, имеет свои особенности, плюсы и минусы [3, с. 288]. Однако этот

бурный рост создал большую практическую проблему — раздробленность. Обычным пользователям и разработчикам приходится разбираться в этой сложной системе: работать с множеством API-ключей и переключаться между разными, часто несовместимыми, сервисами. Ситуация усложняется для пользователей из определенных географических регионов, включая Россию, где доступ ко многим ведущим мировым нейросетевым сервисам ограничен или полностью заблокирован. Именно поэтому создание единой точки доступа является особенно актуальной задачей. Такая система должна скрыть

от пользователя все технические трудности и стать надёжным шлюзом к глобальным технологиям на основе нейросетей.

Целью работы является разработка многофункционального Telegram-бота, который объединяет множество AI-сервисов в одном удобном интерфейсе. Новизна проекта заключается в уникальной возможности использовать одновременно несколько сторонних провайдеров, многие из которых работают на библиотеках, совместимых с OpenAI и Anthropic. При этом бот нативно поддерживает и официальные модели от Gemini, OpenAI и Claude. Добавление новых сторонних провайдеров происходит очень просто через переменные окружения. Кроме того, администратор может добавлять и настраивать модели прямо из интерфейса бота, без вмешательства в код. Это ключевая возможность, так как сторонние провайдеры часто предоставляют доступ к мощным моделям бесплатно, делая технологию доступной для широкого круга пользователей.

В качестве системы управления базами данных была выбрана SQLite, доступ к которой осуществляется через высокопроизводительную асинхронную библиотеку aiosqlite [9; 12, с. 34]. Несмотря на кажущуюся простоту файловой СУБД, структура базы данных была тщательно спроектирована для поддержки всего многообразия функций бота и оптимизирована для асинхронного доступа. Она состоит из нескольких ключевых, нормализованных таблиц (Рис. 1). Таблицы `allowed_users` и `admin_users` хранят целочисленные идентификаторы пользователей Telegram, реализуя простую, но эффективную ролевую модель и систему контроля доступа. Отдельный блок таблиц отвечает за управление AI-

моделями, сгруппированными по модальностям: `models` для текстовых чатов, `whisper_models` для транскрипции голосовых сообщений, а также `image_generation_models` и `image_recognition_models` для генерации и анализа изображений соответственно. Каждая запись в этих таблицах содержит уникальный идентификатор модели, ее человекочитаемое имя для отображения пользователю и строковый идентификатор типа API, к которому она относится. Такая нормализация позволяет администратору гибко и атомарно управлять пулом доступных моделей. Центральной и самой сложной таблицей является `user_contexts`. Она спроектирована как хранилище полного состояния сессии каждого пользователя. Каждая строка, индексированная по `user_id`, содержит всю необходимую информацию для мгновенного восстановления контекста: выбранную пользователем модель, тип API, полную историю сообщений, сериализованную в формат JSON, текущие настройки для генерации изображений, такие как соотношение сторон и флаг улучшения промпта, выбранный голос для синтеза аудиоответов и системную роль, определяющую «личность» бота. Для обеспечения высокой производительности запросов, особенно к таблице `user_contexts`, были созданы B-Tree индексы по всем ключевым полям, по которым происходит фильтрация, в первую очередь по `user_id`, `model` и `api_type`.

Пользовательский интерфейс бота был спроектирован с упором на максимальную простоту, интуитивность и отзывчивость. Он активно использует все доступные инструменты Telegram для создания богатого интерактивного опыта. Основное взаимодействие пользователя с ботом происходит через постоянную клавиатуру, которая предоставляет доступ к главным функциям и избав-

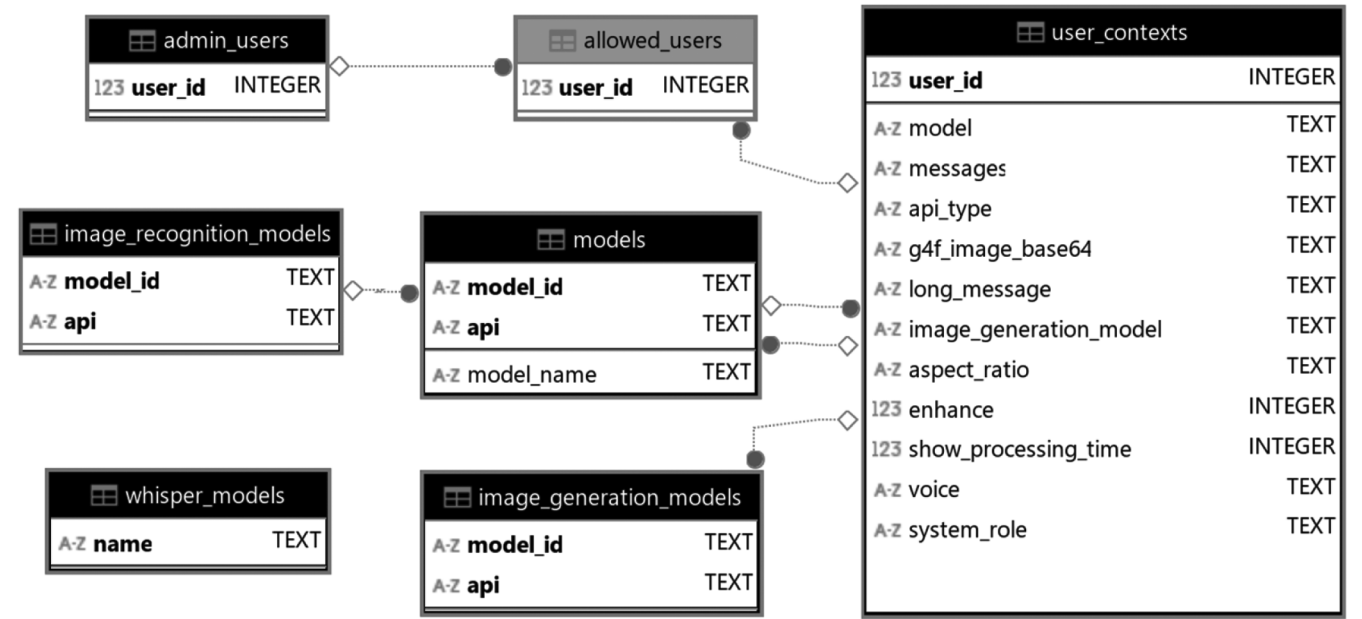


Рис. 1. Диаграмма структуры базы данных

ляет пользователя от необходимости запоминать и вводить текстовые команды.

Кнопка Настройки открывает меню для детальной конфигурации. Здесь пользователь может изменить языковую модель для общения, выбрать модель для генерации изображений, настроить соотношение сторон создаваемых картинок, указать голос для аудиоответов и определить системную роль бота. Кнопка Очистить полностью сбрасывает контекст текущей беседы, удаляя все предыдущие сообщения, но при этом сохраняя все пользовательские настройки. Кнопка Сгенерировать переводит бота в режим ожидания текстового запроса для создания изображения или инструкций для его редактирования. Другие элементы главного меню также предоставляют быстрый доступ к ключевым возможностям. Кнопка Поиск активирует режим поиска в интернете, где следующее сообщение пользователя будет использовано как поисковый запрос для формирования развернутого ответа. Кнопка Аудио переводит бота в режим ожидания аудиофайла для его последующей транскрипции. Кнопка Длинное сообщение включает режим накопления текста, позволяя объединять несколько сообщений в одно большое для последующей обработки. Кроме того, пользователям доступны развлекательные функции через кнопки Мем и Игры.

Особого внимания заслуживает процесс выбора модели, который иллюстрирует принцип контекстнозависимых диалогов. Для более сложных и многовариантных сценариев используются динамически генерируемые инлайн клавиатуры. Например, при нажатии на кнопку выбора модели инициируется продуманный каскадный диалог (Рис. 2). Сначала пользователю предлагается выбрать API, например G4F, Gemini или OpenAI, и только после этого выбора ему показывается отфильтрованный

список моделей, относящихся к данному API. Такой подход предотвращает информационную перегрузку пользователя и делает процесс выбора быстрым и логичным.

Для администратора бота предусмотрен отдельный, защищенный правами доступа, интерфейс, предоставляющий полный контроль над системой без необходимости редактировать код (Рис. 3). Администратор может динамически управлять доступом к боту, добавляя новых пользователей и удаляя существующих. Ключевой возможностью является управление пулом моделей искусственного интеллекта на лету. Администратор может добавлять и удалять модели для чата, генерации и распознавания изображений через интуитивно понятный диалог, и все изменения немедленно отражаются в базе данных. Также реализованы функции для массовых рассылок всем пользователям и отправки точечных сервисных сообщений для информирования и поддержки. Весь интерфейс, от главного меню до многоуровневой административной панели, построен на принципе последовательного раскрытия информации, что делает даже самые сложные операции доступными и понятными.

Функциональные возможности системы

Разработанный бот представляет собой комплексную мультимодальную платформу, функциональность которой выходит далеко за рамки стандартного текстового взаимодействия. Система предоставляет пользователям широкий набор инструментов для работы с различными типами данных, делая его универсальным решением для множества задач.

Одной из ключевых возможностей является глубокая работа с изображениями. Система поддерживает как распознавание, так и генерацию визуального контента.

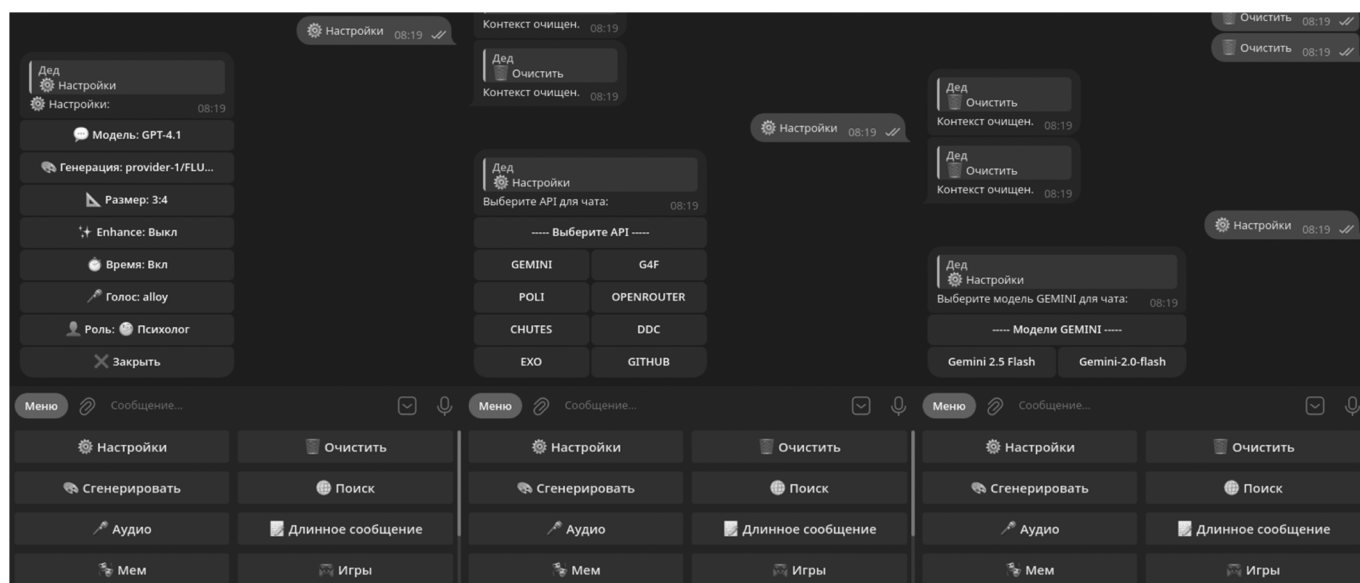


Рис. 2. Визуализация пользовательского пути для смены модели

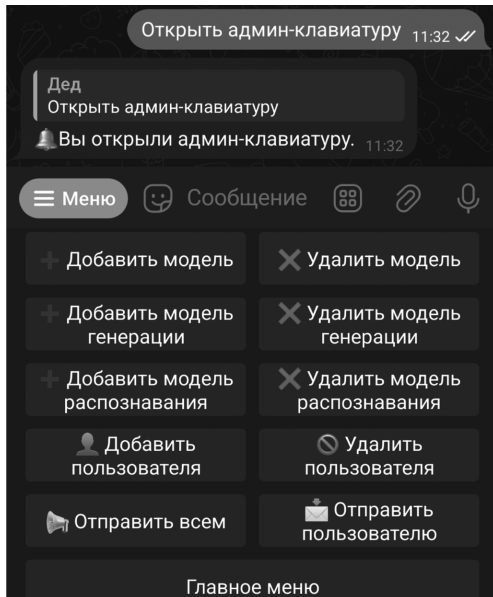


Рис. 3. Меню администратора

Пользователь может отправить изображение вместе с текстовым запросом, после чего бот использует выбранную нейросетевую модель для анализа картинки и ответа на вопрос (Рис. 4). Эта функция реализована с поддержкой нескольких провайдеров, включая Google, OpenAI и Anthropic.

Для генерации изображений пользователь активирует соответствующий режим и вводит текстовое описание. Система обладает двумя важными механизмами улучшения запроса. Во-первых, для достижения наилучшего качества генерации, особенно с моделями, обученными на англоязычных данных, запрос пользователя автоматически переводится на английский язык. Во-вторых, при активации специальной опции Enhance, простой пользовательский запрос передается вспомогательной языковой модели, которая преобразует его в развернутое и детализированное описание, структурированное по профессиональным стандартам, включая указания на ракурс, стиль и технические параметры съемки.

Отдельного внимания заслуживает функция редактирования изображений, реализованная на базе моделей Google Gemini (Рис. 5). Этот мощный инструмент позволяет пользователю отправлять одно или несколько изображений, сопровождая их текстовыми инструкциями по изменению. Система формирует сложный мультимодальный запрос, объединяя визуальные данные и текстовые команды, и направляет его в API Gemini. В результате модель возвращает новое изображение, измененное в точном соответствии с указаниями пользователя, что открывает широкие возможности для креативной работы и быстрой коррекции визуального контента.

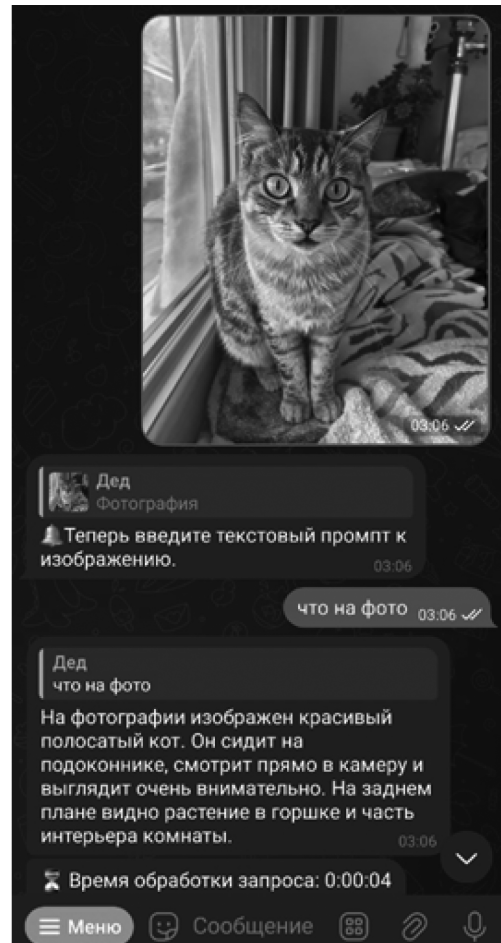


Рис. 4. Анализ фотографий

Система также способна извлекать текстовое содержимое из разнообразных форматов документов. Реализована поддержка файлов PDF, документов Microsoft Word современных версий DOCX и устаревших DOC, а также таблиц Microsoft Excel. Столь широкая совместимость достигается за счет интеграции специализированных библиотек, таких как PyMuPDF для обработки PDF и python-docx для работы с файлами Word [8, с. 225; 1]. Для обеспечения совместимости с устаревшими форматами система прибегает к помощи внешних утилит, например antiword и LibreOffice, которые запускаются в фоновом режиме. Извлеченный из файла текст интегрируется в текущий диалог, позволяя пользователю обсуждать содержимое документа с нейросетью.

Значительное внимание уделено и работе со звуком. Система предлагает две различные аудиофункции (Рис. 6). Первая это транскрибация аудиосообщений. Пользователь может отправить голосовое сообщение или аудиофайл, который будет обработан передовой технологией распознавания речи, такой как Whisper, для получения точной текстовой расшифровки. Второй, более уникальной функцией, является прямое аудиовзаимодействие с моделью openai-audio. В этом режиме голосовое сообщение пользователя не транскрибируется,



Рис. 5. Редактирование фото с помощью модели Gemini

а напрямую передается нейросети, которая в ответ генерирует не текст, а полноценный звуковой файл. Система автоматически выполняет необходимое преобразование формата аудио для его корректного воспроизведения в мессенджере Telegram.

Помимо основных рабочих функций, в боте предусмотрен развлекательный блок для повышения вовлеченности пользователей. Интегрирована функция получения случайных мемов из внешних интернет-источников. Также реализованы классические игры, например Крестики-нолики и Угадай число. Эти игровые механики полностью работают внутри интерфейса бота, используя машину конечных состояний для управления игровым процессом и предоставления интерактивного опыта без необходимости покидать приложение.

Заключение

В ходе выполненной работы был успешно разработан и реализован многофункциональный Telegram-бот, который представляет собой единый шлюз к разнообразным нейросетевым технологиям. Проект достиг поставленной цели, решив ключевую проблему раздробленности AI-сервисов и географических ограничений доступа.

Созданная система эффективно объединяет под одним интерфейсом как официальные модели от ведущих мировых компаний, так и множество сторонних провайдеров, что делает передовые технологии доступными для широкого круга пользователей. Уникальная архитектура, основанная на динамическом управлении моделями через базу данных и административный интерфейс непосредственно в Telegram, обеспечивает беспрецедентный уровень гибкости и масштабируемости, выгодно отличая данное решение от существующих аналогов со статической конфигурацией.

Вопрос масштабируемости также заслуживает внимания. Хотя текущая асинхронная архитектура и использование пула соединений с базой данных SQLite хорошо справляются с нагрузкой, при экспоненциальном росте числа пользователей могут возникнуть узкие места. Переход на более мощную клиент-серверную базу данных, например PostgreSQL, и реализация архитектуры с несколькими рабочими процессами, распределяющими нагрузку, позволили бы обеспечить бесперебойную работу при значительном росте числа пользователей и одновременных запросов.

Еще одним аспектом для будущего развития является отсутствие встроенной системы управления затратами

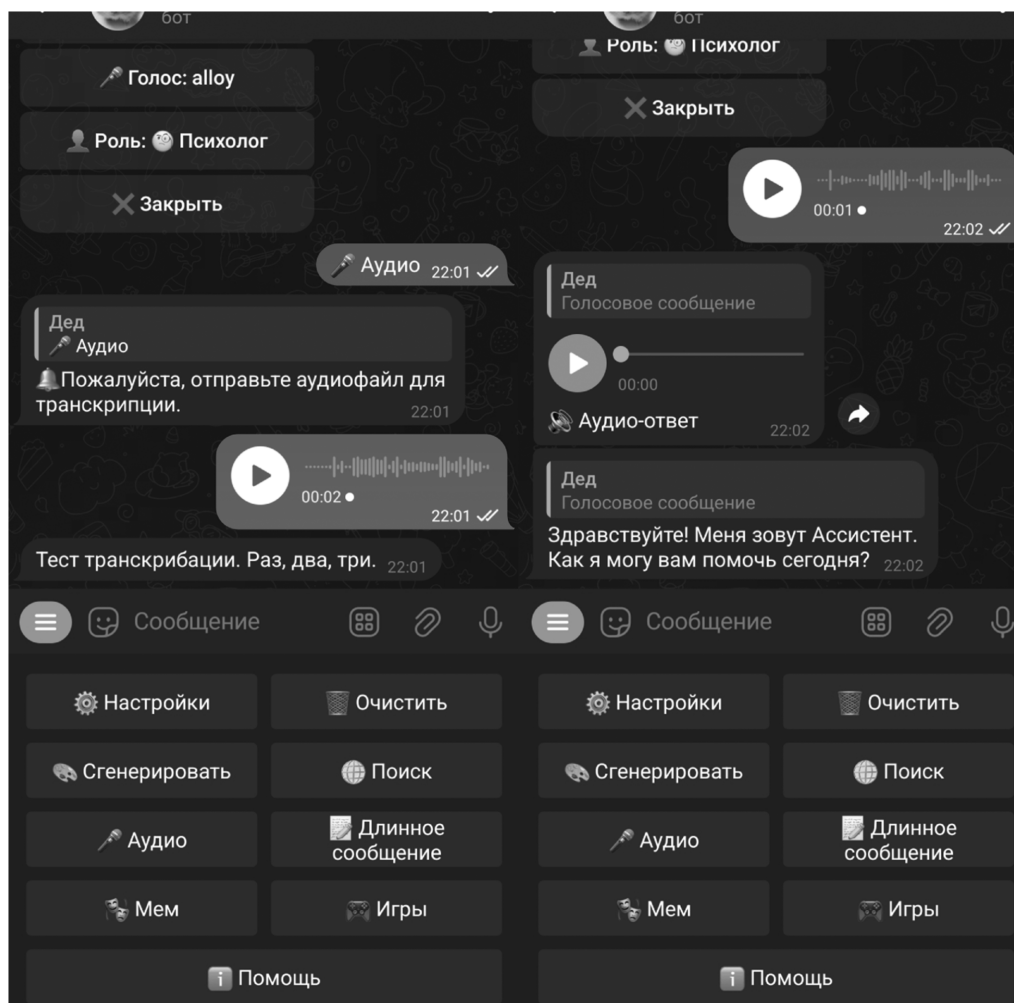


Рис. 6. Работа с аудиофайлами

на использование платных моделей от официальных API. Сейчас эта задача полностью ложится на администратора. Интеграция системы биллинга и персональных лимитов для пользователей могла бы превратить проект в самодостаточную платформу, предлагающую гибкие тарифные планы доступа к премиальным нейросетям. Наконец, повышение общей надежности и улучшение пользовательского опыта представляют собой важные задачи. Разработка более продвинутой системы обработки ошибок, которая бы предоставляла пользователям понятную обратную связь вместо технических сооб-

щений, сделало бы взаимодействие с ботом еще более комфортным и предсказуемым.

Таким образом, созданный бот является не только готовым к использованию решением, но и прочным фундаментом для создания еще более мощной и всеобъемлющей экосистемы искусственного интеллекта, доступной широкому кругу пользователей. Определенные пути для дальнейшего развития открывают большие перспективы для превращения этого проекта в стабильный и коммерчески успешный продукт.

ЛИТЕРАТУРА

1. Валиев А.И., Лысенкова С.А. Применение методов машинного обучения для автоматизации процесса анализа содержания текста // Вестник кибернетики. 2021. №4 (44).
2. Валинурова А.А., Балабанова Н.В., Маценков И.А. Алгоритм разработки Telegram-бота — продуктивного помощника современного бизнеса // Современные наукоемкие технологии. Региональное приложение. 2023. №2 (74). С. 60–67.
3. Жусип М.Н., Жаксыбаев Д.О. Сравнение чат-ботов с использованием трансформеров и нейросетей: исследование применения архитектур gpt и bert // Вестник науки. 2024. Т. 2, №9 (78). С. 287–290.
4. Измайлов М.К. Применение искусственного интеллекта для оптимизации рутинных административных задач: возможности, проблемы и перспективы // Вестник ПГУ. Серия: Экономика. 2024. Т. 19, №4. С. 395–408.
5. Клоков М.Г. Разработка Telegram-бота для управления напоминаниями с использованием Python // Вестник науки. 2025. Т.2, №5 (86) С. 824–831.

6. Насибулин Р.О., Сафонова Е.А., Абдуллин Д.М., Гареева Г.А. Оптимизация инвентарного учета основных средств путем создания Telegram бота // International Journal of Advanced Studies. 2023. Т.13, №1. С. 90–101.
7. Пирматов А.З., Азимов Б.А., Камалов С.С. Искусственный интеллект с использованием Python: технологии и применение // Бюллетень науки и практики. 2023. Т. 9. №11. С. 288–295.
8. Редькина О.Ю., Карпета Т.В., Пономарева Д.С., Вершинина Д.А. Машинное обучение при адаптации учебных текстов: лексический аспект // Вестник ЧелГУ. 2024. №8 (490). С. 220–229.
9. Сударенко Д.А. Разработка базы данных структурно-параметрического описания технологии LTCC с применением Qt SQLite // Труды МАИ. 2021. №121.
10. Токарев Э.А., Галимянов Ф.А. Структурирование обработчиков в telegram-ботах для оптимизации взаимодействия // Вестник науки. 2024. Т.1, №5 (74). С. 523–532.
11. Фаррахов И.Г., Якупов И.М. Автоматизированный инструментальный развертывания облачных сервисов // Мировая наука. 2021. №2 (47). С. 115–118.
12. Химич А.В. К вопросу о реализации баз данных в языке программирования Python // Ученые записки Брянского государственного университета. 2020. №2 (18). С. 33–35.
13. ChatGPT-Telegram-Bot [Электронный ресурс]. — Режим доступа: <https://github.com/yum68686/ChatGPT-Telegram-Bot> (дата обращения: 01.07.25)
14. Grok-ChatGPT: Telegram bot with multiple AI models (GPT-4O, Grok) [Электронный ресурс]. — Режим доступа: <https://github.com/mirvald-space/Grok-ChatGPT> (дата обращения: 01.07.25)
15. TG AI Bot: Telegram bot powered by the Vercel AI SDK [Электронный ресурс]. — Режим доступа: <https://github.com/meetpateltech/tg-ai-bot> (дата обращения: 01.07.25)

© Павлюк Даниил Артемович (liketoeatpizza19@gmail.com); Яковлев Игорь Владимирович (iakovleviv@gmail.com)

Журнал «Современная наука: актуальные проблемы теории и практики»