УДК 004

Разработка Telegram-бота для учета расходов с использованием языковой модели

Лапушкина Елена Павловна
Приамурский государственный университет им. Шолом-Алейхема
Студент

Аннотация
В статье описывается разработка Telegram-бота для учёта расходов по фотографиям кассовых чеков на платформе n8n. Бот принимает изображение чека, передаёт изображение мультимодальной языковой модели OpenAI gpt-5.4-mini и получает структурированный JSON с датой, наименованием магазина, суммой, категорией и оценкой уверенности распознавания. Разработанный рабочий процесс может использоваться как основа для личного и рабочего учёта расходов без написания кода.
Ключевые слова: n8n, Telegram-бот, low-code, распознавание чеков, языковая модель, Google Sheets.

Development of a Telegram bot for expense tracking using a language model.

Lapushkina Elena Pavlovna
Sholom-Aleichem Priamursky State University
Student

Abstract
The article describes the development of a Telegram bot for tracking expenses based on photos of cash receipts using the n8n platform. The bot takes an image of the receipt, passes the image to the OpenAI gpt-5.4-mini multimodal language model, and receives a structured JSON response containing the date, store name, amount, category, and recognition confidence score. The developed workflow can be used as a basis for personal and work expense tracking without writing code.
Keywords: n8n, Telegram bot, low-code, receipt recognition, language model, Google Sheets.

1 Введение

1.1 Актуальность

Учёт личных и рабочих расходов опирается на кассовые чеки. Бумажный чек выцветает и теряется, а ручной перенос даты, суммы и названия магазина в таблицу требует времени и сопровождается опечатками. Мессенджер Telegram уже присутствует на устройстве пользователя, поэтому отправка фотографии чека боту не требует установки отдельного приложения.

Платформа n8n относится к классу low-code решений для автоматизации рабочих процессов: логика собирается из узлов в визуальном редакторе, а

готовые интеграции охватывают мессенджеры, облачные хранилища, таблицы и модели искусственного интеллекта. Мультимодальные языковые модели извлекают из изображения чека структурированные поля без настройки шаблонов под конкретный магазин. Сочетание Telegram, языковой модели и Google Sheets позволяет построить работающую систему учёта без разработки серверной части. Данное обстоятельство обосновывает необходимость создания специализированного рабочего процесса для распознавания и подтверждения чеков.

1.2 Обзор исследований

В. В. Ткаченко и П. Н. Тарлычева определили оптическое распознавание символов как автоматизированный процесс извлечения печатных или рукописных символов из цифровых изображений физических документов и отметили, что ошибки на этапе предобработки приводят к передаче на распознавание неверных данных [1]. Р. И. Шакиров и А. С. Татаурова посвятили исследование разработке бота, решающего проблему доступности расписания в образовательном учреждении; в ключевых словах работы указаны Python и Heroku, облачная платформа для размещения приложений [2]. Б. И. Бийбосунов, С. К. Бийбосунова и Н. Ж. Жолочубеков в работе, описывающей концепцию Telegram-ботов, рассмотрели историю появления ботов и способ создания на языке Python с модулем pyTelegramBotAPI, а также привели пример интерактивного бота с выбором кнопок [3]. Г. А. Гареева и др. исследовали автоматизацию перевода валюты в рубли на основе Telegram, описали создание бота, быстро пересчитывающего любые валюты в рубли, и этапы работы, включающие задание названия бота и получение HTTP API токена [4]. Э. А. Токарев и Ф. А. Галимянов рассмотрели назначение ботов и процесс разработки на Python с библиотекой pyTelegramBotAPI: получение токена через BotFather, создание бота, обработчики сообщений и цикл ожидания сообщений [5].

1.3 Цель исследования

Целью исследования является разработка на платформе n8n Telegram-бота, который принимает фото или документ с чеком, распознаёт дату, магазин, сумму и категорию, сохраняет файл чека и формирует черновик записи в таблице Google Sheets.

2 Материалы и методы

Для реализации использована облачная версия платформы n8n. Рабочий процесс представляет собой цепочку узлов: каждый узел выполняет одно действие и передаёт результат следующему. Распознавание выполняется мультимодальной моделью OpenAI gpt-5.4-mini.

3 Результаты и дискуссии

На начальном этапе необходимо зарегистрироваться на платформе n8n https://n8n.io/. Связь платформы с Telegram устанавливается через токен бота.

Для получения токена в Telegram открывается бот @BotFather, выполняется команда /newbot, задаются название и имя бота, после чего BotFather присылает токен доступа к HTTP API. В n8n в разделе Credentials создаётся credential типа Telegram API, в поле Access Token вставляется полученный токен. Узел Telegram Trigger, принимающий события message и callback_query, подключается к этому credential, а после публикации процесса n8n автоматически регистрирует webhook в Telegram, и новые сообщения боту поступают в рабочий процесс. Далее создаётся узел Route Update, который разделяет события по правилам на выходы Receipt file и Button pressed. Ветка чека включает узлы Recognize receipt и Send draft for review. Статус Published означает, что процесс опубликован и обрабатывает сообщения постоянно (рис.1).

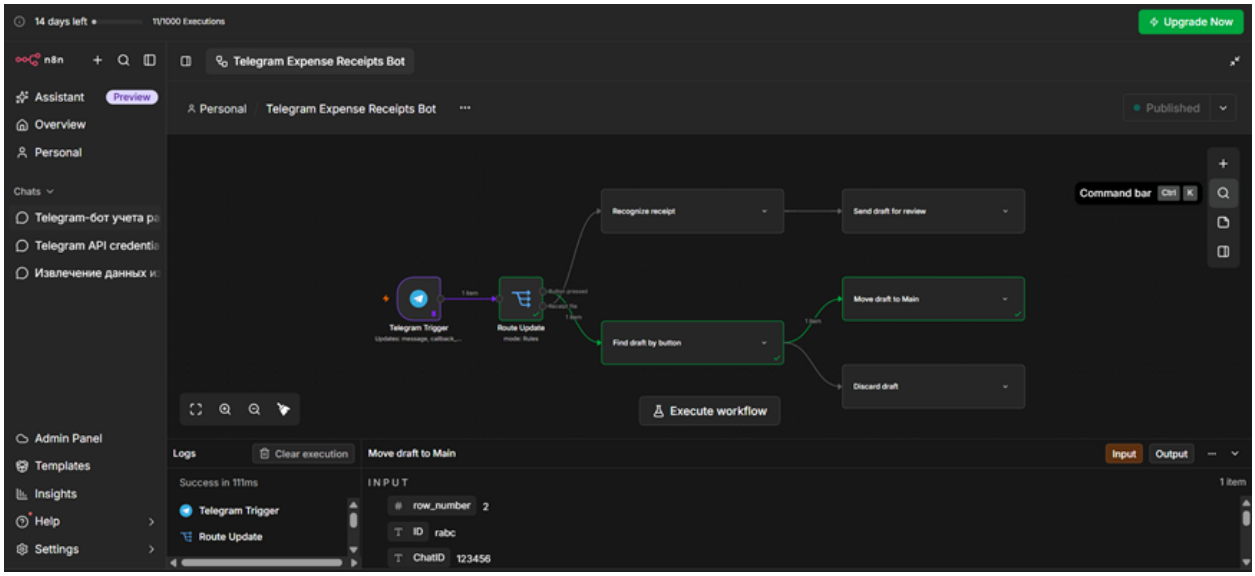


Рисунок 1. Рабочий процесс Telegram Expense Receipts Bot в редакторе n8n

Назначение отдельных узлов процесса приведено в таблице 1.

Таблица 1. Узлы рабочего процесса и назначение узлов

Узел	**Назначение**
Telegram Trigger	Приём входящих сообщений и нажатий кнопок через credential Telegram API с токеном бота, игнорирование чужих чатов
Route Update	Определение типа события: чек или кнопка
Download Receipt	Скачивание файла чека из Telegram
Analyze Receipt	Передача изображения модели gpt-5.4-mini, получение JSON
Build Record	Разбор ответа модели, формирование записи с уникальным ID
Upload to Drive	Загрузка файла в Google Drive, получение ссылки
Save to Temp	Добавление строки в лист Temp
Send Preview	Отправка данных чека и кнопок save:<ID>, del:<ID>

Узел	Назначение
Find Draft	Поиск строки в листе Temp по ID
Save to Main	Копирование строки в лист Main
Remove Draft After Save	Удаление строки из листа Temp после записи
Confirm Saved	Подтверждение записи
Remove Draft	Удаление строки из листа Temp по кнопке «Удалить»
Confirm Deleted	Подтверждение удаления
Acknowledge Button	Немедленный ответ на нажатие кнопки

Ветка чека выполняется в следующем порядке: скачивание файла, анализ изображения моделью, разбор ответа, загрузка файла в Google Drive, добавление строки в лист Temp и отправка предварительного просмотра. Результат работы ветки показан на (рис.2): бот вернул дату 2019-11-21, магазин «ТСД Солнечный город», сумму 1500, категорию Other и уверенность 0.72, а также ссылку на сохранённый чек и две кнопки.

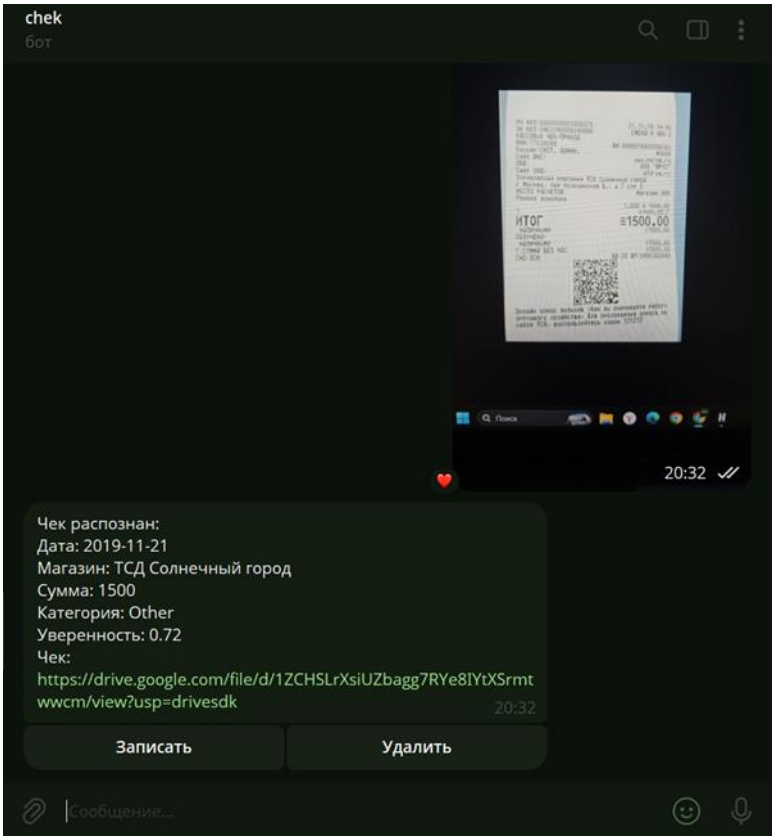


Рисунок 2. Сообщение бота с распознанными данными чека и кнопками «Записать» и «Удалить»

В реальном запуске на другом чеке модель корректно прочитала магазин «ООО Автокафе», сумму 785 и категорию Food. Значение уверенности ниже единицы служит сигналом для внимательной проверки данных до нажатия кнопки «Записать».

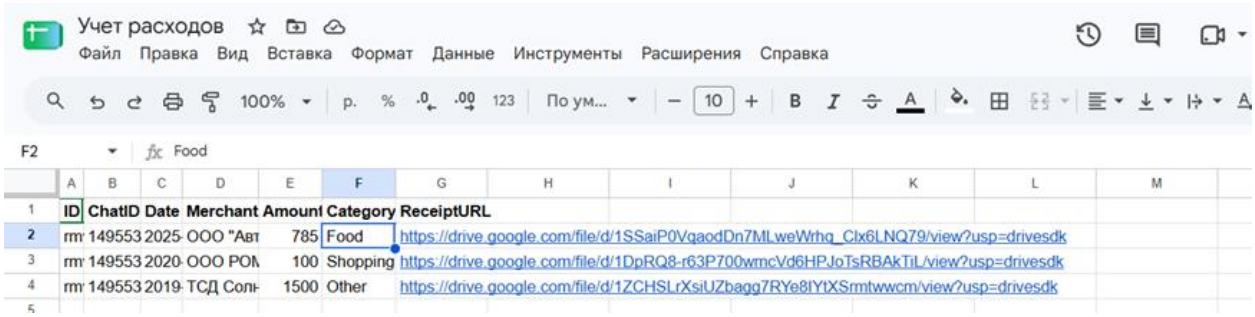


Рисунок 3. Лист таблицы «Учет расходов» с распознанными чеками

Обработка кнопки «Записать» начинается с поиска строки в листе Temp по значению ID из данных кнопки. Найденная строка передаётся на вход узла Move draft to Main, входные данные которого в тестовом запуске показаны на (рис.4): номер строки row_number, ID, ChatID, Date, Merchant, Amount, Category и ReceiptURL. Запуск завершился успешно. Узел копирует запись в лист Main, затем строка удаляется из листа Temp, после чего пользователь получает подтверждение. Ветка «Удалить» удаляет строку из листа Temp и отправляет подтверждение удаления. В обеих ветках узел Acknowledge Button сразу отвечает на нажатие, чтобы кнопка не оставалась в состоянии ожидания.

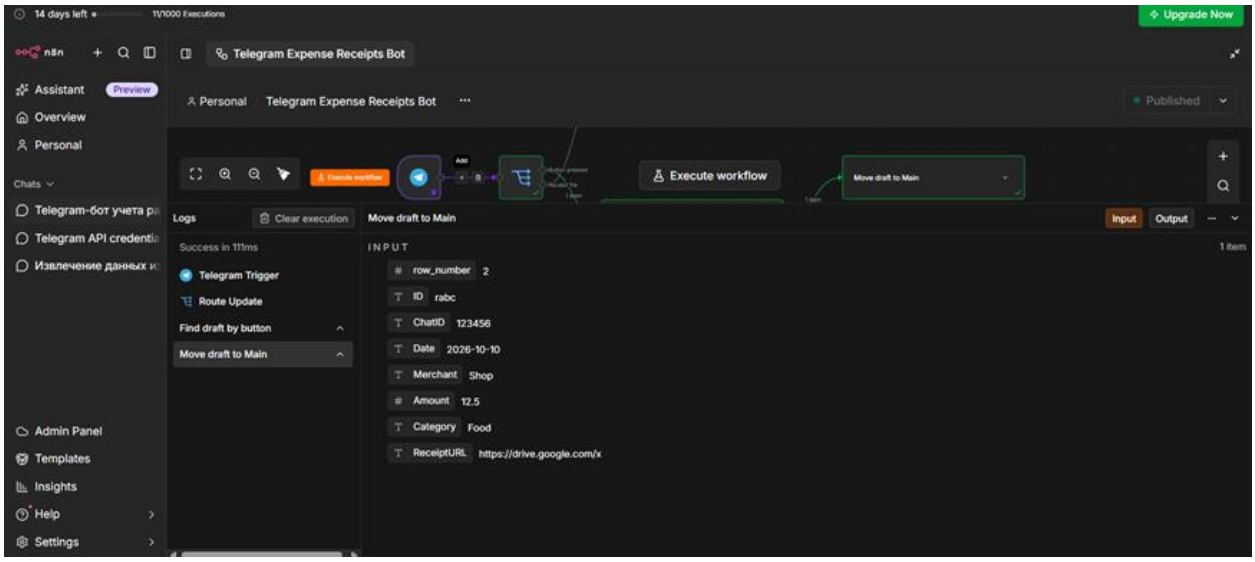


Рисунок 4. Входные данные узла Move draft to Main в журнале выполнения

При настройке доступа к таблице выявлена типичная ошибка: ответ Google Sheets с кодом 403 при записи в лист Temp означает, что у аккаунта, подключённого в credential, нет прав на редактирование. Устранение состоит в выдаче роли «Редактор» на таблицу либо в повторном подключении credential Google Sheets. До устранения ошибки сообщение с кнопками не отправляется, так как отправка расположена после записи черновика.

Тестовые запуски загрузили в Google Drive несколько дубликатов чека. Для исключения повторной загрузки при отладке целесообразно проверять наличие файла до загрузки либо удалять тестовые файлы после завершения проверки.

4 Выводы

В ходе выполненной работы разработан Telegram-бот для учёта расходов по чекам на платформе n8n. Бот принимает фото или документ с чеком, извлекает дату, магазин, сумму и категорию с помощью языковой модели, сохраняет файл в Google Drive и формирует черновик записи в таблице Google Sheets. Двухэтапная схема с листами Temp и Main и подтверждением кнопкой снижает риск попадания ошибочно распознанных данных в основной учёт. Идентификатор записи, вшитый в данные кнопок, связывает сообщение с конкретной строкой черновика, а проверка идентификатора чата ограничивает доступ владельцем. Журнал выполнения n8n позволил локализовать ошибки на уровне отдельных узлов, включая отказ записи с кодом 403. Дальнейшее развитие включает автоматический запрос уточнения при низкой уверенности распознавания, защиту от повторной загрузки одного чека и расширение списка категорий.

Библиографический список

1. Ткаченко В. В., Тарлычева П. Н. Информационная система автоматизированного распознавания реквизитов экономических документов на основе OCR- алгоритмов //Вестник Академии знаний. 2023. №. 1 (54). С. 239-244. URL: https://cyberleninka.ru/article/n/informatsionnaya-sistema-avtomatizirovannogo-raspoznavaniya-rekvizitov-ekonomicheskih-dokumentov-na-osnove-ocr-algoritmov
2. Шакиров Р. И., Татаурова А. С. Автоматизация текущего расписания через Telegram-бот //Символ науки. 2020. №. 1. С. 37-39. URL: https://cyberleninka.ru/article/n/avtomatizatsiya-uchebnogo-raspisaniya-cherez-telegram-bot
3. Бийбосунов Б. И., Бийбосунова С. К., Жолочубеков Н. Ж. Описание концепции Telegram ботов и их разработка //Colloquium-journal. Голопристанський міськрайонний центр зайнятості, 2020. №. 7 (59). С. 3-6. URL: https://cyberleninka.ru/article/n/opisanie-kontseptsii-telegram-botov-i-ih-razrabotka
4. Гареева Г. А. и др. Автоматизация процесса перевода валюты в рубли на основе Telegram //International Journal of Advanced Studies. 2023. Т. 13. №. 4. С. 128-141. URL: https://cyberleninka.ru/article/n/avtomatizatsiya-protsessa-perevoda-valyuty-v-rubli-na-osnove-telegram
5. Токарев Э. А., Галимянов Ф. А. Структурирование обработчиков в Telegram-ботах для оптимизации взаимодействия //Вестник науки. 2024. Т. 1. №. 5 (74). С. 523-532. URL: https://cyberleninka.ru/article/n/strukturirovanie-obrabotchikov-v-telegram-botah-dlya-optimizatsii-vzaimodeystviya