

Студ. Ан. А. Бестемяникова
Науч. рук. ст. преп. Я.А. Игнаткова
(кафедра информатики и веб-дизайна, БГТУ)

СПОСОБЫ СОЗДАНИЯ ЧАТ-БОТОВ НА PYTHON С ПОМОЩЬЮ БИБЛИОТЕК CHATTERBOT ИЛИ API TELEGRAM

Чат-боты стали неотъемлемым инструментом в сферах бизнеса, клиентской поддержки и автоматизации процессов. Согласно исследованию Grand View Research, мировой рынок чат-ботов в 2024 году оценивается в \$7,76 млрд при ежегодном росте 23,3%. В Python выделяют два основных подхода к разработке чат-ботов: использование специализированных библиотек (например, ChatterBot) и интеграция с мессенджерами через API (например, Telegram Bot API).

Библиотека ChatterBot представляет собой ML-решение для генерации контекстно-зависимых ответов на основе предварительно обученных корпусов данных, что продемонстрировано в листинге:

```
from chatterbot import ChatBot
from chatterbot.trainers import ChatterBotCorpusTrainer
bot = ChatBot('ExampleBot')
trainer = ChatterBotCorpusTrainer(bot)
trainer.train("chatterbot.corpus.english")
response = bot.get_response("Привет")
print(response)
```

Точность ответов составляет 60-70% в зависимости от обучения. Библиотека поддерживает более 15 языков, включая русский, и среднее время обработки запросов до 0.5 секунды.

Telegram предоставляет полнофункциональное API для создания интегрированных ботов. На данный момент количество активных телеграм-ботов превышает 500 млн. Скорость ответа меньше одной секунды, максимальная нагрузка до 30 сообщений в секунду на одного бота.

Благодаря этим характеристикам Telegram является одной из самых популярных платформ для разработки чат-ботов, предлагая возможности для автоматизации множества процессов. Боты могут использоваться для широкого спектра задач – от обработки заказов до предоставления персонализированного контента.

Выбор инструмента для реализации чат-бота зависит от задач проекта. ChatterBot оптимален для простых кейсов, а Telegram Bot API подходит для коммерческих решений с высокой нагрузкой. Оба подхода имеют свои ниши применения и продолжают развиваться.