

Лабораторная работа № 4.2

Умный Бот ВКонтакте Raspberry Pi

Теоретическая часть:

Боты — это специальные автоматизированные программы, действующие по запланированному сценарию, командам. На многих сайтах, в социальных сетях чат-боты поддерживают беседу с пользователями, рассказывают о скидках и акциях, и просто создают видимость «людей».

В этой статье я расскажу, что такое боты, для чего они нужны, каким командам они подчиняются. Также подробно опишу 10 преимуществ от настройки бот-программ и как их можно создать, не обладая техническими знаниями в своем аккаунте во ВКонтакте. Что такое боты во ВКонтакте

Боты — это пользовательские «фейковые» страницы, которые предназначены для общения, рассылки спама, сообщений, накрутки подписок и просто в рекламных целях. Зачем нужны чат-боты

Боты во ВКонтакте, как и в других социальных сетях нужны для автоматизации подписок, лайков и вступлений или отписок в сообщества. Правильно запрограммированный бот избавит вас от часовых сидений за компьютером или у телефона, от рутины.

Допустим, у вас есть довольно раскрученный аккаунт во ВКонтакте, вы занимаетесь продажей некоего товара через него. С чем вы сталкиваетесь ежедневно? Именно. С одними и теми же вопросами потенциальных покупателей. Что может сделать купленный или созданный вами бот:

- отвечать на вопросы;
- предоставлять информацию о цене, размерах, ассортименте;
- рассылать сообщения о действующих акциях или скидках;
- автоматизировать прием и обработку заказов;
- облегчить поиск нужного товара или услуги по категориям.

Иными словами, боты помогают в продвижении не только самого аккаунта, но и с продажами, и коммуникацией среди покупателей или гостей страницы.

Виды ботов

По степени сложности программы боты бывают:

Примитивные — простенькая утилита, которая способна переписываться с пользователями по односложному сценарию, рассылать спам и сообщения. Если в базе ответа на заданный вопрос пользователя нет, то чат-бот пришлет сообщения «Я вас не понимаю».

Продвинутые — это боты с искусственным интеллектом. Такие программы способны просчитывать несколько вариантов ответов на вопросы, угадывать настроение написавшего человека, и предоставить ту информацию, которая нужна пользователю.

Как работает бот ВКонтакте?

Принцип работы любого бота — получить сообщение от пользователя, сформировать ответ и отправить его обратно.

К сожалению, используя API ВКонтакте, сделать это в рамках одного метода невозможно: отправка и получение сообщений никак не связаны между собой.

На текущий момент получение сообщений проще всего реализовывать на основе Callback API. Работает оно следующим образом: вы подписываетесь на определенные типы событий, а когда они происходят, ВКонтакте отправляет HTTP-запрос с данными о произошедшем событии на заданный вами URL.

Независимо от того, собираетесь ли вы взаимодействовать с этим событием, в ответ на запрос нужно вывести HTTP-статус «ок», его код равен 200. Если этого не произойдет, сервер ВКонтакте будет считать, что уведомление не получено и продолжит отправлять его с некоторой периодичностью, а потом временно прекратит отправку уведомлений.

Ключи доступа 2 **Callback API**

Callback API используется для получения уведомлений о событиях в Вашем сообществе. [Подробнее »](#)

Настройки Типы событий Запросы

Адрес Вашего сервера ?

Обновить

Для получения уведомлений нужно подтвердить адрес сервера. На него будет отправлен **JSON-запрос**:

```
{"type": "confirmation"}
```

Строка, которую должен вернуть сервер:

```
c48a19cc
```

Если вы не знакомы с работой Callback API, рекомендуем изучить документацию перед продолжением чтения статьи.

Ответ пользователю мы будем отправлять методом `messages.send` с ключом доступа сообщества. Получить его можно на странице настроек сообщества, рядом с вкладкой Callback API.

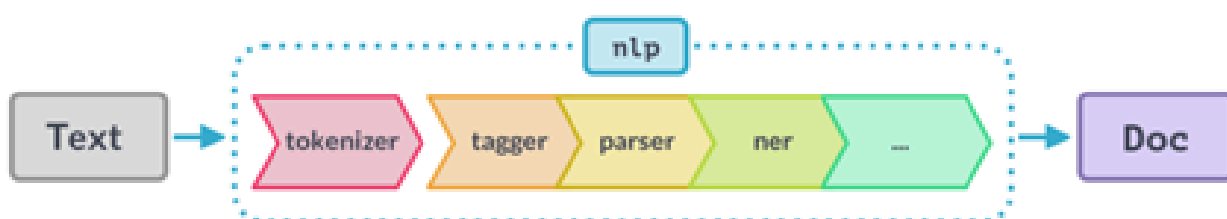
С токенами сообщества и связаны «подводные камни». Во-первых, один ключ имеет ограничение: 3 запроса в секунду. Решается это выпуском новых ключей — их число неограниченно, а значит можно забыть о лимитах. Во-вторых, новый ключ сообщества может оказаться недействительным, тогда любой запрос приведет к ошибке «Access denied». Защититься от этого можно лишь проверяя каждый новый ключ в рабочем приложении.

NLP модель

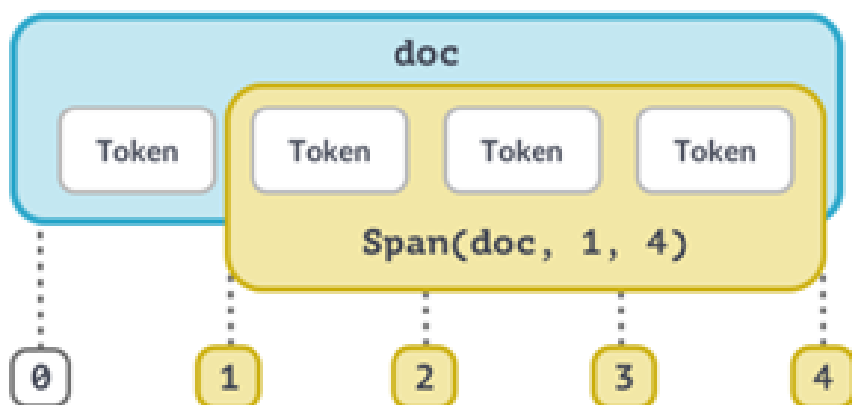
Обработка естественного языка сейчас используется повсеместно: стремительно развиваются голосовые интерфейсы и чат-боты, разрабатываются модели для обработки больших текстовых данных, продолжает развиваться машинный перевод.

В этой статье мы рассмотрим относительно новую библиотеку SpaCy, которая на данный момент является одним из самых популярных и удобных решений при обработке текста в Python. Её функционал позволяет решать очень широкий спектр задач: от определения частей речи и выделения именованных сущностей до создания собственных моделей для анализа.

Для начала давайте наглядно рассмотрим, как происходит обработка данных в SpaCy. Загруженный для обработки текст последовательно проходит через различные компоненты обработки и сохраняется как экземпляр объекта Doc:

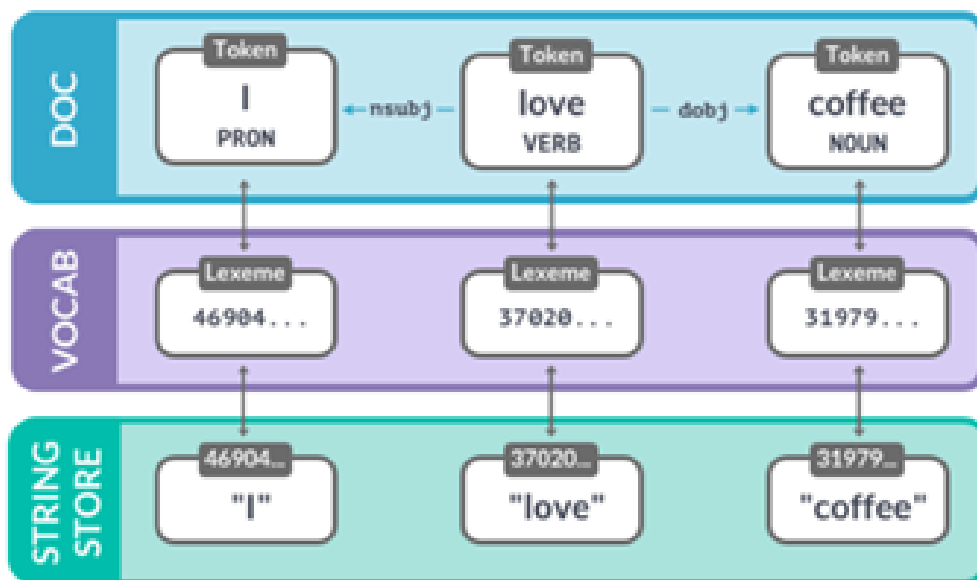


Doc является центральной структурой данных в SpaCy, именно в нём хранятся последовательности слов или, как их ещё называют, токенов. Внутри объекта Doc можно выделить два других типа объекта: Token и Span. Token представляет собой ссылку на отдельные слова документа, а Span – ссылку на последовательность из нескольких слов (их можно создавать самостоятельно):



Ещё одной важной структурой данных является объект Vocab, который хранит набор справочных таблиц, общий для всех документов. Это позволяет экономить память и обеспечивать единый источник информации для всех обрабатываемых документов.

Токены документов связаны с объектом Vocab через хеш, используя который можно получить начальные формы слов или другие лексические атрибуты токенов:



Теперь мы знаем, как устроено хранение и обработка данных в библиотеке SpaCy. А как воспользоваться возможностями, которые она предоставляет? Давайте последовательно рассмотрим операции, с помощью которых можно обработать текст.

Статистические модели

В то время как некоторые функции spaCy работают независимо, другие требуют загрузки статистических моделей, которые позволяют spaCy **предсказывать** лингвистические аннотации, например, является ли слово глаголом или существительным. В настоящее время spaCy предлагает статистические модели для множества языков, которые можно установить как отдельные модули Python. Модели могут различаться по размеру, скорости, использованию памяти, точности и содержащимся в них данным. Выбор модели всегда зависит от вашего варианта использования spaCy и текстов, с которыми вы работаете. Как универсальный вариант выбора по умолчанию используется небольшие модели. Модели обычно включают в себя следующие компоненты:

- **Двоичные веса (Binary weights)** для теггера (tagger) части речи, анализатора зависимостей и распознавателя именованных сущностей (NER) для прогнозирования этих аннотаций в контексте.
- **Лексические записи (Lexical entries)** в словаре, то есть слова и их независимые от контекста атрибуты, такие как форма или написание.
- **Файлы данных**, такие как правила лемматизации и таблицы поиска.
- **Векторы слов (Word vectors)**, то есть многомерные смысловые представления слов, которые позволяют определить, насколько они похожи друг на друга.
- **Параметры конфигурации**, такие как язык и параметры конвейера обработки (pipeline), для перевода spaCy в правильное состояние при загрузке модели.

Практическая часть:

Создание бота для сообщества ВКонтакте включает в себя несколько шагов, включающих регистрацию сообщества, настройку доступов и обработку сообщений.

1. Создание сообщества ВКонтакте

- Если у вас еще нет сообщества, то следует создать его. Это можно сделать перейдя по ссылке <https://vk.com/groups?tab=admin> и нажав кнопку "Создать сообщество".

2. Получение ключа доступа

- Перейдите в меню "Управление сообществом" и выберите пункт "Работа с API".
- Создайте ключ доступа, выбрав необходимые права доступа.
- Токен доступа должен иметь права на отправку сообщений и управление сообществом.

Инструкция:

1. Установите библиотеки vk_api и spacy на Raspberry Pi, если у вас их нет:
`pip install vk_api spacy`.
2. Установите NLP модель spaCy на Raspberry Pi:
`python -m spacy download en_core_web_sm`.
3. Получите токен доступа для своего бота через настройки сообщества в ВКонтате и подставьте его в переменную token в коде.
4. Запустите данный скрипт на Raspberry Pi. Он будет подключаться к ВКонтате, ожидать новые сообщения и отвечать на них, используя NLP модель spaCy.

Помните, что для работы этого бота на Raspberry Pi необходимо подключение к интернету. Также удостоверьтесь, что Raspberry Pi имеет установленный Python и подходящие библиотеки.

Этот пример представляет собой основу для создания текстового бота с использованием NLP модели на Raspberry Pi. Для улучшения обработки естественного языка и создания более информативных ответов бот может быть расширен с добавлением более сложных моделей NLP или интеграцией с другими сервисами обработки текста.

```
import vk_api
from vk_api.longpoll import VkLongPoll, VkEventType
import spacy
# Загрузка NLP модели spaCy
nlp = spacy.load("en_core_web_sm")

def write_msg(user_id, message):
    vk_session.method('messages.send', {'user_id': user_id,
    'message': message, 'random_id': 0})

# Авторизация в ВКонтате
token = ''
vk_session = vk_api.VkApi(token=token)
#vk = vk_session.get_api()
longpoll = VkLongPoll(vk_session)
# Обработка сообщений
for event in longpoll.listen():
    if event.type == VkEventType.MESSAGE_NEW and event.to_me:
        text = event.text
        print(text)
        # Применение NLP модели для обработки текста
        doc = nlp(text)
        # Получение ответа на основе анализа текста
        response = "Привет! Я бот. Вы сказали: " + text
```

```
write_msg(event.user_id, response)
```

Усложняем задачу:

- 1) Создайте подключение к базе данных postgres. Создайте таблицу для записи туда текста сообщения и время отправления
- 2) Подключите открытую API для получения данных о погоде. По команде «погода». Бот должен выдавать данные о погоде в соответствующем режиме.
- 3) Подключите парсер новостной ленты, которая будет выдавать подходящий пост по запросу Новость «Тема»

- 4) Создайте систему, которая будет искать наиболее подходящего для тебя друга среди подписчиков группы. (Просто рандом не принимается, должна присутствовать модель анализа данных со страницы пользователя)