

Министерство образования и науки Российской Федерации  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ Н.Г.ЧЕРНЫШЕВСКОГО»

Кафедра дискретной математики и  
информационных технологий

**Разработка информационно-справочной системы по ремонту и  
обслуживанию ТС на основе методов машинного обучения**

АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ

студента 4 курса 421 группы  
направления 09.03.01 – Информатика и вычислительная техника  
факультета КНиИТ  
Алексаненковой Екатерины Павловны

Научный руководитель  
к.ф.-м.н., доцент

\_\_\_\_\_

И.Д. Сагаева

Заведующий кафедрой  
к.ф.-м.н., доцент

\_\_\_\_\_

Л.Б. Тяпаев

Саратов 2018

**Введение.** Двадцать первый век по праву считается веком информационных технологий. Чтобы быстро найти какую-либо нужную информацию достаточно иметь при себе смартфон. Разработчики разных операционных систем предоставляют огромный выбор приложений, которые нацелены на помощь пользователю. Можно сказать, что современный смартфон - это портативная копия компьютера, им легко пользоваться как в повседневной жизни, так и в заграничных поездках.

Актуальной задачей является предоставление сервиса автолюбителям, который позволит избавить их от длительного поиска информации на просторах интернет-паутины.

В сети много информации по ремонту автомобилей, но редко получается найти необходимую инструкцию или цену на запасную часть быстро, так как разнообразие комплектаций у каждой марки авто велико. Существует огромное количество форумов для автолюбителей, где можно задать интересующий вопрос, но, к сожалению, эти форумы не гарантируют правильность ответа.

В настоящее время нет аналогов данной информационно-справочной системы, работающей в виде чат-бота.

Целью данной работы является создание программной информационно-справочной системы по ремонту и обслуживанию ТС, которая будет помогать автолюбителю в поиске автозапчастей, информировать по вопросам ремонта и обслуживания. Пользование данной информационно-справочной системой будет удобным, эффективным и быстрым, благодаря интеграции с общедоступной и популярной социальной сетью «ВКонтакте».

Объектом исследования выступает процесс организации информационно-справочной системы по ремонту и обслуживанию ТС.

Предметом исследования в данной ВКР является технология для построения системы информационно-справочной системы по ремонту и обслуживанию ТС.

В ходе дипломной работы описана разработка виртуального собеседника, работающего с нейронной сетью.

Информационно-справочная система ориентирована, в частности, на автолюбителей. Для этого были выделены и решены следующие задачи:

- подключение чат-бота к существующей странице «ВКонтакте»;

- подключение Long Poll к серверу для получения обновлений, связанных с сообщениями;
- реализация автоматической отправки и приема сообщений;
- применение алгоритма нечеткого поиска для распознавания сообщений с опечаткой;
- реализация структуры диалога с переходами между состояниями;
- создание базы данных для хранения обращений пользователя и их состояний в диалоге;
- импорт данных с сайтов доставки автозапчастей;
- реализация отправки видеофайлов и изображений в качестве ответов;
- применение глубинного обучения для реализации классификации вопроса.

В главе 1 «Виртуальный собеседник» даны основные определения, касающиеся программы-собеседника.

В главе 2 «Анализ предметной области» определены основные проблемы автолюбителей.

В главе 3 «Описание информационно-справочной системы» описывается схема работы информационно-справочной системы.

В главе 4 «Подключение чат-бота» содержится описание работы VK API, рассмотрены преимущества платформы «ВКонтакте», описаны настройка получения обновлений и метод VK API messages.send.

В главе 5 «Хранение состояний» содержится описание структуры чат-бота, состояний и переходов между ними.

В главе 6 «Нечёткий поиск. Расстояние Дамерау-Лавенштейна» дано определение нечёткого поиска и описан алгоритм нечёткого поиска Дамерау-Лавенштейна. Также представлено решение задачи нечёткого поиска для распознавания сообщений с опечаткой.

В главе 7 «Импорт данных из веб-страниц» дано определение модуля BeautifulSoup и продемонстрирован импорт информации с веб-страниц.

В главе 8 «Машинное обучение» даны основные определения, связанные с машинным обучением, поставлена задача классификации вопросов

пользователя, описана технология TensorFlow. Также описана архитектура нейронной сети, её обучение и запуск графа и получение результата.

В главе 9 «Формирование ответа» описаны функции состояний чат-бота и основные методы VK API, использованные в данной работе.

В главе 10 «Запуск проекта» продемонстрированы команды для установки необходимых модулей.

**1**            **Виртуальный собеседник.** Виртуальный собеседник — это компьютерная программа, которая создана для имитации речевого поведения человека при общении с одним или несколькими пользователями. По отношению к виртуальным собеседникам употребляется также название программа-собеседник или чат-бот.

Среди программ-собеседников есть как искусственные интеллекты, так и эмуляторы. Программе не обязательно быть слишком сложной, чтобы понравиться обратившемуся пользователю, - и наоборот, можно затратить много сил и времени, разработать сложный алгоритм, а программа не вызовет у пользователей удовлетворения, и пользоваться программой они не будут.

При разработке таких программ нужно не только уметь программировать, но и немного владеть знаниями психологии человека и иметь основные понятия рассматриваемой предметной области, на которую направлен виртуальный собеседник.

Современные чат-боты не опираются исключительно на текст, а ещё и умеют показывать всевозможные полезные изображения, видеофайлы, ссылки, формы, создавая атмосферу использования приложения [1, 2].

**2**            **Подключение чат-бота.** Каждый автовладелец рано или поздно сталкивается с поломками в автомобиле. Более того, внимательные и ответственные проходят плановое ТО примерно раз в полгода. Когда возникает вопрос по поводу неисправностей или сроков планового технического обслуживания, то владелец ищет ответ с помощью запросов в поисковой системе интернета. Чтобы найти нужный ответ приходится отсеивать сотни страниц результатов поиска с информацией. Но еще и не факт, что данную поломку можно починить обычному человеку, без специальных навыков и соответствующего образования.

Каждый раз при обращении к поисковой системе пользователь вводит марку и модель своего авто, а при поиске автомобильных деталей в каждом интернет-магазине нужно подтверждать город получателя. Именно для решения этих проблем, в рамках данной ВКР, должна быть разработана программная система, которая позволит осуществлять хранение и обработку данных о марке и модели автомобиля, а так же города доставки онлайн-покупок.

**3 Описание информационно-справочной системы.** Для реализации решения задачи классификации было решено выбрать социальную сеть «ВКонтакте», которая предоставляет платформу для создания чат-ботов.

- Сообщество — это точка входа для пользователей, которые будут общаться с нашим ботом. Стоит позаботиться о том, чтобы оно выглядело привлекательно и содержало описание нашего бота, иначе пользователь не будет понимать как им пользоваться [4].
- VK API — это посредник между разработчиком приложений и какой-либо средой, с которой это приложение должно взаимодействовать.
- Python Script - скрипт чат-бота, в нем выполняются следующие функции:
  - обработка входящих сообщений;
  - переходы между состояниями;
  - обращение к базе данных;
  - выполнение поиска деталей;

отправка сообщения, видео или изображения в качестве результата.

- Deep Learning - глубокое обучение. С помощью технологии TensorFlow определяется класс, к которому принадлежит сообщение от пользователя.

## **4 Подключение чат-бота.**

**4.1 VK API.** Для подключения чат-бота можно использовать любое своё сообщество «ВКонтакте» — группу, встречу или публичную страницу. В данном случае было создано сообщество «CarService».

API — это посредник между разработчиком приложений и какой-либо средой, с которой это приложение должно взаимодействовать. API упрощает создание кода, поскольку предоставляет набор готовых классов, функций или структур для работы с имеющимися данными [5].

### **4.1.1 Преимущества платформы.**

- Социальный граф.

«ВКонтакте» может работать с социальным графом, анализируя связи и интересы пользователя. Можно отслеживать количество посещений и

количество уникальных пользователей. Например, для создания рекомендательной системы или сервиса персональной инфографики.

- Мультимедиа.

Огромная библиотека медиаконтента, который можно использовать для создания подборок фотографий, видео или аудиозаписей.

- Кроссплатформенность.

Сообщения сообществ работают в полной и мобильной версиях, а также во всех официальных клиентах. Это значит, что пользователь может взаимодействовать с ботом на любой платформе.

- Документация.

У «ВКонтакте» есть техническая документация, с простыми примерами реализаций, разбором запросов и описанием собственных технологий «ВКонтакте».

**4.2 Сообщения сообщества.** Сообщения сообществ — сервис для прямого диалога между пользователем и сообществом «ВКонтакте».

Сообщения сообществ работают в полной и мобильной версиях «ВКонтакте», а также во всех официальных приложениях.

Для активации отправки и приема сообщений необходимо включить сообщения («Управление сообществом» -> «Сообщения»), когда бот будет готов к использованию, чтобы ему можно было написать.

**4.3 Ключ доступа.** Ключ доступа потребуется нам для получения обновлений (в Long Poll), а также для работы с API.

Можно создать несколько ключей с разными правами доступа. Ключи нельзя размещать публично — узнав его, третье лицо может обращаться к API ВКонтакте от имени нашего сообщества. Если ключ был скомпрометирован, необходимо удалить его из списка — после этого он станет недействителен.

**4.4 Получение событий и обновлений.** Чтобы реагировать на какие-либо события, наш скрипт должен о них узнать. Для этого существуют два подхода — Callback API и Long Poll. В данной ВКР настройка Callback будет осуществляться с помощью REST API.

**4.4.1. Отправка и прием сообщений.** Для отправки сообщения используется метод `messages.send`. С ключом доступа сообщества можно совершать до 20

запросов в секунду. Можно сократить число запросов к API, если группировать сообщения с одинаковым текстом (параметр `message`) для разных получателей — для этого перечислим идентификаторы получателей в параметре `user_ids`.

**5 Хранение состояний.** Для хранения информации будем использовать СУБД PostgreSQL. В базе данных будем хранить обращения пользователей, текущее состояние в диалоге и модели автомобилей.

Все пользователи, обратившиеся к чат-боту заносятся в таблицу «chats».

- `id` - идентификатор обращения;
- `uid` - идентификатор пользователя в социальной сети;
- `state` - JSON данные о текущем состоянии в диалоге;
- `created_at` - время первого обращения;
- `updated_at` - время последнего обращения;
- `user_brand` - информация о модели автомобиля.

Для того, чтобы чат-бот запоминал состояния в диалоге, реализуем конечный автомат состояний.

- `Start` - начальное состояние присваивается когда пользователь в первый раз обращается к чат-боту, либо при вызове команды «старт».
- `Reply1` - выбор модели автомобиля. Из любого состояния можно перейти в `Reply1`, введя команду «Изменить автомобиль».
- `Reply2` - выбор типа вопроса. «Изменить автомобиль» для перехода в `Reply1`.
- `Reply3` - поиск детали. Бот выводит цены на автозапчасть у разных поставщиков. «Изменить автомобиль» для перехода в `Reply1`.
- `Reply4` - пользователь задает вопрос. Текст сообщения проходит через нейронную сеть, определяет категорию вопроса и формирует ответ. «Изменить автомобиль» для перехода в `Reply1`.
- `Reply7` - уточнение вопроса, связанного с ремонтом. Выводит в качестве результата текст. «Изменить автомобиль» для перехода в `Reply1`.



- Reply6 - уточнение вопроса по замене детали. Выводит в качестве результата изображение. «Изменить автомобиль» для перехода в Reply1.
- Reply5 - уточнение вопроса по сроку замены. Выводит в качестве результата видеофайл. «Изменить автомобиль» для перехода в Reply1.

**6 Нечёткий поиск.** Расстояние Дамерау-Лавенштейна. Нечёткий поиск - это поиск информации, при котором выполняется сопоставление информации заданному образцу поиска.

Алгоритмы нечеткого поиска являются основой систем проверки орфографии и полноценных поисковых систем вроде Google или Yandex. Например, такие алгоритмы используются для функций наподобие «Возможно вы имели в виду ...» в тех же поисковых системах [5, 6].

Расстояние Дамерау-Лавенштейна показывает, за сколько операций удаления, вставки, замены и транспозиции символов можно перейти от одной строки к другой. По данным строкам функция будет выдавать количество операций для преобразования одной строки в другую [7, 8].

**7 Импорт данных из веб-страниц.** Чтобы информационно-справочная система могла быстро находить нужную автозапчасть по артикулу или по названию, необходимо организовать выгрузку цен с интернет-магазинов. С помощью модуля BeautifulSoup импортируем название детали и цену.

Beautiful Soup - это парсер для синтаксического разбора файлов HTML/ XML, написанный на языке программирования Python, который может преобразовать даже неправильную разметку в дерево синтаксического разбора. Он поддерживает простые и естественные способы навигации, поиска и модификации дерева синтаксического разбора.

**8 Машинное обучение.** В данной ВКР для определения категории вопроса используется машинное обучение.

Машинное обучение - это подобласть искусственного интеллекта и термин — им описывают методы построения алгоритмов, которые учатся на своем опыте, без написания специальной программы. Например, если нужно, чтобы алгоритм определял лица, то необходимо показать ему десять тысяч разных

лиц, отметить, где именно находится лицо, и тогда программа научится определять его самостоятельно [9].

Искусственный нейрон — это по сути функция, которая преобразует несколько фактов в один. Факты имеют вес, который повлияет на то, как факт будет использован нейроном [10].

Глубинное обучение – это по сути техника обучения нейросети, которая использует множество слоев для решения сложных проблем с помощью шаблонов.

**8.1 Постановка задачи.** Решаемая в данной работе задача классификации более формально ставится так: имеется множество объектов, разделённых некоторым образом на классы. Каждый объект задаётся в виде числового вектора (вектора атрибутов) фиксированной длины. Задано конечное множество объектов, для которых известно, к каким классам они относятся. Это множество называется обучающей выборкой. Требуется построить алгоритм, способный классифицировать произвольный объект из исходного множества.

Нас интересует задача, в которой объекты могут принадлежать одному из нескольких классов. На стадии определения типа вопроса существует два класса:

- класс 0 - сообщение пользователя относится к вопросу по ремонту автомобиля;
- класс 1 - сообщение пользователя относится к вопросу по техническому обслуживанию автомобиля;

**8.2 TensorFlow.** Чтобы решить задачу классификации будем использовать открытую программную библиотеку для машинного обучения TensorFlow.

TensorFlow также производит сложные вычисления вне Python, но он идет еще дальше чтобы избежать дополнительной нагрузки. Вместо того, чтобы запускать одну дорогую операцию независимо от Python, TensorFlow позволяет нам описать граф взаимодействующих операций, которые работают полностью вне Python.

**8.3 Архитектура нейронной сети.** В данной ВКР разработана нейронная сеть с тремя скрытыми слоями. Задача каждого скрытого слоя заключается в том, чтобы превратить входные данные во что-то, что мог бы использовать слой вывода, чтобы выдать правильный результат.

Каждый узел (нейрон) умножается на вес, то есть имеет значение веса. В ходе обучения нейронная сеть регулирует эти показатели, чтобы выдать правильные выходные значения. Нейросеть также добавляет смещение, переключая функцию активации влево или вправо, что может иметь решающее значение для успешного обучения.

Далее в нашей архитектуре данные передаются функции активации, которая определяет окончательный вывод каждого узла сети. Например, представим, что каждый узел — это светодиод, а функция активации указывает, будет ли светодиод светиться или нет

**8.4 Обучение нейронной сети.** Веса и смещения хранятся в переменных `tf.Variable`, которые содержат состояние в графе между вызовами `run()`. Чтобы `tensorflow` мог обучать модель также необходимы функция потерь и алгоритм оптимизации.

Существует много методов, как посчитать потерю. Для задач на классификацию, лучшим способом вычисления ошибки является метод перекрестной энтропии.

Для обучения сети используется алгоритм оптимизации Adam.

**8.5 Запуск графа и получение результата.** Будем использовать набор данных, содержащий тысячу постов на 40 тем. Чтобы загрузить такой объем информации, воспользуемся библиотекой `scikit-learn`.

`Scikit-learn` работает с двумя подмножествами: одним для обучения, вторым для проверки.

**9 Формирование ответа.** Полный скрипт для чат-бота представлен в приложении Г. В этой главе данной выпускной квалификационной работы соединяются все составляющие, рассмотренные в предыдущих главах.

Одна из функций чат-бота - поиск детали по артикулу или по названию детали в состоянии `Reply3`. Для этого в главе 7 был рассмотрен импорт данных с веб-страниц.

В состоянии `Reply4`, с помощью нейронной сети, определяется категория вопроса, префикс, если он есть и формируется ответ. Если вопрос задан с префиксом «как поменять/заменить», то в качестве ответа бот присылает видеозапись с инструкцией. Если префикс вопроса совпадает со строкой «когда менять», то, с помощью встроенных методов `VK API`, бот отправляет

пользователю изображение с информацией. Иначе, если, чат-бот определяет категорию «ремонт», то в зависимости от узла автомобиля он оценивает сложность ремонта, чтобы предупредить пользователя.

Для реализации отправки видеофайла в качестве ответа воспользуемся видеохостинговой компанией «Youtube», предоставляющей пользователям услуги хранения, доставки и показа видео.

Определив категорию элемента замены в автомобиле будем отправлять в браузерную строку его наименование и наименование модели автомобиля пользователя.

**10 Запуск проекта.** Для запуска проекта необходимо установить модули `vk_api`, `psycopg2`, `bottle`, `requests`, `numpy`, `tensorflow`, `BeautifulSoup`.

**Заключение.** В рамках настоящей выпускной квалификационной работы был проведён анализ проблем, связанных с ремонтом и обслуживанием ТС. В данной ВКР даётся описание работы информационно-справочной системы по ремонту и обслуживанию ТС, ориентированной на автолюбителей.

Описывается интерфейс VK API, его основные параметры, преимущества и возможности. Реализовано обращение к чат-боту, с помощью VK API.

Выстроена и подробно разобрана структура диалога между пользователем и ботом.

Помимо этого, при помощи технологии TensorFlow, была создана многослойная нейронная сеть, рассмотрена её архитектура и обучение. Показан запуск модели и результаты её тестирования.

В финальной части работы были рассмотрены функции для формирования ответа и интеграция нейронной сети в структуру чат-бота. Нейронная сеть в данной работе выполняет задачу классификации вопросов пользователя, что помогает определить конкретный узел автомобиля.

Результатом данной работы является информационно-справочная система, работающая в виде чат-бота в социальной сети «ВКонтакте». Чат-бот имеет выстроенную структуру диалога и интеграцию с многослойной нейронной сетью.

Поставленная задача была решена достаточно полно. Программу можно усовершенствовать и со временем дорабатывать. Обращаться к разработанному чат-боту может любой зарегистрированный пользователь.

## Список использованных источников

- 1 Академик - Электронный словарь. Виртуальный собеседник [Электронный ресурс] сайт. URL: <https://dic.academic.ru/dic.nsf/ruwiki/107552> (дата обращения 20.05.2018) Загл. с экрана. Яз. рус.
- 2 Шовин В.А. Программа chatbot - чат-бот или виртуальной собеседник // Математические структуры и моделирование. 2016. №4(40). С. 96-101
- 3 VK Developers. API для чат-ботов. [Электронный ресурс] сайт. URL: [https://vk.com/dev/bots\\\_docs](https://vk.com/dev/bots\_docs) (дата обращения 25.10.2017) Загл. с экрана. Яз. рус.
- 4 Habr — ресурс для IT-специалистов. Нечёткий поиск в тексте и словаре. [Электронный ресурс] сайт. URL: <https://habr.com/post/114997/> (дата обращения 15.02.2018) Загл. с экрана. Яз. рус.
- 5 Шовин В.А. Программа chatbot - чат-бот или виртуальной собеседник // Математические структуры и моделирование. 2016. №4(40). С. 96-101
- 6 Мосалев П.М. Обзор методов нечеткого поиска текстовой информации // Математика и Математическое моделирование. МГТУ им. Н.Э. Баумана. Электрон. журн. 2017. № 03. С. 64-76
- 7 Левенштейн, В.И. Двоичные коды с исправлением выпадений, вставок и замещений символов // Доклады Академии наук СССР. – 1965. – Т. 163. № 4. – С. 845–848.
- 8 Kukich K. Techniques for automatically correcting words in text //ACM Computing Surveys (CSUR). – 1992. – Т. 24. – №. 4. – С. 377-439.
- 9 Хайтек. Что такое deep learning и почему о нем все говорят? [Электронный ресурс] сайт. URL: <https://hightech.fm/2018/04/18/what-is-deep-learning> (дата обращения 20.05.2018) Загл. с экрана. Яз. рус.
- 10 Интернет-журнал про образовательную робототехнику и роботов. Нейросети. [Электронный ресурс] сайт. URL: [http://robotoved.ru/artificial\\_neural\\_network/](http://robotoved.ru/artificial_neural_network/) (дата обращения 10.01.2018) Загл. с экрана. Яз. рус.