Федеральное государственное автономное образовательное учреждение высшего образования

"Российский университет транспорта" (МИИТ)

**Институт управления и цифровых технологий**

Кафедра «Цифровые технологии управления транспортными процессами»

Курсовой проект

по дисциплине

**«Машинное обучение и анализ данных»**

Студенты: Юханссон М.Л.

Карпухин М.А.

Медведев А.О.

Группа: УВП-171

Проверил: Ахмед С.Х.

Москва 2023

**Оглавление**

[Введение 2](#_Toc635835347)

[Цели и задачи 3](#_Toc1072508309)

[Технологический стек 3](#_Toc1408986553)

[Поиск данных для обучения модели 3](#_Toc856754973)

[Разработка решения 5](#_Toc2006435384)

[Результаты выполнения работы 7](#_Toc1928650134)

# **Введение**

На данный момент в интернете существует такая проблема, как проблема оставляемых пользователями некорректных оценок в отзывах о товарах или о выполнении услуг. Нередко вызванные разовым ситуационным недопониманием и выставленные необоснованно, негативные отзывы могут навредить репутации и заработку добросовестных компаний и исполнителей. В данном курсовом проекте наша бригада проанализирует рейтинг отзывов об оказанных услугах на примере отзывов об отелях.

## **Цели и задачи**

Целью данного курсового проекта является разработка модели искусственного интеллекта, которая будет способна анализировать отзывы и корректно определять оценку конкретного отзыва.

В ходе выполнения курсового проекта наша бригада поставила перед собой следующие задачи:

1. Сбор и обработка большого объёма данных об отзывах и оценках клиентов.
2. Разработка алгоритмов машинного обучения для анализа данных и выявления общих проблем и недостатков этих подходов.
3. Обучение модели (нескольких моделей) искусственного интеллекта на основе собранных данных, чтобы модель могла анализировать характер отзыва и выставлять ему конкретную оценку.

## **Технологический стек**

1. Pandas — библиотека для работы с большими объёмами данных. Объект DataFrame позволяет манипулировать данными и преобразовывать их в кривые, диаграммы, графики и гистограммы.
2. NumPy — библиотека для работы со сложными математическими функциями и многомерными массивами.
3. Scikit-Learn (sklearn) — пакет для машинного обучения модели.
4. Natural Language Toolkit (NLTK) — пакет библиотек для обработки естественных языков и работы со словесными конструкциями.

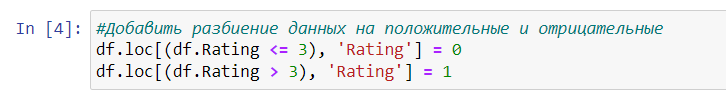
## **Поиск данных для обучения модели**

Данный этап предназначен для определения направления данных. Нашей бригадой был использован готовый датасет с сервиса Kaggle, содержащий в себе 20 тысяч записей с отзывами об отелях разного характера с сайта Trip Advisor.



С помощью этого датасета мы постараемся проанализировать, какие слова в отзыве делают его негативным, а какие наоборот — придают ему более положительную окраску.

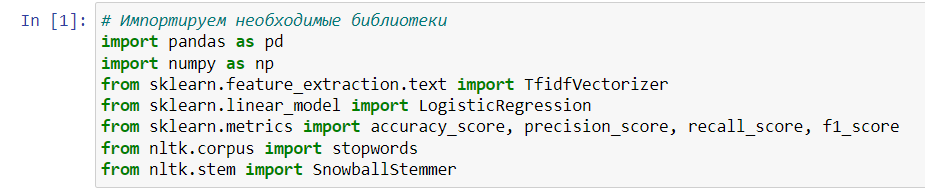
Так как часто невозможно отличить отзывы с похожими оценками, было принято решение разбить все отзывы на положительные и отрицательные, данный подход способствовал резкому повышению точности итогового решения.



## **Разработка решения**

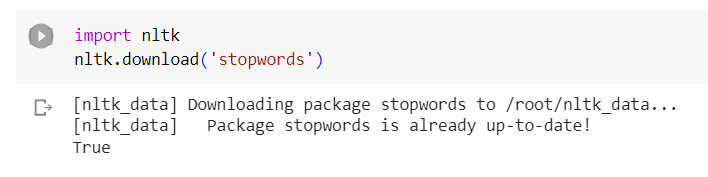
Для разработки модели, её обучения и демонстрации её работы мы использовали приложение Jupyter Notebook — интерактивный веб-блокнот для разработки на языке Python, позволяющий писать и воспроизводить код отдельными блоками.

На начальном этапе мы подключаем все необходимые для работы пакеты и библиотеки.

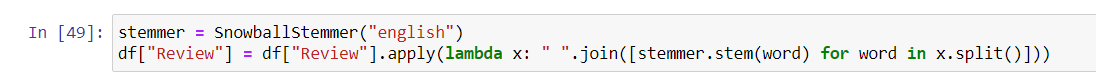


Далее считываем csv-файл с датасетом и записываем его в упомянутый ранее объект типа DataFrame.

С помощью библиотеки Natural Language Toolkit мы загружаем массив английских слов, которые обычно не влияют на эмоциональную окраску текста и не передаёт его конечной мысли — так называемых «стоп-слов». Чаще всего, это предлоги, союзы, артикли и местоимения.

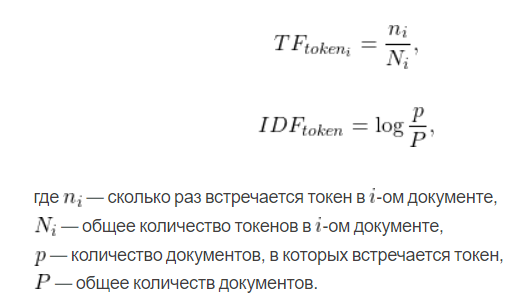


Далее был написан код, который использует алгоритм SnowballStemmer из библиотеки Natural Language Toolkit. SnowballStemmer предназначен для стемминга — процесса приведения слова к его основе путём избавления от префиксов, суффиксов и окончаний. Основа слова не обязательно должна совпадать с его морфологическим корнем. С помощью метода join() приведём все оставшиеся в столбце «Review» слова к виду стеммов и вклеим обратно в отзыв с помощью метода join() и сохраняются в том же столбце «Review».



После этого мы проводим векторизацию текста из столбца «Review». В данном случае будет использоваться метод TF-IDF — произведение частоты появления слова в документе на инверсию частоты документа. В TF-IDF редкие слова и слова, которые встречаются во всех документах, несут мало информации. Кроме того, IDF можно считать и другими способами, например, в Python-библиотеке Scikit-learn этот параметр гибко регулируется. TF-IDF считается следующим образом:

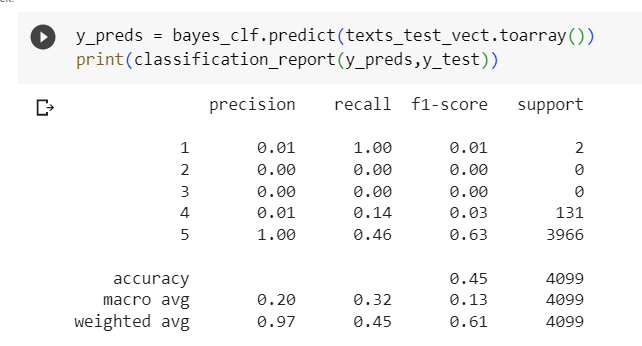




Был написан код, который использует TfidfVectorizer из библиотеки SciKit-Learn для векторизации текста в столбце «Review». Затем метод fit\_transform() применяется к столбцу «Review», чтобы создать матрицу признаков X, где каждый столбец представляет собой вектор для соответствующего отзыва. Столбец «Rating» используется как целевая переменная y.

Данные были разбиты на две выборки: обучающую и тестовую. В итоге модель была обучена на полученной выборке. В качестве модели была применена логистическая регрессия. Обученная модель будет использоваться для предсказания целевой переменной на тестовой выборке.

Также была попытка использования варианта с наивным Байесом, но она оказался неудачной, точность оказалась слишком низкой.



## **Результаты выполнения работы**

В результате выполнения курсового проекта нами была спроектирована и обучена модель, способная с довольно высокой точностью определить, является ли отзыв положительным или отрицательным. В конце выполнения работы была произведена оценка точности, и она показала следующие результаты.

