# ОТЧЕТ ПО РАБОТЕ «ТЕСТОВОЕ ЗАДАНИЕ ДЛЯ JUNIOR-СПЕЦИАЛИСТОВ ПО НАПРАВЛЕНИЮ DATA SCIENCE»

В данной работе была попытка создать модель, способную предсказывать тип отзыва на фильм(положительный, отрицательный) и оценку, которую поставит пользователь(1,2,3,4,7,8,9). Нейтральные отзывы и соответствующие им оценки не рассматривались и не учитывались. Работа выполнялась на языке python.В качестве обучающего набора брались все данные положительных и отрицательных отзывов из папки train, в качестве оценочного – из папки test.

Этапы работы: загрузка данных, предобработка, выбор оптимального алгоритма, обучение, оценка результатов, создание прототипа на открытой площадке. Остановимся подробнее на каждом.

**Загрузка данных:** использовались библиотеки os и tqdm.

**Предобработка:** удалены теги html, удалены повторяющиеся элементы, пропуски данных. Убраны стоп-слова(библиотека nltk), знаки препинания(re) и проведено сравнение эффективности модели с лемматизацией и стемминга. Лемматизация дает лучший результат по большинству алгоритмов.

В случае многоклассовой классификации добавлены новые переменные: lenght(количество слов в предложении), punct%(процент знаков препинания в предложении), polarity(полярность текста) и subjectivity(субъективность) последние 2 из библиотеки TextBlob. Кроме того, добавлена бинарная переменная binary, определяющая является ли отзыв положительным или отрицательным. Такую информацию можно получить с помощью алгоритма бинарной классификации, который является более точным. Добавление последней переменной помогло увеличить точность модели на 10%. Далее в обоих случая для бинарной и многоклассовой классификации проводилась токенизация текста методом TfidfVectorizer.

**Выбор оптимального алгоритма:** рассматривались основные алгоритмы классического машинного обучения(логистическая регрессия, XGBoost, SGD Classifier, LinearSVC, метод опорных векторов. А также полносвязная нейронная сеть. Для всех алгоритмов проводился поиск по сетке оптимальных параметров. Применялась кросс валидация.

Кроме того, была попытка понижения размерности с помощью метода главных компонент. На 100 искусственных переменных результат оказался несколько хуже, однако скорость расчета возросла.

**Оценка результатов:** проводилась на тестовой выборке. В качестве метрики качества бралась accuracy. Но также оценивались precision, recall и f1-score.

Для многоклассовой классификации была достигнута точность **90%** на тестовой выборке(XGBoost)

0.90128

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 1 | 0.81 | 0.95 | 0.87 | 5022 |
| 2 | 0.98 | 0.70 | 0.82 | 2302 |
| 3 | 0.96 | 0.76 | 0.84 | 2541 |
| 4 | 0.92 | 0.83 | 0.87 | 2635 |
| 7 | 0.98 | 0.94 | 0.96 | 2307 |
| 8 | 0.93 | 0.98 | 0.95 | 2850 |
| 9 | 0.96 | 0.94 | 0.95 | 2344 |
| 10 | 0.88 | 0.98 | 0.93 | 4999 |
| accuracy | | | 0.90 | 25000 |
| macro avg | 0.93 | 0.88 | 0.90 | 25000 |
| weighted avg | 0.91 | 0.90 | 0.90 | 25000 |

Для бинарной классификации была достигнута точность **98%**(нейронная сеть).

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.98 | 0.99 | 0.98 | 12500 |
| 1 | 0.99 | 0.98 | 0.98 | 12500 |
| accuracy | | | 0.98 | 25000 |
| macro avg | 0.98 | 0.98 | 0.98 | 25000 |
| weighted avg | 0.98 | 0.98 | 0.98 | 25000 |

**Создание прототипа:** на базе фреймворка Django разработан веб-сервис для ввода отзыва о фильме с автоматическим присвоением рейтинга (от 1 до 10) и статуса комментария (положительный или отрицательный). Однако, при выводе сервиса в открытый доступ, не удалось справиться с ограничением, связанным с работой алгоритма (не более определенного количества секунд, около 30). Сервис развернут наполовину(только для бинарной классификации) и для сокращения времени вместо нейронной сети там выставлена логистическая регрессия с точность 93% на тестовой выборке. На стационарном пк сервис работает и выводит на сервер оба элемента (расчет занимает несколько минут).

Ссылка на сервис: https://bc-egorov.herokuapp.com/reviews/

Ссылка на GitHub: https://github.com/AlexanderEgorovNRNUMEPHI/ML

Ссылка на черновые ноутбуки(при необходимости): https://drive.google.com/drive/folders/1X3LruYXYX4bIbdxXIg1DfkBOgGvS7a99?usp=sharing

Выполнил Егоров Александр