Практическое задание №4. Осень 2024

Постановка задачи

Целью работы является разработка метода, позволяющего решать задачу онлайн группировки текстов новостей. Метод по очереди получает тексты новостей и возвращает метку (номер) группы, к которой относится переданный документ. В одну группу должны попадать новости, относящиеся к описанию одного и того же события или темы.

В качестве тестового набора данных используются некоторые группы новостей из проекта Яндекс.Новости.

Решение задачи

Теоретические аспекты

Задача группировки новостей может решаться как задача кластеризации, которая относится к задачам обучения без учителя. Однако в отличие от классической кластеризации в рамках практического задания существует ground truth разбиение текстов по группам.
Особое внимание нужно обратить на то, что метод группировки сообщений принимает тексты новостей по одному и сразу же должен принять решение о том, к какой группе относится эта новость.

Дополнительную сложность задания добавляют существенные ограничения на размер итоговой модели.

Тестирование

На личной странице (practicum.tpc.ispras.ru/submissions/clusterize) находится статистика со всеми результатами в т.ч. результатами последнего тестирования (дата, метрики качества).
На странице practicum.tpc.ispras.ru/results/clusterize доступны результаты всех участников.
Решения перезапускаются раз в неделю по средам в 00:00.

Практические Аспекты

Решения должны быть написаны на языке Python (версия 3.10.7). Можно использовать все стандартные библиотеки, а также:

- pandas==1.5.1
- pymystem3==0.2.0
- regex==2023.10.3
- gensim==4.2.0
- transformers==4.23.1
- tensorflow==2.10.0
- tensorflow-addons==0.18.0
- scikit-learn==1.1.3
- torch==2.5.1+cpu
- torchvision==0.20.0+cpu
- torchaudio==2.5.1+cpu
- nltk==3.9.1 и все его пакеты
- xgboost==2.1.2
- catboost==1.2.7

В случае необходимости использования дополнительных библиотек, сообщите об этом организаторам практикума (библиотеки будут добавлены для всех студентов). Доступ в интернет на проверяющей машине закрыт.

Загрузка решения

Загружаемый файл должен представлять собой zip архив с любым именем. Архив должен обязательно содержать:

- Решение в файле solution.py. В файле должен быть класс `Solution`, содержащий метод `predict(self, text: str) -> str`, который получает на вход текст новости и возвращает метку группы.
- Описание применяемых методов в файле description.txt. Пожалуйста, напишите подробное описание, какие методы и признаки использовались. Это описание будет выложено вместе с решением после завершения курса.
- Все используемые ресурсы, необходимые для корректной работы метода.
- Код, позволяющий выполнить обучение модели (при использовании машинного обучения) и инструкции по его запуску. Инструкции должны быть достаточно подробными для возможности воспроизведения модели без дополнительных консультаций с автором метода. Отсутствие таких инструкций будет приравниваться к невозможности воспроизведения. При решении можно использовать дополнительные ресурсы для обучения, но они должны быть доступны организаторам в момент воспроизведения решения.

Пример решения, помещающего все тексты в одну группу:

```
class Solution:
    def predict(self, text: str) -> str:
        return "1"
```

Ограничения

- Каждую неделю можно послать не более 10 решений.
- Внимание! Итоговое тестирование будет проводиться на последнем загруженном решении.
- Время тестирования одного решения не должно превышать 30 минут.
- Тестовый набор данных состоит из примерно 1500 текстов.
- Размер загружаемого архива не должен превышать 100Мб.
- На проверяющей машине доступно 16 Гб оперативной памяти и 8 CPU.

Внимание: при использовании PyTorch или Tensorflow (или другого фреймворка машинного обучения, использующего многопоточность) явным образом установите ограничение на максимальное количество потоков.

- set_num_threads для PyTorch
- set_inter_op_parallelism_threads для Tensorflow

Оценка качества

Для оценки качества используется B^3F_1-мера:

$$R = \frac{1}{N}\sum_{k\in K}\sum_{s\in S}\frac{|k \cap s|^2}{|k|} \qquad P = \frac{1}{N}\sum_{s\in S}\sum_{k\in K}\frac{|s \cap k|^2}{|s|} \qquad F_1 = \frac{2PR}{P+R},$$

где K - ожидаемый набор групп, S - предсказанный набор групп, $|k|$ - размер группы, N - общее количество текстов.