Федеральное государственное автономное образовательное учреждение высшего образования

«Национальный исследовательский университет

«Высшая школа экономики»

Факультет компьютерных наук

Отчёт о прохождении производственной (научно-исследовательской) практики

Выполнил студент: Комарова Анна

Образовательной программы: НОД ИССА

Отчет проверил руководитель практики от НИУ ВШЭ: Паринов Андрей Андреевич

Подпись студента: 

Москва, 2022

Введение

В качестве направления работы научно-исследовательской практики была выбрана адаптация и воспроизведение алгоритма кластеризации для продуктов онлайн-магазинов, представленного в статье «Topic Modelling with BERT»[1]. Решение задачи анализа и классификации описания продуктов онлайн-магазинов сводится к построению тематической модели (тематическое моделирование, topic modelling).

Тематическая модель – это модель коллекции текстовых документов, которая определяет, к каким темам относится каждый документ коллекции. Алгоритм построения тематической модели получает на входе коллекцию текстовых документов. На выходе для каждого документа выдаётся числовой вектор, составленный из оценок степени принадлежности данного документа каждой из тем. Существует несколько базовых алгоритмов тематического моделирования: алгоритм скрытого распределения Дирихле (Latent Dirichlet Allocation, LDA), алгоритм неотрицательной матричной аппроксимации (Non-Negative Matrix Factorization, NMF).

В данной работе я, следуя за автором статьи «Topic Modelling with BERT», концентрирую внимание на модели BERT. Модель BERT (Bidirectional Encoder Representations from Transformers) – модель на основе трансформеров, применение таких предобученных моделей содержит более точные представления слов и предложений. BERT обучается одновременно на двух задачах — предсказания следующего предложения (next sentence prediction) и генерации пропущенного токена (masked language modeling), таким образом, BERT обучает контексто-зависимые представления.

Данные

В работе используются данные по 3472 товарам интернет-магазина Ozon, включающие название товара, его краткое и полное описание, стоимость и принадлежность категориям. В качестве предварительной обработки текста были удалены стандартные непечатаемые символы, удалены дубликаты.

Кластеризация текстов

Итак, первым шагом в разработке алгоритма кластеризации являлось получение векторного представления текстовых данных описания продуктов. Для этой цели

[1] https://www.kdnuggets.com/2020/11/topic-modeling-bert.html

использовалась модель BERT, реализованная в библиотеке bertopic[2]. Для работы с текстами на русском языке использовалась многоязыковая модель distiluse-base-multilingual-cased-v1.

Далее перед применением моделей кластеризации необходимо воспользоваться алгоритмами снижения размерности. Для этой цели я использовала алгоритм UMAP[3]. Решая дилемму между сохранением информации и структуры входных данных и снижением размерности для задача кластеризации, я остановилась на следующих параметрах модели: размерность модели = 5, количество соседей = 15.

Для решения задачи кластеризации использовался алгоритм иерархической пространственной кластеризации на основе плотности - HDBSCAN. Алгоритм HDBSCAN рассматривает кластеры как области высокой плотности, отдельно от областей низкой плотности. Преимуществом этого алгоритма является его подход к обработке выбросов – таким объектам не присваиваются кластеры принудительно, они остаются выбросами. На рис. 1 представлены результаты кластеризации, для визуализации объектов размерность была снижена до 2. Как можно видеть, пространство объектов – с низкой плотностью. Всего был сформирован 501 кластер.

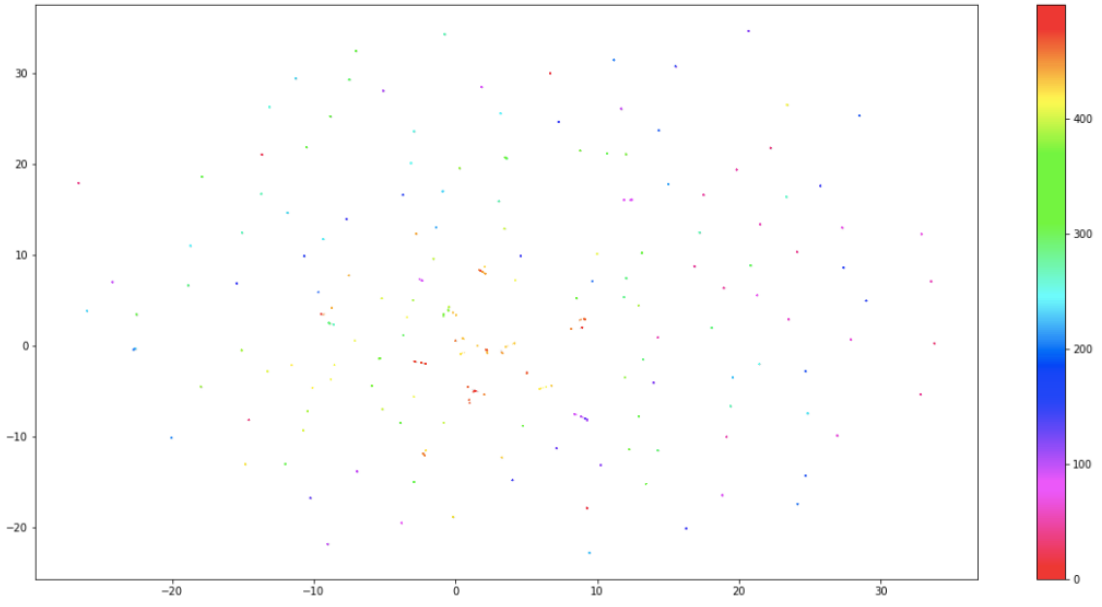

Рисунок 1

Оценка качества алгоритма кластеризации происходила на основе метрики «силуэт» (silhouette score), которая учитывает среднее внутрикластерное расстояние и

[2] https://github.com/MaartenGr/BERTopic
[3] https://github.com/lmcinnes/umap

среднее расстояние до ближайшего кластера по каждому объекту. Применение алгоритма HDBSCAN показало лучше результаты, чем алгоритм OPTICS.

Теперь, когда кластеры сформированы, интересно выяснить, чем они отличаются и какие темы являются определяющими для каждого кластера. Для этих целей использовалась модификация TF-IDF для классов[4]:

$$c - TF - IDF_i = \frac{t_i}{w_i} log \frac{m}{\sum_j^n t_j}$$

В отличие от стандартного TF-IDF в c-TF-IDF показатель считается для каждого слова и каждого класса, где *i* – класс, *t* – конкретное слово, w_i – число слов в классе, *m* – общее количество документов, а *n* – общее количество классов.

Теперь, когда для каждого слова есть статистическая величина его важности внутри кластера, возьмем для формирования тем внутри кластеров топ 10 слов по важности.

Ниже приведены топики для некоторых кластеров, в скобках указаны значения c-TF-IDF показателя, чем выше значение, тем более характерным является указанное слово для кластера.

```
top_n_words[182][:10]
```

```
[('520', 0.21619223396455733),
 ('ion', 0.13681928367879104),
 ('литий', 0.12542479649627727),
 ('ионный', 0.12542479649627727),
 ('перезаряда', 0.12542479649627727),
 ('7в', 0.1092074110465495),
 ('защитой', 0.1092074110465495),
 ('ма', 0.10805107643361295),
 ('li', 0.10561156822514135),
 ('серебристый', 0.09984007481051423)]
```

```
top_n_words[197][:10]
```

```
[('14', 0.27620949135216044),
 ('авто', 0.14966382980100532),
 ('см', 0.14953659832524366),
 ('магнитная', 0.14717644068868604),
 ('вариант', 0.11210745679050513),
 ('внешняя', 0.09327601224362643),
 ('антенн', 0.09287350989253262),
 ('антенны', 0.09248431336037567),
 ('кабеля', 0.08663577018282263),
 ('длина', 0.08516912144078427)]
```

```
top_n_words[100][:10]
```

```
[('480mm', 0.5481789167680045),
 ('s6', 0.4804271865879051),
 ('d6mm', 0.4734073086299571),
 ('100', 0.36162736398103656),
 ('ast', 0.36080125526020124),
 ('телескопическая', 0.3440266967146592),
 ('радиоантенна', 0.1768283788759631),
 ('антенна', 0.16865030199095055),
 ('япония', 0.0),
 ('mhz', 0.0)]
```

Результаты

В результате практической работы с помощью алгоритмов кластеризации были выделены группы кластеров и выявлены темы для наборов описаний товаров. Для выполнения данных задач были изучены и использованы следующие алгоритмы: BERT, HDBSCAN, UMAP, и библиотеки Python bertopic, umap.

[4] c-TF-IDF https://github.com/MaartenGr/cTFIDF