Домашнее задание 3

В данном домашнем задании Вам предстоит реализовать простые модели полнотекстового поиска – tfidf и BM25, а также подобрать оптимальные параметры ранжирования на данной коллекции запросов и документов.

1. Датасет

В качестве документов для поиска, требуется использовать статьи википедии, скачанные в одном из предыдущих домашних заданий. Чтобы процесс занимал разумное время, поиск требуется производить только по некоторому заданному набору документов. Список целевых статей описан в файле *selected_docs.tsv* по одной на строку.

Реализуйте получение содержимого документа по названию статьи в соответствии с выбранной вами схемы хранения документов. Также, преобразуйте содержимое в текст (не html) любым разумным способом. Постарайтесь при этом вырезать заведомо бесполезные для поиска данные со страницы.

2. Поиск

Реализуйте разбиение текста на термы. Рассчитайте статистики термов и документов, которые понадобятся для реализации моделей tfidf и BM25: частота терма для документа, обратная документная частота терма и прочие.

Реализуйте поиск лучших 10 документов в модели tfidf и BM25 с параметрами b = 1, k1 = 1, k2=1.

Строить инвертированный индекс не требуется (но и не запрещается).

3. Оптимизация качества

Для измерения качества поиска вам предоставляется список из пар (запрос, название статьи), которая означает, что по данному запросу данная статья является релевантной (а остальные – нерелевантны). Пары описаны в файле *queries.tsv* по одной на строку.

Оценим поиск по нескольким метрикам: accuracy – доля запросов, где на первой позиции был найден релевантный документ; accuracy@10 – доля запросов, где релевантный документ попал в первую десятку, mrr@10 – средняя обратная позиция релевантного документа в первой десятке.

Сравните реализованные Вами алгоритмы tfidf и BM25 по этим метрикам.

Подберите оптимальные параметры BM25 для этого набора запросов и документов.

~~Дополнительно~~

4. Поиск в векторной модели (+1 балл)

Используйте готовые эмбединги или энкодер нейронной модели, и преобразуйте с помощью него запросы и документы в вектора небольшой размерности.

Реализуйте поиск лучших документов по косинусной мере или скалярному произведению между векторами запроса и документа. Сравните результаты с моделями из предыдущих пунктов по метрикам и позапросно – выигрывает ли векторный вариант на тех примерах, где предыдущие модели не справляются?