Московский авиационный институт (национальный исследовательский университет)

Факультет информационных технологий и прикладной математики

Кафедра вычислительной математики и программирования

Лабораторная работа №4 по курсу «Информационный поиск»

Студент: П. А. Харьков
Преподаватель: А. А. Кухтичев
Группа: М8О-406Б

Дата:
Оценка:
Подпись:

Москва, 2023

Лабораторная работа №4

Необходимо сделать ранжированный поиск на основании схемы ранжирования TF-IDF.

1 Описание

После индексации всех файлов, я начинаю индексировать данные для ранжирования с помощью TF-IDF. Для каждого слова в каждой зоне обратного индекса, я прохожусь по документу:

- Подсчитывается количество вхождений слов в зоне документа.
- Вычисляется TF по формуле

$$\mathrm{TF}(t,d) = \frac{f_{t,d}}{\max f_{w,d} : w \in d}$$

  где $f_{t,d}$ - количество раз, когда термин t появляется в документе d, а $\max f_{w,d} : w \in d$ - максимальное количество раз, когда любой термин появляется в документе d.

- Вычисляется IDF по формуле

$$\mathrm{IDF}(t,D) = \log \frac{|D|}{|d \in D : t \in d|}$$

  где $|D|$ - общее количество документов в коллекции, а $|d \in D : t \in d|$ - количество документов в коллекции, которые содержат термин t.

- Получается значение

$$\mathrm{TF} - \mathrm{IDF}(t,d,D) = \mathrm{TF}(t,d) \times \mathrm{IDF}(t,D)$$

2 Исходный код

```
double SearchEngine::tf(int term_count, int zone_word_count) {
    return (double)term_count / zone_word_count;
}

double SearchEngine::idf(int doc_count, int total_docs) {
    return log((double)total_docs / (1 + doc_count));
}

int SearchEngine::term_count(doc_id doc, Zone zone, const string& word) {
    return count(forward_index[doc][zone].begin(), forward_index[doc][zone].end(),
        get_word_id(word));
}

void SearchEngine::compute_tf_idf() {
    int total_docs = docs_dict.size();
    for (const auto& word_entry : inverted_index) {
        const string& word = word_entry.first;
        for (const auto& zone_entry : word_entry.second) {
            Zone zone = zone_entry.first;
            const vector<doc_id>& doc_ids = zone_entry.second;
            int doc_count = doc_ids.size();
            for (doc_id doc : doc_ids) {
                int term_cnt = term_count(doc, zone, word);
                int zone_word_count = doc_zone_word_count[doc][zone];
                double tf_value = tf(term_cnt, zone_word_count);
                double idf_value = idf(doc_count, total_docs);
                tf_idf_index[doc][zone][get_word_id(word)] = abs(tf_value * idf_value);
            }
        }
    }
}
```

3 Выводы

Выполнив четвертую лабораторную работу по курсу «Информационный поиск», я научился реализовывать ранжирование TF-IDF. В итоге, у меня получилось улучшить результаты при поисковой выдаче.

Список литературы

[1] Маннинг, Кристофер Д. Введение в информационный поиск [Текст] / Кристофер Д. Маннинг, Прабхакар Рагхаван, Хайнрих Шютце ; пер. с англ. М. Л. Суркова. - Москва : Вильямс, 2020. - 528 с.