Московский авиационный институт (национальный исследовательский университет)

Факультет информационных технологий и прикладной математики

Кафедра вычислительной математики и программирования

Лабораторная работа №2 по курсу «Информационный поиск»

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

Дата:
Оценка:
Подпись:

Москва, 2023

Лабораторная работа №2

Требуется построить поисковой индекс, пригодный для булева поиска, по подготовленному в ЛР1 корпусу документов. Требования к индексу:

- Самостоятельно разработанный бинарный формат представления данных. Формат необходимо описать в отчете, в побайтовой представлении.
- Формат должен предполагать расширение, т.к. в следующих работах он будет меняться под требования новых лабораторных работ.
- Использование текстового представления или готовых баз данных не допускается.
- Кроме обратного индекса, должен быть создан прямой индекс, содержащий в себе как минимум заголовки документов и ссылки на них.
- Для термов должна быть как минимум понижена капитализация.

В отчете должно быть отмечено как минимум:

- Выбранное внутреннее представление документов после токенизации.
- Выбранный метод сортировки, его достоинтсва и недостатки для задачи индексации.

1 Описание

Индексирование происходит следующим образом:

1. Выбирается обработанный в ЛР1 документ.
2. Из него считываются первые три строки: каждая отвечает за свою зону.
3. Затем для каждого слова в строке:
   (a) В обрабтный индекс для слова в зоне добавляется номер документа.
   (b) В прямой индекс для документа в зоне добавляется номер слова.

Для работы программы необходимо хранить словарь названий файлов, словарь слов, прямой индекс и обратный индекс. Каждый из этих обьектов я храню в отдельном файле. Обьекты сохраняются в бинарный файл наивным образом, в том виде, как они и хранятся в памяти.

2 Исходный код

```
word_id SearchEngine::get_word_id(const string& word) {
    static word_id next_word_id = 0;
    auto it = words_dict.find(word);
    if (it == words_dict.end()) {
        words_dict[word] = next_word_id;
        return next_word_id++;
    }
    return it->second;
}
void SearchEngine::index_zone(istringstream& iss, doc_id doc_counter, Zone zone) {
    string word;
    while (iss >> word) {
        word_id id = get_word_id(word);
        inverted_index[word][zone].push_back(doc_counter);
        forward_index[doc_counter][zone].push_back(id);
        doc_zone_word_count[doc_counter][zone]++;
    }
}
void SearchEngine::index_file(const string& file_path, int doc_counter) {
    ifstream infile(file_path);
    string title, first_paragraph, rest;

    getline(infile, title);
    getline(infile, first_paragraph);
    getline(infile, rest);

    infile.close();

    istringstream iss_title(title), iss_first(first_paragraph), iss_rest(rest);
    index_zone(iss_title, doc_counter, TITLE);
    index_zone(iss_first, doc_counter, FIRST_PARAGRAPH);
    index_zone(iss_rest, doc_counter, REST);
}
void SearchEngine::index_folder() {
    doc_id doc_counter = 0;
    for (const auto& entry : filesystem::directory_iterator(INPUT_DIR)) {
        if (entry.is_regular_file()) {
            string file_path = entry.path().string();
            docs_dict[doc_counter] = file_path;
            index_file(file_path, doc_counter);
            doc_counter++;
        }
    }
    compute_tf_idf();
}
```

3 Выводы

Выполнив вторую лабораторную работу по курсу «Информационный поиск», я научился реализовывать прямой и обратные индексы для того, чтобы в последствии использовать их для булева поиска. Наибольшей сложностью для меня в этой лабораторной работе было не написание правильной индексации файлов, а написание сохранения и загрузки проиндексированных данных в бинарные файлы.

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.