

**Московский авиационный институт
(национальный исследовательский университет)**

**Институт информационные технологии и прикладной
математики**

Кафедра вычислительной математики и программирования

Курсовой проект по курсу

«Информационный поиск»

ПРОВЕРКА ОРФОГРАФИИ ПРИ ПОИСКЕ

| | |
|----------------|---------------|
| Студент: | Е.М. Стифеев |
| Преподаватель: | А.А. Кухтичев |
| Группа: | М8О-109М-21 |
| Дата: | 03.12.21 |
| Оценка: | |
| Подпись: | |

Москва, 2021

Оглавление

| | |
|---|----|
| Задание | 2 |
| 1. Описание существующей поисковой системы | 3 |
| Корпус | 3 |
| Индекс | 5 |
| Поисковая система. Виды запросов | 10 |
| Поисковая система. Интерфейс | 11 |
| 2. Реализация проверки орфографии | 16 |
| Расстояние Левенштейна..... | 16 |
| Расстоянии Дameraу-Левенштейна. Некорректный алгоритм | 19 |
| Расстоянии Дameraу-Левенштейна. Корректный алгоритм | 20 |
| Окончательный алгоритм..... | 23 |
| 3. Исходный код | 26 |
| Структура проекта..... | 26 |
| Запуск и сборка..... | 27 |
| 4. Выводы и результаты | 32 |
| Литература | 34 |

Задание

Необходимо реализовать проверку опечаток в поисковых запросах для системы, разработанной по итогам лабораторных работ по курсу «Информационного поиска» и «Обработки текстов на естественном языке».

1. Описание существующей поисковой системы

Корпус

Поисковая система обрабатывает запросы для корпуса документов, хранящегося на диске.

По итогам лабораторных работ по курсу с помощью веб-скрапинга по нескольким сайтам был получен корпус документов (доступен по ссылке <https://cloud.mail.ru/public/ZfkX/gccM7hnDR>), который имеет следующую структуру:

- films1.txt (94 Мб, 15000 документов, UTF-8)
- films2.txt (96 Мб, 15000 документов, UTF-8)
- films3.txt (184 Мб, 15000 документов, UTF-8)
- films4.txt (219 Мб, 15000 документов, UTF-8)
- films5.txt (322 Мб, 15000 документов, UTF-8)
- films6.txt (711 Мб, 15000 документов, UTF-8)
- films7.txt (823 Мб, 15000 документов, UTF-8)
- films8.txt (226 Мб, 15000 документов, UTF-8)
- films9.txt (67 Мб, 15000 документов, UTF-8)
- films10.txt (75 Мб, 15000 документов, UTF-8)
- films11.txt (99 Мб, 15000 документов, UTF-8)
- films12.txt (78 Мб, 15000 документов, UTF-8)
- films13.txt (41 Мб, 6109 документов, UTF-8)

$$\Sigma_{Gb} = 2,899 \text{ Gb}, \Sigma_{docs} = 186109$$

Получение одного документа зачастую включало проход по нескольким html-страницам и обработку динамически подгружаемых страниц, поэтому общее количество обработанных страниц было >800'000.

Всего была обкачено три сайта:

- <https://www.kinopoisk.ru/>
- <https://shikimori.one/>
- <http://filmplace.ru/>

В каждом файле *.txt документы хранятся следующим образом:

- 1 строка 1 документ {....}
- 2 строка 2 документ {....}
- *n* строка *n* документ {....}

Каждый документ снабжён прямой ссылкой на источник, откуда был скачен, и хранит только выделенный из html-кода текст в кодировке UTF-8. Например, 234 строка файла films1.txt выглядит так:

```
{ "page_url": "https://www.kinopoisk.ru/media/article/1773537/", "title": "Артур Смольянинов: «Я сомневался, что смогу сыграть ангела»", "body": "2 января в российский прокат вышла романтическая комедия Веры Сторожевой „Мой парень — ангел“, главные роли в которой исполнили Артур Смольянинов и Анна Старшенбаум. Мы подготовили небольшой видеосюжет с участием создателей картины...Студентка Саша с большим трудом верит в чудеса. Ангелу Серафиму приходится приложить немало усилий, чтобы доказать ей, что ангелы существуют. Но он не учел одного: если девушка тебе поверит, она, скорее всего, тебя полюбит.\n\n\n\n\n\n\n\n\nАвтор: Дарико Цулая", "comments": ""}.
```

Индекс

Готовый индекс хранится в четырёх файлах (доступен по ссылке <https://cloud.mail.ru/public/wynT/adagiBjh9>):

- **docs_id.data** (42 Мб)

Файл служит для отображения индекса документа (doc_id) в его текстовое представление в файлах *.txt. Поддерживается переменная длина пути до файлов с документами.

Структура

| | | | | | |
|-------------------|----------------|----------------------|------------------|--------------------|--|
| n_docs | | | | | |
| offset[0] | offset[1] | ... | offset[n_docs-1] | offset[n_docs] | |
| n_chars[0] | name[0] | doc_offset[0] | | doc_size[0] | |
| n_chars[1] | name[1] | doc_offset[1] | | doc_size[1] | |
| ... | ... | ... | | ... | |
| n_chars[n_docs-1] | name[n_docs-1] | doc_offset[n_docs-1] | | doc_size[n_docs-1] | |

Описание полей

| Название | Тип | Назначение |
|--------------------------------------|--------|---|
| n_docs | uint | Число документов в корпусе |
| offset[0],..., offset[n_docs] | uint | Смещения в байтах до строк таблицы с описанием документов, расположенной ниже. Таким образом, если понадобится открыть документ с doc_id = 5, то можно будет сразу переместить головку диска (fseek) до offset[5], прочитать это поле и сразу сместиться до нужной строки в таблице на offset[5], чтобы попасть в начало n_chars[5]. offset имеет на один элемент больше чем нужно (offset[n_docs]), чтобы работала блочная индексация и слияние блоков |
| n_chars[0],..., n_chars[n_docs-1] | uint | Число символов wchar в абсолютном пути до файла, где хранится документ |
| name[0],..., name[n_docs-1] | *wchar | Абсолютный путь до файла *.txt в кодировке UTF-16, т.е. два байта на символ |

| | | |
|---|------|--|
| doc_offset[0],... doc_size[n_docs-1] | uint | Смещение в байтах до начала документа в файле *.txt |
| doc_size[0],... doc_size[n_docs-1] | uint | Размер документа в байтах. |

- **terms.data** (54 Мб)

Файл служит для хранения словаря с терминами и ссылок (смещений) на файл с словопозициями и координатами. Поддерживается переменная длина термина. Термины упорядочены в лексикографическом порядке.

Структура

| n_terms | | |
|--------------------|-----------------|-----------------------------|
| n_chars[0] | term[0] | offset_post_list[0] |
| n_chars[1] | term[1] | offset_post_list[1] |
| ... | ... | ... |
| n_chars[n_terms-1] | term[n_terms-1] | offset_post_list[n_terms-1] |

Описание полей

| Название | Тип | Назначение |
|--|----------|--|
| n_terms | uint | Число терминов в корпусе/ число списков словопозиций |
| n_chars[0],..., n_chars[n_terms-1] | uint | Число символов в термине |
| term[0],..., term[n_terms-1] | *wchar_t | Термин в кодировке UTF-16 |
| offset_post_list[0],..., offset_post_list[n_docs-1] | uint | Смещение в файле со словопозициями |

- **postings_list.data** (2.68 Гб)

Файл служит для хранения словопозиций и координат терминов в документе. Слопозиции упорядочены по возрастанию идентификаторов документов.

Структура

| n_terms | | | | | | |
|-----------|-----------|-------------|-----------------|---------------|-----------|----------|
| n_docs[0] | doc...doc | freq...freq | offset...offset | begin...begin | end...end | begin... |
| n_docs[1] | doc...doc | freq...freq | offset...offset | begin...begin | end...end | begin... |
| ... | | | | | | |

Описание полей

| Название | Тип | Назначение |
|---|-------|---|
| n_terms | uint | Число терминов в корпусе/ число списков словопозиций и координат |
| n_docs[0],..., n_docs [n_terms -1] | uint | Число словопозиций для конкретного термина |
| doc[i][0],..., doc[i][n_docs[i]-1], i = 0...n_terms-1 | *int | Вектор идентификаторов документов, в которых встречается термин (слопозиции) |
| freq[i][0],..., freq [i][n_docs[i]-1], i = 0...n_terms-1 | *int | Вектор частот вхождений термина в документы |
| offset[i][0],..., offset[i][n_docs[i]-1], i = 0...n_terms-1 | *uint | Относительные смещения до координат. Таким образом, если понадобятся координаты i-терма в j-м документе, то сначала выполнится смещение до нужной строки в таблице <i>posting_list.data</i> с помощью смещений в словаре. Затем, зная число документов n_docs[i], можно быстро считать freq и offset, не читая остальные данные. Далее, с помощью offset выполняется смещение до блока, в котором находятся координаты begin...begin, end...end термина в документе. Их количество равно значению freq. |

| | | |
|----------------|------|---|
| begin... begin | *int | Координаты начал термина в документе. Координаты измеряются в символах от начала документа |
| end...end | *int | Координаты концов термина в документе. Координаты измеряются в символах от начала документа |

- **tf.data** (939 Мб)

Файл служит для быстрого получения компонент документа, как вектора в пространстве терминов. Он нужен для быстрого ранжирования на основе косинуса между вектором запроса и вектором документа. См. подробности в ЛР по ранжированию.

Структура

| | | | | | | |
|---------|-------|-----|---------------|-------|-----|---------------|
| n_docs | | | | | | |
| n_terms | ti[0] | ... | ti[n_terms-1] | tw[0] | ... | tw[n_terms-1] |
| n_terms | ti[0] | ... | ti[n_terms-1] | tw[0] | ... | tw[n_terms-1] |
| ... | | | | | | |

Описание полей

| Название | Тип | Назначение |
|---|---------|--|
| n_docs | uint | Число документов в корпусе |
| n_terms[i] i=0...n_docs-1 | *uint | Число терминов в i-м документе |
| ti[0]... ti[n_terms-1], ti[0]... ti[n_terms-1], ... | *int | Вектор идентификаторов терминов |
| tw[0]... tw[n_terms-1], tw[0]... tw[n_terms-1], ... | *double | Вектор весов. Вес соответствует идентификатору |

Построение индекса для корпуса с учётом лемматизации терминов с помощью отечественной NLP-системы Natasha [1] занимает 4 часа при распараллеливании на 4 OMP-потока (больше не позволяет размер оперативной памяти) процессора Intel Core i7 9700K (3.6 GHz). Блочный

индекс (до слияния) доступен по ссылке
<https://cloud.mail.ru/public/F3Fe/eTEiUPHt6>.

Поисковая система. Виды запросов

Система поддерживает несколько видов запросов, более подробно рассмотренных в соответствующих лабораторных работах:

1. Булев поиск

Синтаксис поисковых запросов:

- Пробел или два амперсанда, «&&», соответствуют логической операции «И».
- Две вертикальных «палочки», «||» – логическая операция «ИЛИ»
- Восклицательный знак, «!» – логическая операция «НЕТ»
- Могут использоваться скобки.

Парсер поисковых запросов устойчив к переменному числу пробелов, максимально толерантен к введённому поисковому запросу.

Примеры запросов:

- [московский && авиационный && институт];
- [(красный || желтый) автомобиль];
- [руки !ноги].

2. Цитатный поиск

Синтаксис этого элемента следующий:

- [«что где когда»] – кавычки, включают режим цитатного поиска для терминов внутри кавычек. Этому запросу удовлетворяют документы, содержащие в себе все термины что, где и когда, причём они должны встретиться внутри документа ровно в этой последовательности, без каких-либо вкраплений других терминов.
- [«что где когда» / 5] – аналогично предыдущему пункту, но допускаются вкрапления других терминов так, чтобы расстояние от первого термина цитаты до последнего не превышало бы 5.

Новый элемент может комбинироваться с другими стандартными средствами булева поиска, например:

- [«что где когда» && друг]
- [«что где когда» || квн]
- [«что где когда» && !«хрустальная сова»]

3. Нечёткий поиск

Если запрос содержит в себе только термины через пробелы, то он трактуется как нечёткий запрос, т.е. допускается неполное соответствие документа терминам запроса и т.п. Примеры запросов:

- [роза цветок]
- [московский авиационный институт]

Если запрос содержит в себе операторы булева поиска, то запрос трактуется как булев, т.е. соответствие должно быть строгим, но порядок выдачи должен определён ранжированием TF-IDF. Например:

- [роза && цветок]
- [московский && авиационный && институт]

Поисковая система. Интерфейс

Реализовано десктоп-приложение для ОС семейства Windows.

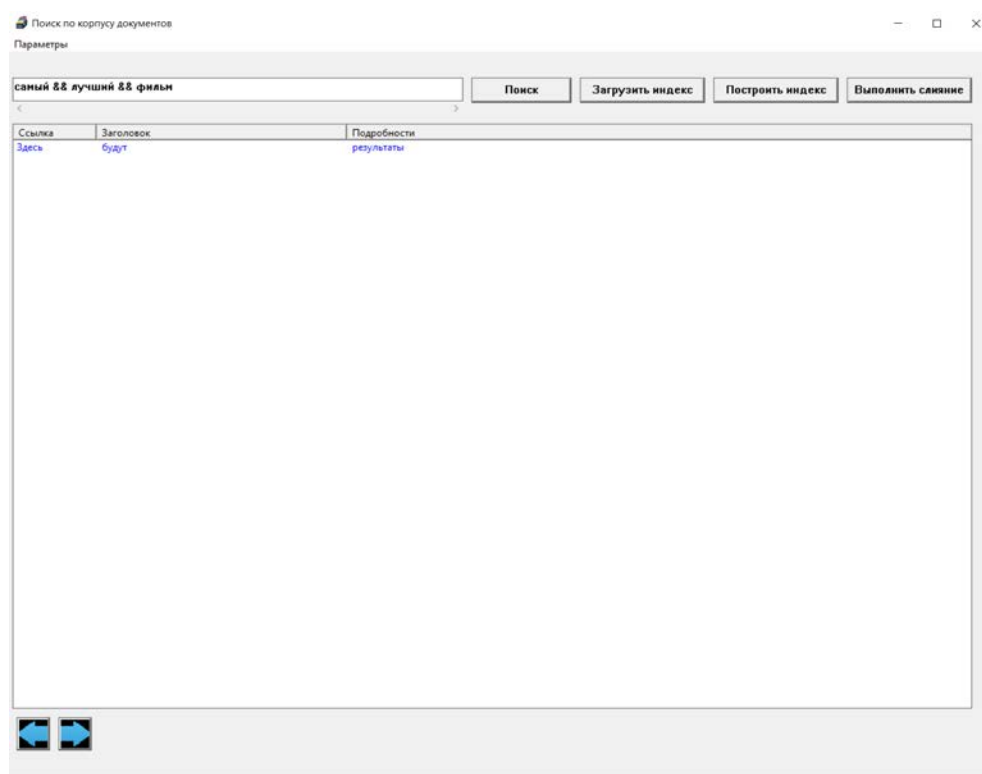


Рис. 1 – Стартовое графическое окно приложения

При запуске приложения пользователь видит два окна: графическое (с элементами управления) и консольное – для логирования.

```
D:\Мои документы\Курсовые работы и проекты\5 курс 1 семестр\Информационный поиск\Исправление опечаток\Release...
Консоль подключена!
Чтение термов : [#####] 1908410/1908410
[INFO] Загружено 1908410 термов
Создание линейного списка терминов : [#####] 1908410/1908410
[INFO] Загрузка завершена
Запрос: 'самый && лучший && фильм'
По запросу: 'самый & хороший & фильм' найдено 54490 документов
[INFO] Обработка результатов поиска
page_url: https://www.kinopoisk.ru/media/news/2528553/
title: "Самые яркие и безумные новости года"
Детали: ..528553/", "title": "Самые яркие и безумные но....говые материалы:\n\nлучшие фильмы 2014 года по версии..
page_url: https://www.kinopoisk.ru/media/news/1127648/
title: "С Новым Годом!"
Детали: ..кучных праздников и самых лучших подарков, позитивно....личество интересных фильмов и кинособытий, о ко..
page_url: https://www.kinopoisk.ru/media/news/230973/
title: "Кейси Аффлек убьет Брэда Питта"
Детали: ..Оппонентом Кейси по фильму станет Брэд Питт, э.... считается одним из лучших стрелков на Западе....меется, хо
чет стать самым лучшим, сместив с т..
page_url: http://filmplace.ru/film/bbc-zhivaya-priroda-rebyatam-o-zveryatah.html
title: "BBC: Живая природа. Ребятам о зверятах"
Детали: ..ательный сериал для самых маленьких зрителей. В этом фильме дети смогут не толь....но.Качайте", "Очень хороший
док сериал! Детям, ..
page_url: https://www.kinopoisk.ru/media/news/3098389/
title: "Читатели КиноПоиска и «ВКонтакте» подведут итоги 2017 года"
Детали: ..осование за главные фильмы, самых достойных актеров и....лено 15 номинаций: «Лучший фильм», «Лучший реж..
page_url: https://www.kinopoisk.ru/film/465042/
title: "Подожди, пожалуй (2002)"
Детали: ..3", "description": "Фильм о любви и смерти.",....ными невзгодами. Не самую лучшую анимацию (хотя отме..
page_url: https://www.kinopoisk.ru/media/news/3280840/
title: "Читатели КиноПоиска и «ВКонтакте» выберут лучшие фильмы за 15 лет"
```

Рис. 2 – Консольное окно приложения

При взаимодействии пользователя с системой последняя через консольное окно ведёт оповещение, чем она «занята» в данный момент. Прежде делать запрос к корпусу необходимо загрузить индекс с помощью кнопки «Загрузить индекс» или, если индекс не создан, то необходимо создать блочный индекс и затем выполнить слияние с помощью соответствующих кнопок. Корпус должен храниться в директории на диске в формате, описанном в информации о корпусе (нумерация файлов необязательна – названия - произвольные). При этом один файл с документами считается системой одним блоком, которые обрабатываются параллельно, так что к выбору размера файла нужно подходить разумно.

В меню «параметры» настраивается количество потоков и пути до нужных директорий. При закрытии и запуске приложения система запоминает все пути, количество потоков, а также последний сделанный пользователем запрос.

?

Параметры

×

Путь к директории с корпусом

и документы\Лабы и рефераты\5 курс 1 семестр\Информационный поиск\Корпус

<

>

Путь к директории с индексом

менты\Лабы и рефераты\5 курс 1 семестр\Информационный поиск\Корпус_index

<

>

Путь к директории с временными файлами

я и проекты\5 курс 1 семестр\Информационный поиск\Исправление опечаток\tp

<

>

Число потоков

5

▲

▼

Рис. 3 – Выбор параметров

После загрузки индекса, можно делать запросы в формах, описанных в предыдущем разделе. Результаты сортируются по косинусному правилу TF-IDF.

Поиск по корпусу документов

Параметры

— □ ×

[самый && лучший && фильм] || "что где когда" / 5

Поиск

Загрузить индекс

Построить индекс

Выполнить слайне

| Ссылка | Заголовок | Подробности |
|----------------------|---|---|
| https://www.ki... | «Оскар-2017»: Шорт-лист фильмов с лучшими ви... | ...ар-2017»: Шорт-лист ФИЛЬМОВ с ЛУЧШИМИ визуальными эффектами...тендентов значится САМЫЙ кассовый фильм 2016...тастик... |
| https://filmplace... | Непростые вопросы: Путешествие в страну шама... | «итальный», "Русский ФИЛЬМ", "description": "...яться и радоваться САМЫМ простым вещам... В ...ло. Я могу отличить ХОРОШЕИ... |
| https://www.ki... | 9 лучших трейлеров недели: Человек-невидимка, ... | ...2525/", "title": "9 ЛУЧШИХ трейлеров недели: Ч...анипат в 1761 году, ГДЕ сразились силы импе... сравнение считается САМЫМ кру... |
| https://www.ki... | «Август. Восьмого»: Репортаж со съемочной пл... | ...596/", "title": "10 ЛУЧШИХ трейлеров недели: Т...льно главные роли в ФИЛЬМЕ о противостоянии дв...мы предупредим вас, КОГД... |
| https://www.ki... | 10 лучших трейлеров недели: Темные воды, альп... | ...5337/", "title": "9 ЛУЧШИХ трейлеров недели: Х...лился на «Оскар». В ФИЛЬМЕ «Вне игры» есть все, ЧТО может привлечь внима...а... |
| https://www.ki... | 9 лучших трейлеров недели: Харли Квинн, нове... | ...1759/", "title": "8 ЛУЧШИХ трейлеров недели: А...емало драматически: ФИЛЬМОВ, но это первая карт...е генераторы, из-за ЧЕГО... |
| https://www.ki... | 8 лучших трейлеров недели: Атомные самураи, д... | ...ериале «Теин Пикс», ГДЕ он сыграл франкокана...ерифа Эрла МакГро в ФИЛЬМАХ «От заката до рассв... навсегда останется ЛУЧШ... |
| https://www.ki... | Ушел из жизни актер Майкл Паркс | «Драма», "Зарубежный ФИЛЬМ", "description": "...тонаселенный район, ГДЕ проживала девушка, ... на свои злодеяния. САМОЕ п... |
| https://filmplace... | 37 | ...s", "title": "Высы: ФИЛЬМ / Air Movie", "comm...ни... Единственное ЧТО не понравилось в фи...у — что это одна из ЛУЧШИХ полнс... |
| https://shikimo... | Высы: Фильм / Air Movie | ...297/", "title": "10 ЛУЧШИХ трейлеров недели: Д...я весной 1917 года, КОГДА от действий двух со...дупредим вас, когда ФИЛЬМ стан... |
| https://www.ki... | 10 лучших трейлеров недели: Джентльмены, хищ... | ...но и на Западе. В ФИЛЬМЕ повествуется об обы...й можно делать все, ЧТО угодно, а потом вык... меня ностальгию))) ХОРОШИЙ |
| https://filmplace... | Приключения пингвиненка Лоло | ...е Rotten Tomatoes у ФИЛЬМА лишь 17% «свежести»...скусства, созданное ЛУЧШИМИ людьми в мире. Я пр...ожет стать одним из... |
| https://www.ki... | Звезда «Кошек» Джейсон Дрүроул ответил критика... | ...первой ленты серии, ГДЕ за главными героями...нстр. Кто сыграет в ФИЛЬМЕ и КОГДА ждать его выхода, п...ужой» стал одним из... |
| https://www.ki... | «Чужого» подвергнут перезагрузке | ...и пяти номинациям на ЛУЧШИЙ ФИЛЬМ", "body": "Десять фи...миллиона зрителей, ЧТО на 16 % меньше, чем...церемонии оказал... |
| https://filmplace... | Киноакадемия может вернуться к пяти номинаци... | ...ючения", "Советский ФИЛЬМ", "description": "...ебыванием в местах, ГДЕ живут люди, создают...а это делает... Ни ЧЕГО сложной... |
| https://www.ki... | Новые приключения Дони и Микки | ...0403/", "title": "7 ЛУЧШИХ трейлеров недели: ...ый бульдозер, после ЧЕГО поехал на нем верши...гинцева на создание ФИЛЬМА... |
| https://www.ki... | 7 лучших трейлеров недели: «Правосудие Спенсе... | ...6120/", "title": "9 ЛУЧШИХ трейлеров недели: ...нсы в первых восьми ФИЛЬМАХ режиссера. Создатель...ом и стал одним из САМ... |
| https://www.ki... | 9 лучших трейлеров недели: «Звездные войны», К... | ...5596/", "title": "9 ЛУЧШИХ трейлеров недели: Х...снимают очень много ФИЛЬМОВ ужасов. Новый хорро...мы предупредим вас, К... |
| https://filmplace... | Холодное сердце, Ир... | ...ы 3 сезон сняли.", "ХОРОШИЙ и интересный сериал...не сериал зашел, не ЧЕГО особенного, дешево...е всеми остальными, СА... |
| https://www.ki... | Рagnarok | ...моги: стал одним из САМЫХ ожидаемых ФИЛЬМОВ года, он может заяв...е один фильм, после ЧЕГО хоч...отойти от биз...смляе... |
| https://www.ki... | Режиссер «Новолуния» хочет уйти из кино | ...0-х Ривер с помощью ФИЛЬМА «Бешеный бык» заста...ачнешь играть — вот ЧТО ты начнешь делать", ...с считался одним из СА... |
| https://www.ki... | Хоакин Феникс поблагодарил брата Ривера за сво... | ...ую ветвь», одну из САМЫХ престижных кинопрем...ера Пака. Произведа ХОРОШЕЕ впечатление. Ки-у...мы предупредим вас, Н... |
| https://www.ki... | Премьера на КиноПоиске: Трейлер «Паразитов» | ...ть и ждать", "что ЛУЧШЕ ждать пока те сериа...ностью или смотреть ФИЛЬМЫ?@KiritoAsuna, @Roma...йдет, чем дожидешься КОГ... |
| https://shikimo... | Chain Chronicle: Haecceitas no Hikari Part 2 | «Драма», "Зарубежный ФИЛЬМ", "Комедия", "Мелод...т, что в комплексе, ГДЕ он живет, вселилась...мормоны - одна из САМЫХ с... |
| https://filmplace... | Порыв ветра | «етраженный», "Русский ФИЛЬМ", "description": "...ого человека. День, КОГДА все против него. Де...талантливо. 6/10", "ХОРОШАЯ д... |
| https://www.ki... | Что? Где? Когда? | ...656730/", "title": "ЧТО? ГДЕ? КОГДА?", "alternative_tit... |
| https://filmplace... | Лиса и Заяц | ...и этот мультфильм!", "САМОЕ ЛУЧШЕЕ для меня в этом мульт...зверушек почти весь ФИЛЬМ показывается в анфа... думал, что ег... |
| https://www.ki... | 5 лучших трейлеров недели: «Харли Квинн», «Хок... | ...0958/", "title": "5 ЛУЧШИХ трейлеров недели: «...винн, «Хокусай» и «ЧЕМ мы заняты в тени»...мы предупредим вас, КОГДА прои... |
| https://www.ki... | Кирилл Серебренников напишет и поставит мини... | ...арковский — один из САМЫХ прославленных совет...ких режиссеров, чьи ФИЛЬМЫ хорошо известны дал...включаются в список... |
| https://www.ki... | Пять лет и один день (2012) | ...аре, ее муж живет с ЛУЧШЕЙ подружкой, которую б...рхжи, единственное, ЧТО есть у женщины — же... этот замечательный ФИЛЬ... |
| https://filmplace... | Помадные джунгли | ...«После долгого дня САМОЕ тоТакой лёгкий!», ...лицу. Но, по моему, ЛУЧШЕ жить плодотворной, ...му за 30!", "Искала ФИЛЬМ г... |
| https://www.ki... | День, когда я нравился девушкой (2006) | ...т", "title": "День, КОГДА я нравился девушкой...мною, считаются его ЛУЧШЕЙ работой. А конкретн... больше понравился СА... |
| https://filmplace... | Генезис 2.0 | «Зарубежный ФИЛЬМ", "description": "...ые можно продать за ХОРОШИЕ деньги как САМЫЕ кассовые фильмы 201...аторию Южной Кореи, Г... |
| https://www.ki... | «Оскар-2018»: Лонг-лист фильмов с лучшими виз... | ...ар-2018»: Лонг-лист ФИЛЬМОВ с ЛУЧШИМИ визуальными эффектами...листе оказались как САМЫЕ кассовые фильмы 201...извест... |
| https://filmplace... | Легенда о царе Соломоне | ...ми решениями даже в САМЫХ сложных ситуациях...сином. Анимационный ФИЛЬМ «Легенда о царе Сол...менный зритель см... |
| https://www.ki... | Проклятые фотографии (2006) | ...ews": ["Фотографии\ФИЛЬМ мне попался соверше... случайно, я искала ХОРОШИЙ фильм ужасов и наткнулась на этот КОГДА... |
| https://www.ki... | Министр культуры не исключил раздвигек кино... | ...а релиза зарубежных ФИЛЬМОВ, выход которых запл...ики — время премьер ЛУЧШИХ российских фильмов...и декабристов. Ж... |
| https://www.ki... | Нарушение общественного порядка | ...lres": ["Зарубежный ФИЛЬМ", "Комедия", "desc...прожил в Австралии, ГДЕ работал обычным про... встречался с тремя ЛУЧШИ... |
| https://www.ki... | Интерстеллар: Премьера финального дублירו... | ...ь ленту. В итоге за ФИЛЬМ взялся Кристофер Но...нтерстеллар» станет САМЫМ длинным фильмом Кри...акрытом показе, это Л... |
| https://filmplace... | Проклятие куклы Роберт | ...lres": ["Зарубежный ФИЛЬМ", "Ужасы", "desc...на работу в музей, ГДЕ много странных эксп... именем Роберт. Все САМОЕ не... |

1/1091

←

→

Рис. 4 – Поисковая выдача по комбинированному запросу (графическое окно)

13


```
D:\Мои документы\Курсовые работы и проекты\5 курс 1 семестр\Информационный поиск\Исправление опечаток\64\Release...
все всех самых важных новостей из ..
Запрос: ' (самый && лучший && фильм) | "что где когда" / 5 '
По запросу: ' (самый & хороший & фильм) | "что где когда" / 5 ' найдено 54525 документов
[INFO] Обработка результатов поиска
page_url: https://www.kinopoisk.ru/media/news/2866427/
title: "«Оскар-2017»: Шорт-лист фильмов с лучшими визуальными эффектами"
Детали: ..ар-2017»: Шорт-лист фильмов с лучшими визуальными эффектами...етендентов значатся самый кассовый фильм 2016....
статистические твари и где они обитают).\n\nЧл....известен 24 января, когда будут обнародованы ..
page_url: http://filmpolice.ru/film/neprostye-voprosy-puteshestvie-v-stranu-shamanov-altaj.html
title: "Непростые вопросы: Путешествие в страну шаманов - Алтай"
Детали: ..нтальный", "Русский фильм"], "description": "...яться и радоваться самым простым вещам... В ....ло. Я могу о
тличить хорошее от плохого, я вижу....я ее только тогда, когда мне это действительно...плохой человек. А что же не та
к? Почему я....ьше? ..... Кто я? Где мой дом? Со всеми э..
page_url: https://www.kinopoisk.ru/media/news/3432525/
title: "9 лучших трейлеров недели: Человек-невидимка, Ип Ман и душа Pixar"
Детали: ..2525/", "title": "9 лучших трейлеров недели: Ч...анипат в 1761 году, где сразились силы импе.... сражение счи
тается самым крупным в XVIII век....мы предупредим вас, когда проект станет досту....риях. Одна из них – фильм о мире ду
ш, где зар....о-то инопланетного, что не поддается объясн..
page_url: https://www.kinopoisk.ru/media/article/1592949/
title: "«Август. Восьмого»: Репортаж со съемочной площадки"
Детали: .. Она сама не знает, чего ей хочется, поэтому....ильон «Мосфильма», где режиссер Джаник Фай.... создает свой н
овый фильм «Август. Восьмого»....ой, и стал одним из самых кассовых российских....).\n\nВ тот день, когда КиноПоиск п
риехал в....ость.\n\nВпрочем, лучше посмотрите все сами..
page_url: https://www.kinopoisk.ru/media/news/3413596/
title: "10 лучших трейлеров недели: Темные воды, альпинисты и хранители"
Детали: ..596/", "title": "10 лучших трейлеров недели: Т...льно главные роли в фильме о противостоянии дв....мы предупр
едем вас, когда фильм выйдет в кино....аются заработать на самых известных произведе....развивается в поле, где происход
ит что-то с..
page_url: https://www.kinopoisk.ru/media/news/3435337/
title: "9 лучших трейлеров недели: Харли Квинн, новенький Соник и губка в бегах"
```

Рис. 5 – Поисковая выдача по комбинированному запросу (консольное окно)

По результатам запроса пользователь открывает ссылку в первом столбце в любом, привычном ему, браузере.

Также существует полностью консольная версия приложения, существующая в т.ч. для тестирования и измерения метрик поисковой системы. Ей на вход в качестве аргументов программы подаются на вход все вышеуказанные параметры, а также путь к данным для просчёта метрик (см. подробности в соответствующей ЛР).

```
D:\Мои документы\Курсовые работы и проекты\5 курс 1 семестр\Информационный поиск\Исправление опечаток\64\Release\Исправление о...
По запросу: ' " тихий место 2 " ' найдено 50 документов
Запрос: ' аватар скачать '
По запросу: ' !! аватар скачать ' найдено 474 документов
Запрос: ' "re zero" / 4 '
По запросу: ' " re zero " / 4 ' найдено 266 документов
Запрос: ' режиссер назад в будущее '
По запросу: ' !! режиссер назад в будущее ' найдено 6233 документов
Запрос: ' фильм для интеллектуалов '
По запросу: ' !! фильм для интеллектуал ' найдено 1286 документов
Запрос: ' фильм сериал с самым большим рейтингом '
По запросу: ' !! фильм сериал с самым большим рейтингом ' найдено 4820 документов
Запрос: ' фильмы Макото Синкай '
По запросу: ' !! фильм макото синкай ' найдено 271 документов
Запрос: ' лучшие фильмы квентина тарантино '
По запросу: ' !! хороший фильм квентин тарантино ' найдено 1266 документов
Запрос: ' как звали главного героя коносуэ '
По запросу: ' !! как звать главный герой коносуэ ' найдено 85 документов
Запрос: ' самый лучший фильм '
По запросу: ' !! самый лучший фильм ' найдено 54490 документов
Запрос: ' сериалы с рейтингом 18+ '
По запросу: ' !! сериал с рейтинг 18 ' найдено 2061 документов
Запрос: ' высшая школа демонов '
По запросу: ' !! высший школа демон ' найдено 1478 документов
Запрос: ' джокер '
По запросу: ' джокер ' найдено 1353 документов
Точность на уровне 5 = 0.400000
DCG на уровне 5 = 1.289830
nDCG на уровне 5 = 0.437459
ERR на уровне 5 = 0.717949
```

Рис. 6 Утилита тестирования системы

Целью курсового проекта является улучшение существующей системы путём добавление проверки орфографии.

2. Реализация проверки орфографии

Существует [3] две формы исправления ошибок: *исправление изолированного термина* (isolated-term correction) и *исправление с учётом контекста* (context-sensitive correction). При исправлении изолированного термина термины запроса исправляются по отдельности. Пример со словом **carot** иллюстрирует этот подход. Этот метод не позволяет, например, обнаружить, что запрос **flew form Heathrow** содержит искаженный термин **from**, поскольку каждый из терминов запроса по отдельности записан правильно.

По проблеме исправления изолированного термина существует два метода решения задачи: расстояние редактирования и перекрытие k -грамм.

Рассмотрим более подробно методы, основанные на расстоянии редактирования.

Расстояние Левенштейна

Расстояние редактирования между двумя строками символов s_1 и s_2 — это минимальное количество операций редактирования (edit operations), с помощью которых строку s_1 можно трансформировать в строку s_2 . Операции редактирования, позволяющие это сделать, включают в себя следующие преобразования:

- 1) вставка символа в строку,
- 2) удаление символа из строки,
- 3) замена символа в строке другим символом.

При указанном наборе операций редактирования расстояние редактирования называется расстоянием Левенштейна (Levenshtein distance). Например, расстояние редактирования между словами **cat** и **dog** равно трем. Расстояние редактирования можно обобщить, если разным операциям редактирования присвоить разные веса. Например, операции замены символа **s** символом **p** можно присвоить более высокий вес, чем операции замены символа **s** символом **a** (так как буква **a** ближе к букве **s** на клавиатуре). Присвоение весов с учетом правдоподобности замены на практике оказалось очень эффективным [3]. Однако в нашем изложении мы будем придерживаться допущения, что все операции редактирования имеют один и тот же вес.

Хорошо известно, как вычислить расстояние редактирования между двумя строками за время $O(|s_1| \times |s_2|)$, где $|s_i|$ — длина строки s_i . Для этого используется алгоритм динамического программирования, представленный на рис. 7, где строки s_1 и s_2 представлены в виде массивов. Этот алгоритм заполняет все (целочисленные) ячейки матрицы m , размеры которой равны длинам соответствующих строк; после выполнения алгоритма элемент (i, j) содержит расстояние редактирования между строками, состоящими из первых i символов строки s_1 и первых j символов строки s_2 . Основной шаг динамического программирования отображен в строках 8–10 (рис. 7), в которых вычисляется минимум трех величин с учетом замены символа в строке s_1 , вставки символа в строку s_1 и вставки символа в строку s_2 .

`EditDistance(s_1, s_2)`

```

1  int  $m[|s_1|, |s_2|] = 0$ 
2  for  $i \leftarrow 1$  to  $|s_1|$ 
3  do  $m[i, 0] = i$ 
4  for  $j \leftarrow 1$  to  $|s_2|$ 
5  do  $m[0, j] = j$ 
6  for  $i \leftarrow 1$  to  $|s_1|$ 
7  do for  $j \leftarrow 1$  to  $|s_2|$ 
8      do  $m[i, j] = \min\{m[i-1, j-1] + \text{if}(s_1[i] = s_2[j]) \text{ then } 0 \text{ else } 1, \text{fi},$ 
9           $m[i-1, j] + 1,$ 
10          $m[i, j-1] + 1\}$ 
11 return  $m[|s_1|, |s_2|]$ 
```

Рис. 7 – Алгоритм вычисления расстояния Левенштейна без учёта весов

Реализация алгоритма на C++, представлена алгоритмом 1.

Алгоритм 1 – реализация вычисления расстояния Левенштейна на C++

```
double LevenshteinDistance(const wstring &s1, const wstring &s2, vector<double> &work)
{
    int m = s1.size() + 1,
        n = s2.size() + 1;

    work.resize(m * n);
    vector<double> &M = work;

#define ij2i(i, j) ((i) * n + (j))

    M[ij2i(s1.size(), s2.size())] = 0;

    int i, j;

    for (i = 1; i <= s1.size(); i++)
    {
        M[ij2i(i, 0)] = i;
    }

    for (j = 1; j <= s2.size(); j++)
    {
        M[ij2i(0, j)] = j;
    }

    for (i = 1; i <= s1.size(); i++)
    {
        for (j = 1; j <= s2.size(); j++)
        {
            M[ij2i(i, j)] = min3(M[ij2i(i-1, j-1)] + (s1[i - 1] == s2[j - 1] ? 0 : 1),
                                   M[ij2i(i - 1, j)] + 1,
                                   M[ij2i(i, j - 1)] + 1);
        }
    }

    return M[ij2i(s1.size(), s2.size())];

#undef ij2i
}
```

Вектор `work` – рабочее пространство, позволяет экономить вызовы `malloc`'а или память в стеке. Матрица хранится по строкам в линейаризованном массиве (с доступом по макросу `ij2i(i, j)`) для ускорения доступа путём попадания крайних элементов в одну кеш-линию.

Расстоянии Дамерау-Левенштейна. Некорректный алгоритм

Расстояние Дамерау-Левенштейна (Damerau-Levenshtein distance) между двумя строками, состоящими из конечного числа символов — это минимальное число операций вставки, удаления, замены одного символа и транспозиции двух соседних символов, необходимых для перевода одной строки в другую. Является модификацией расстояния Левенштейна, отличается от него добавлением операции перестановки. Дамерау показал, что 80% человеческих ошибок при наборе текстов составляют перестановки соседних символов, пропуск символа, добавление нового символа, и ошибка в символе [2], поэтому имеет смысл использовать именно эту метрику.

Существует несколько алгоритмов реализации вычисления этого расстояния: упрощённый и корректный алгоритмы.

Упрощённый алгоритм не решает задачу корректно, но бывает полезен на практике. Здесь и далее будем использовать следующие обозначения: S и T — строки, между которыми требуется найти расстояние Дамерау-Левенштейна; M и N — их длины соответственно.

Рассмотрим алгоритм, отличающийся от алгоритма поиска расстояния Левенштейна одной проверкой (храним матрицу D , где $D(i, j)$ — расстояние между префиксами строк: первыми i символами строки S и первыми j символами строки T). Рекуррентное соотношение имеет вид:

Ответ на задачу — $D(M, N)$, где

$$D(i, j) = \begin{cases} \min(A, D(i-2, j-2) + tCost), & \text{если } i > 1, j > 1, S[i] = T[j-1], S[i-1] = T[j] \\ A, & \text{иначе} \end{cases}$$

$$A = \begin{cases} 0, & \text{если } i = 0, j = 0 \\ i \cdot dCost, & \text{если } j = 0, i > 0 \\ j \cdot iCost, & \text{если } i = 0, j > 0 \\ D(i-1, j-1), & \text{если } S[i] = T[j] \\ \min(\\ \quad D(i, j-1) + iCost \\ \quad D(i-1, j) + dCost \\ \quad D(i-1, j-1) + rCost \\ \quad), & \text{если } j > 0, i > 0, S[i] \neq T[j] \end{cases}$$

$dCost$ — стоимость удаления символа,

$iCost$ — стоимость вставки,

$rCost$ – стоимость удаления,

$tCost$ – стоимость перестановки символов.

Таким образом для получения ответа необходимо заполнить матрицу D , пользуясь рекуррентным соотношением. Сложность алгоритма: $O(M \cdot N)$. Затраты памяти: $O(M \cdot N)$.

Контрпример: $S = 'CA'$ и $T = 'ABC'$. Расстояние Дамерау-Левенштейна между строками равно 2 ($CA \rightarrow AC \rightarrow ABC$), однако функция приведённая выше возвратит 3. Дело в том, что использование этого упрощённого алгоритма накладывает ограничение: любая подстрока может быть отредактирована не более одного раза. Поэтому переход $AC \rightarrow ABC$ невозможен, и последовательность действий такая: ($CA \rightarrow A \rightarrow AB \rightarrow ABC$). Упрощенный алгоритм Дамерау-Левенштейна не является метрикой, так как не выполняется правило треугольника: $DLD('CA', 'AC') + DLD('AC', 'ABC') \neq DLD('CA', 'ABC')$.

Несмотря на то, что условие многих практических задач в не предполагает многократного редактирования подстрок [2], в задаче исправления опечаток в запросах его недостаточно, ниже представлен более сложный алгоритм, который корректно решает задачу поиска расстояния Дамерау-Левенштейна.

Расстоянии Дамерау-Левенштейна. Корректный алгоритм

В основу алгоритма положена идея динамического программирования по префиксу. Будем хранить матрицу $D[0..M+1][0..N+1]$, где $D[i+1][j+1]$ — расстояние Дамерау-Левенштейна между префиксами строк S и T , длины префиксов — i и j соответственно. Для учёта транспозиции потребуются хранение следующей информации. Инвариант:

$lastPosition[x]$ — индекс последнего вхождения x в S ,

$last$ — на i -й итерации внешнего цикла индекс последнего символа $T: T[last] = S[i]$.

Тогда если на очередной итерации внутреннего цикла положить: $i' = lastPosition[T[j]], j' = last$, то

$D(i, j) = \min(A, D(i', j') + (i - i' - 1) \cdot dCost + tCost + (j - j' - 1) \cdot iCost)$,
где

$$A = \begin{cases} 0, & \text{если } i = 0, j = 0 \\ i \cdot dCost, & \text{если } j = 0, i > 0 \\ j \cdot iCost, & \text{если } i = 0, j > 0 \\ D(i-1, j-1), & \text{если } S[i] = T[j] \\ \min(& \\ \quad D(i, j-1) + iCost & \\ \quad D(i-1, j) + dCost & \text{если } j > 0, i > 0, S[i] \neq T[j] \\ \quad D(i-1, j-1) + rCost & \\ \quad), & \end{cases}$$

Сложность алгоритма: $O(M \cdot N \cdot \max(M, N))$. Затраты памяти: $O(M \cdot N)$. Однако скорость работы алгоритма может быть улучшена до $O(M \cdot N)$.

Псевдокод алгоритма изображён на рис. 8.

```
int DamerauLevenshteinDistance(S: char[1..M], T: char[1..N]; deleteCost, insertCost, replaceCost, transposeCost: int):
    // Обработка крайних случаев
    if (S == "")
        if (T == "")
            return 0
        else
            return N
    else if (T == "")
        return M
    D: int[0..M+1][0..N+1] // Динамика
    INF = (M + N) * max(deleteCost, insertCost, replaceCost, transposeCost) // Большая константа

    // База индукции
    D[0][0] = INF
    for i = 0 to M
        D[i+1][1] = i * deleteCost
        D[i+1][0] = INF
    for j = 0 to N
        D[1][j+1] = j * insertCost
        D[0][j+1] = INF

    lastPosition: int[0..количество различных символов в S и T]
    // для каждого элемента C алфавита задано значение lastPosition[C]

    foreach (char Letter in (S + T))
        lastPosition[Letter] = 0

    for i = 1 to M
        last = 0
        for j = 1 to N
            i' = lastPosition[T[j]]
            j' = last
            if S[i] == T[j]
                D[i+1][j+1] = D[i][j]
                last = j
            else
                D[i+1][j+1] = min(D[i][j] + replaceCost, D[i+1][j] + insertCost, D[i][j+1] + deleteCost)
                D[i+1][j+1] = min(D[i+1][j+1], D[i'][j'] + (i - i' - 1) * deleteCost + transposeCost + (j - j' - 1) * insertCost)
            lastPosition[S[i]] = i

    return D[M][N]
```

Рис. 8 – Корректный алгоритм вычисления расстояния Дамерау-Левенштейна

Реализация этого алгоритма на C++ представлена алгоритмом 2.

```
double DamerauLevenshteinDistance(const wstring &S, const wstring &T,
                                vector<double> &work, dict<wchar, int> iwork,
                                double deleteCost = 1,
                                double insertCost = 1,
                                double replaceCost = 1,
                                double transposeCost = 0.8)
{
    int M = S.size(),
        N = T.size();

    // Обработка крайних случаев
    if (S == L"")
        if (T == L"")
            return 0;
        else
            return N;
    else if (T == L"")
        return M;

    work.resize((M + 2) * (N + 2));
    double *D = work.data();

#define ij2i(i, j) ((i) * (N + 2) + (j))

    double INF = (M + N) * max4(deleteCost, insertCost, replaceCost, transposeCost);

    // База индукции
    D[ij2i(0, 0)] = INF;

    int i, j;
    for (i = 0; i <= M; i++)
    {
        D[ij2i(i + 1, 1)] = i * deleteCost;
        D[ij2i(i + 1, 0)] = INF;
        for (j = 0; j <= N; j++)
        {
            D[ij2i(1, j + 1)] = j * insertCost;
            D[ij2i(0, j + 1)] = INF;
        }
    }

    int diff = 0;
    dict<wchar, int> &lastPosition = iwork;
    lastPosition.clear();

    for (i = 0; i < S.size(); i++)
    {
        if (std::find(T.begin(), T.end(), S[i]) != T.end())
        {
            lastPosition[S[i]] = 0;
        }
    }

    int last,
        i_, j_;
    for (i = 1; i <= M; i++)
    {
```

```

last = 0;
for (j = 1; j <= N; j++)
{
    i_ = lastPosition[T[j - 1]];
    j_ = last;
    if (S[i - 1] == T[j - 1])
    {
        D[ij2i(i + 1, j + 1)] = D[ij2i(i, j)];
        last = j;
    }
    else
    {
        D[ij2i(i + 1, j + 1)] = min3(D[ij2i(i, j)] + replaceCost,
                                     D[ij2i(i + 1, j)] + insertCost,
                                     D[ij2i(i, j + 1)] + deleteCost);
    }
    D[ij2i(i + 1, j + 1)] = min(D[ij2i(i + 1, j + 1)],
                                D[ij2i(i_, j_)] +
                                (i - i_ - 1) * deleteCost +
                                transposeCost +
                                (j - j_ - 1) * insertCost);
    lastPosition[S[i - 1]] = i;
}

return D[ij2i(M, N)];

#undef ij2i
}

```

Здесь `work` – рабочее пространство, `iwork` – реализация словаря. Например `#define dict std::unordered_map`.

Окончательный алгоритм

Рассмотрим общую задачу. Дан запрос (не важно в какой форме), состоящий из множества терминов $T = \{T_0, \dots, T_m\}$ и множество терминов в корпусе $\{S_0, \dots, S_n\}$, для каждого термина определена его документная частота df . Требуется исправить возможные опечатки в терминах.

Предлагается решать эту задачу по алгоритму, представленному ниже.


```
def(T, S, Corpus) -> [множество изменённых термов T]
{
    // если найдено документов больше чем thresh1
    if len(search(T, Corpus)) >= thresh1
        return T
    for(i = 0; i < m; i++) // цикл по терминам из запроса
    {
        if(df(T[i]) < thresh2) // документная частота термина меньше порога
        {
            min_dist = INFINITY
            max_df = 0
            for(j = 0; j < n; j++) // цикл по всем терминам в словаре
            {
                dist = theta * DamerauLevenshteinDistance(T[i], S[j]) +
                    (1-theta) * (-log10(df(S[j]) / len(Corpus)))
                if(dist < min_dist)
                {
                    min_dist = dist
                    max_df = df(S[j])
                }
                else if(dist == min_dist)
                {
                    if(max_df < df(S[j]))
                    {
                        max_df = df(S[j])
                    }
                }
            }
        }
    }
}
```

```

        }
    }
}

T[i] = [термин с минимальным расстоянием min_dist и частотой
        max_df]

}

return T
}

```

В алгоритме 3 учтено не только расстояние до терма, но и его документная частота с помощью взвешенного соотношения:

$$dist(T, S) = \theta \cdot \text{DamerauLevenshteinDistance}(T, S) + (1 - \theta) \left(-\log_{10} \frac{df(S)}{|Corpus|} \right)$$

В практической реализации алгоритма 3 цикл по терминам в словаре можно распараллелить (реализовано) и/или применить эвристику того, что опечатка редко бывает в первом символе (таким образом, можно ограничиться только терминами, начинающимися на одну букву с термином запроса)

3. Исходный код

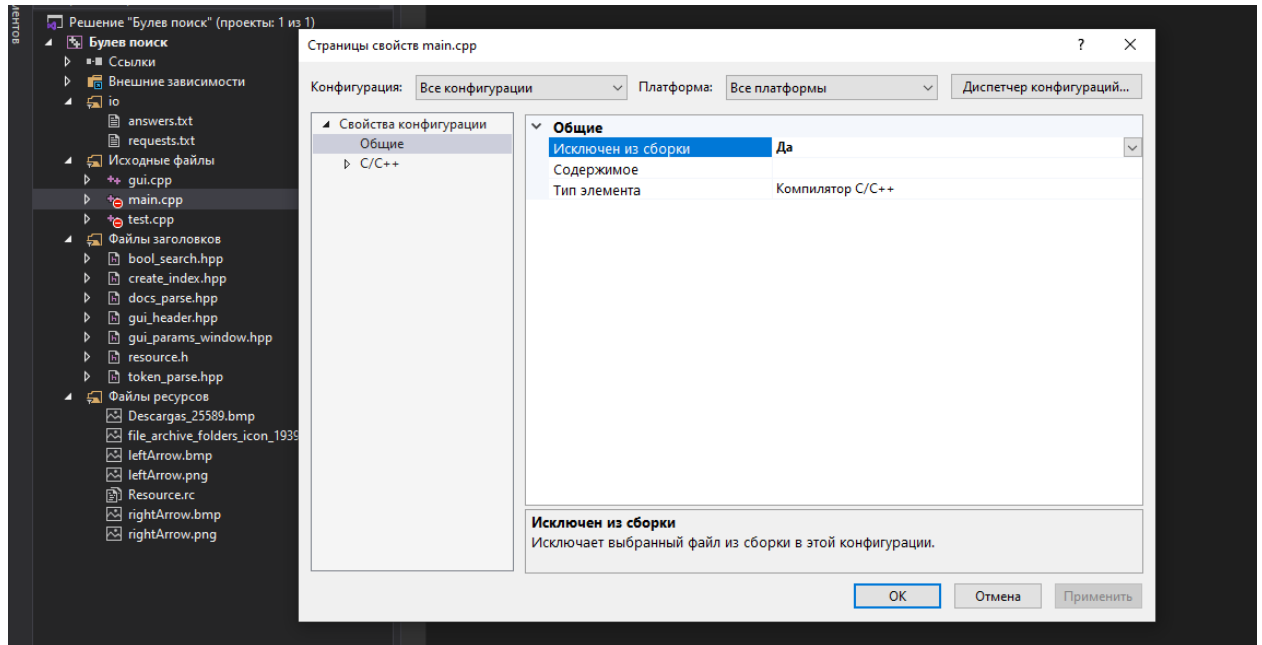
Структура проекта

- include
 - algebra.hpp (простейшие операции с векторами)
 - create_index.hpp (создание, чтение индекса)
 - defs.hpp (подключение внешних библиотек, макросы)
 - docs_parse.hpp (извлечение полей из корпуса)
 - gui_defs.hpp (подключение внешних библиотек, макросы, глобальные переменные)
 - gui_params_window.hpp (окно с выбором параметров)
 - resource.h (подключение изображений, иконок и прочего)
 - search.hpp (реализация всех видов поиска)
 - token_parse.hpp (функции для преобразования токенов в термы)
 - typos_correction.hpp (реализация исправления опечаток)
- python
 - lemmatizator.py (лемматизация документа)
 - lemmatizator_setup.py (компиляция lemmatizator.py в exe-файл)
 - request_parse.py (лемматизация запроса)
 - request_parse_setup.py (компиляция request_parse.py в exe-файл)
- io
 - answers.txt
 - requests.txt
- src
 - gui.cpp (точка входа в оконный интерфейс)
 - main.cpp (точка входа в консольный интерфейс тестирования программы)
- resources (файлы ресурсов для оконного приложения)

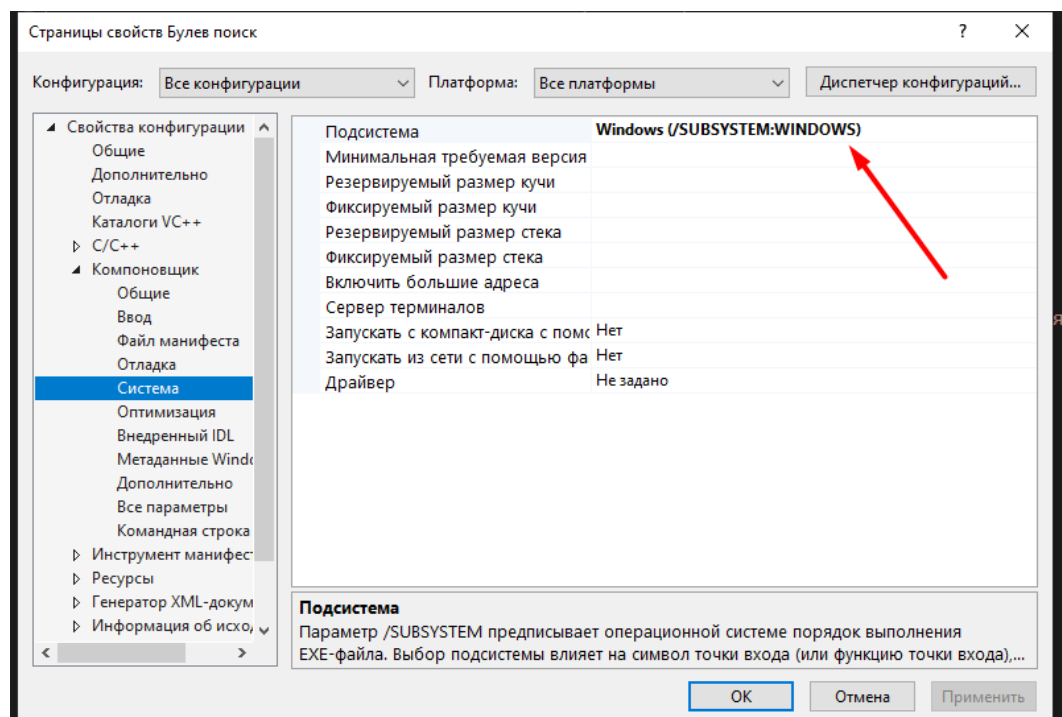
Проект был написан с помощью Microsoft Visual Studio 2019 эксклюзивно для ОС семейства Windows. Исходный код доступен по <https://github.com/Stifeev/Information-retrieval/tree/main/Курсовой%20проект>.

Запуск и сборка

Переключение между тремя точками входа осуществляется с помощью флага «исключить из сборки»:



Не забудь при переключении между консольными и оконными приложениями менять подсистему в настройке проекта:



Консольное приложение поддерживает флаги запуска:

- -i 'путь к корпусу'
- -o 'путь к индексу'
- -t 'путь к директории с блочным индексом'
- -m 'путь к директории с эталонами для метрик'
- -p кол-во_процессов_для_распараллеливания
- -create : создать блочный индекс
- -merge : выполнить слияние блочного индекса
- -clear : очистить папку с временными файлами после слияния
- -search : выполнить поиск
- -metric : высчитать метрики

Пример создания блочного индекса из корпуса:

```
$ ./prog.exe -p 4 -create -i "..\..\Корпус" -o -t "tmp"
```

Вывод

```
[INFO] Создание индекса для блоков
```

```
[INFO] Thread 0 processing block 1/13 : D:\Мои документы\Лабы и рефераты\5 курс 1 семестр\Информационный поиск\Корпус\films13.txt
```

```
[INFO] Thread 1 processing block 2/13 : D:\Мои документы\Лабы и рефераты\5 курс 1 семестр\Информационный поиск\Корпус\films9.txt
```

```
[INFO] Thread 2 processing block 3/13 : D:\Мои документы\Лабы и рефераты\5 курс 1 семестр\Информационный поиск\Корпус\films10.txt
```

```
[INFO] Thread 3 processing block 4/13 : D:\Мои документы\Лабы и рефераты\5 курс 1 семестр\Информационный поиск\Корпус\films12.txt
```

```
[INFO] Block 1 has 232216 terms
```

```
<...>
```

```
[INFO] Block 10 has 597482 terms
```

```
[INFO] Block 11 has 669497 terms
```

```
[INFO] Block 12 has 921883 terms
```

```
[INFO] Block 13 has 1383148 terms
```

```
[INFO] Создание очередей термов: 13 блок из 13
```

```
[INFO] Слияние docs_id: 13 блок из 13
```

[INFO] Слияние слопозиций термов

[INFO] Осталось термов: 0

[INFO] Очистка временных файлов

[INFO] Общее число термов в словаре = 2809203

Время выполнения = 145,5 sec, размер корпуса = 2,899 Gb, документов = 186109

Средняя скорость на документ = 0,782 ms

Средняя скорость на килобайт = 0,048 ms

Пример слияние блочного индекса:

```
$ ./prog.exe -p 4 -merge -clear -i "..\..\Корпус_index" -t "tmp"
```

Вывод

[INFO] Слияние блочного индекса

[INFO] Создание очередей термов: 13 блок из 13

[INFO] Слияние docs_id: 13 блок из 13

[INFO] Слияние слопозиций термов

[INFO] Осталось термов: 0

[INFO] Общее число термов в словаре = 1908410

Документов = 186109

[INFO] Время на слияние блочного индекса: 35 sec

[INFO] Вычисление статистики

Первый проход. Термов осталось: 0

Второй проход. Документов осталось: 0

[INFO] Вычисление статистики закончено

Пример просчёта метрик:

```
$ ./prog.exe -p 4 -metric -i "..\..\Корпус" -o "..\..\Корпус_index"
-m "..\..\Корпус_metric"
```

Вывод

Чтение термов : [#####] 1908410/1908410
[INFO] Загружено 1908410 термов
Запрос: ' "тихое место 2" '
По запросу: ' " тихий место 2 " ' найдено 50 документов
Запрос: ' аватар скачать '
По запросу: ' !! аватар скачать ' найдено 474 документов
Запрос: ' "re zero" / 4 '
По запросу: ' " re zero " / 4 ' найдено 266 документов
Запрос: ' режиссёр назад в будущее '
По запросу: ' !! режиссер назад в будущее ' найдено 6233 документов
Запрос: ' фильм для интеллектуалов '
По запросу: ' !! фильм для интеллектуал ' найдено 1286 документов
Запрос: ' фильм сериал с самым большим рейтингом '
По запросу: ' !! фильм сериал с самый больший рейтинг ' найдено 4820 документов
Запрос: ' фильмы Макото Синкая '
По запросу: ' !! фильм макото синкай ' найдено 271 документов
Запрос: ' лучшие фильмы квентина тарантино '
По запросу: ' !! хороший фильм квентин тарантино ' найдено 1266 документов
Запрос: ' как звали главного героя коносубы '
По запросу: ' !! как звать главный герой коносуб ' найдено 85 документов
Запрос: ' самый лучший фильм '
По запросу: ' !! самый хороший фильм ' найдено 54490 документов

Запрос: ' сериалы с рейтингом 18+ '

По запросу: ' !! сериал с рейтинг 18 ' найдено 2061 документов

Запрос: ' высшая школа демонов '

По запросу: ' !! высокий школа демон ' найдено 1478 документов

Запрос: ' джокер '

По запросу: ' !! джокер ' найдено 1353 документов

Точность на уровне 30 = 0.241026

DCG на уровне 30 = 2.655877

nDCG на уровне 30 = 0.289893

ERR на уровне 30 = 0.732372

4. Выводы и результаты

$\theta = 0.7$, остальные параметры алгоритма подобраны таким образом, чтобы происходило автоисправление ($thresh_1 = 30, thresh_2 = 30$).

| Запрос | Исправленный запрос |
|--------------------------|--------------------------|
| самый && лучший && фильм | самый && лучший && фильм |
| красавый | красивый |
| висна | видно |
| весно | весна |
| пвлапопвап | полуподвал |
| друзь | друг |
| пачему | почему |
| полупокер | покупка |
| gjdfkgjkdjgkdfjgdf | dfxgdfgdfgdfg |
| пекап | показ |
| юзернет | юзернейм |
| нормлвы | норма |

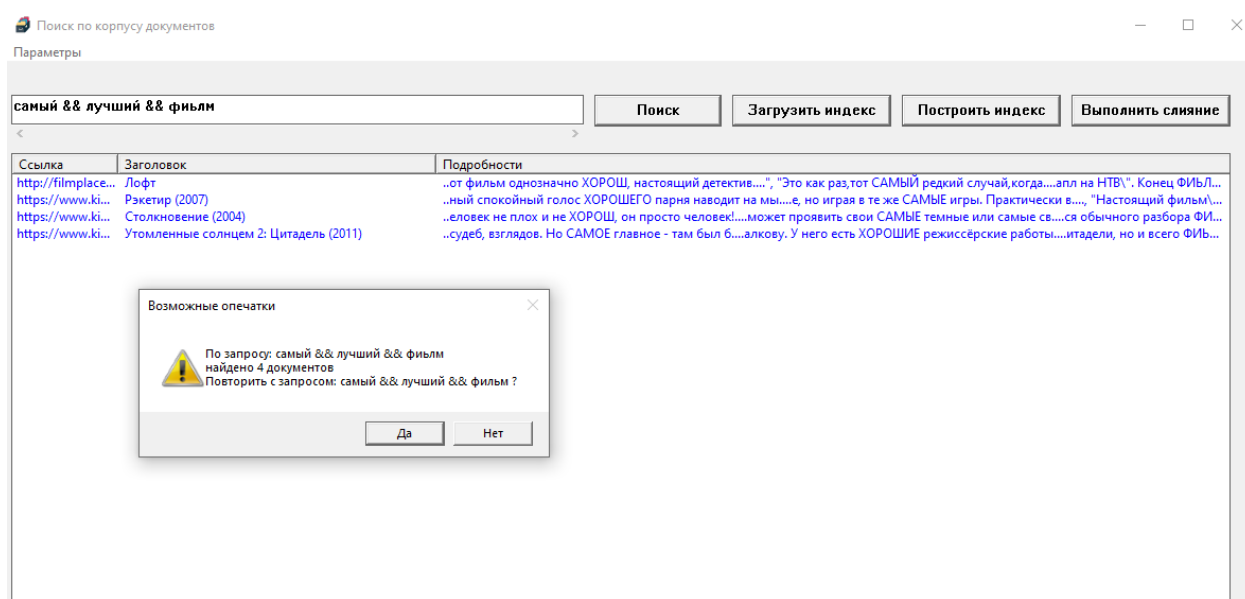


Рис. 9 – Реализация исправления опечаток в пользовательском интерфейсе (до исправления)

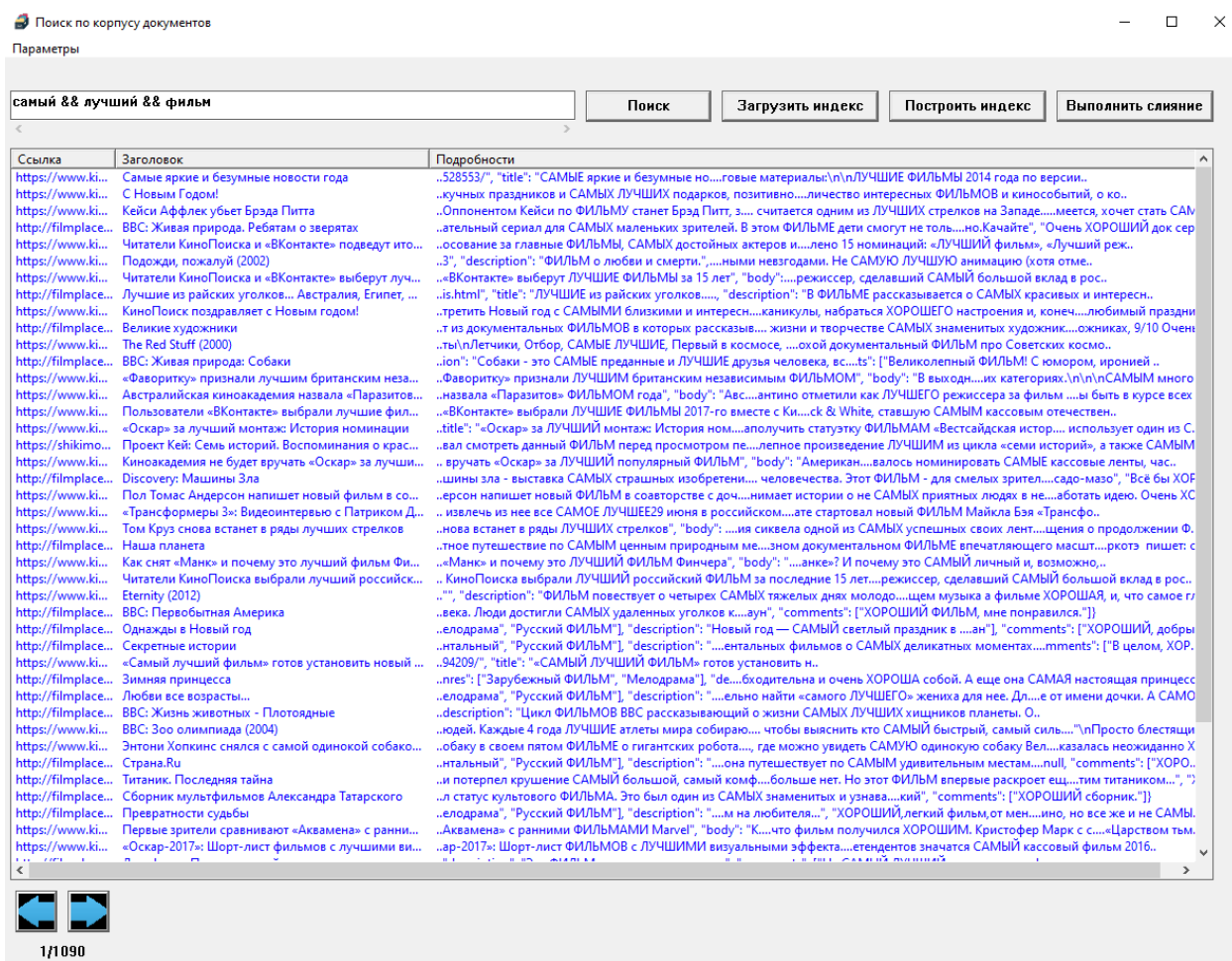


Рис. 10 Реализация исправления опечаток в пользовательском интерфейсе (после исправления)

По приведённым результатам можно сделать вывод, что автоисправление ошибок работает вполне корректно.

В ходе выполнения курсового проекта я научился выполнять автоисправление запросов к коллекции документов.

Литература

- [1] <https://natasha.github.io/>
- [2] https://neerc.ifmo.ru/wiki/index.php?title=Задача_о_расстоянии_Дамерау-Левенштейна
- [3] Кристофер Д.Маннинг, Прабхакар Рагхаван, Хайнрих Шютце. Введение в информационный поиск. 2020, изд. Вильямс.