1 Цель работы

Целью данной работы является изучение основных этапов построения простейшей поисковой системы. В рамках работы требуется сформировать корпус документов, подготовить его для последующей обработки, проанализировать статистические свойства текстов, а также реализовать индексирование и поиск по документам. Дополнительно необходимо рассмотреть примеры работы существующих поисковых систем и выявить их основные недостатки.

2 Описание данных

В качестве источника данных был выбран корпус статей из англоязычной версии Википедии, относящихся к тематике видеоигр. Данный источник предоставляет открытый программный интерфейс (API), позволяющий автоматически получать тексты статей и метаданные. Для формирования корпуса была выбрана корневая категория `Video games`, обход которой осуществлялся рекурсивно с ограничением глубины.

В результате работы парсера был сформирован корпус из **30 000 документов**. Каждый документ сохранён в отдельном текстовом файле и содержит основной текст статьи без служебной разметки. Для каждого документа также формируется файл метаданных, содержащий идентификатор документа, заголовок статьи, ссылку на источник и размер файла.

Средний размер текстов в корпусе составляет **7057**, медианный размер — **3989**. Минимальный и максимальный размеры документов составляют **201** и **308614** символов соответственно.

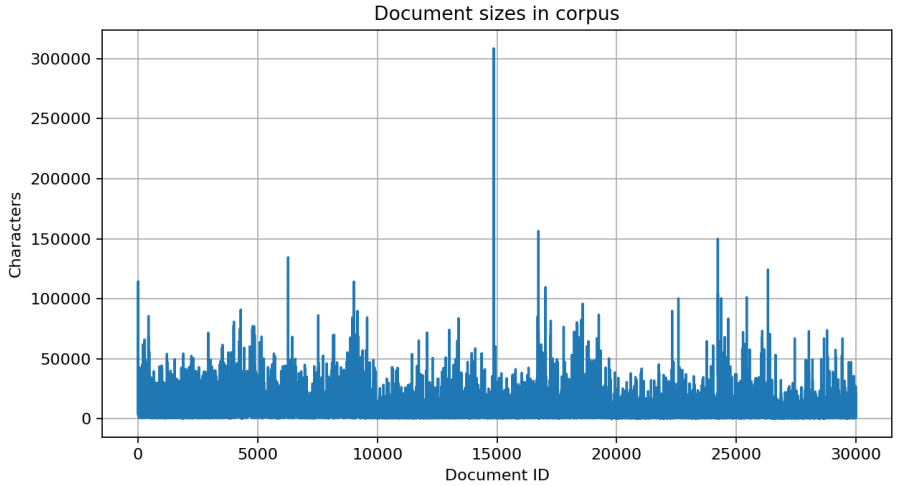

Рис. 1: количетсво символов в документах

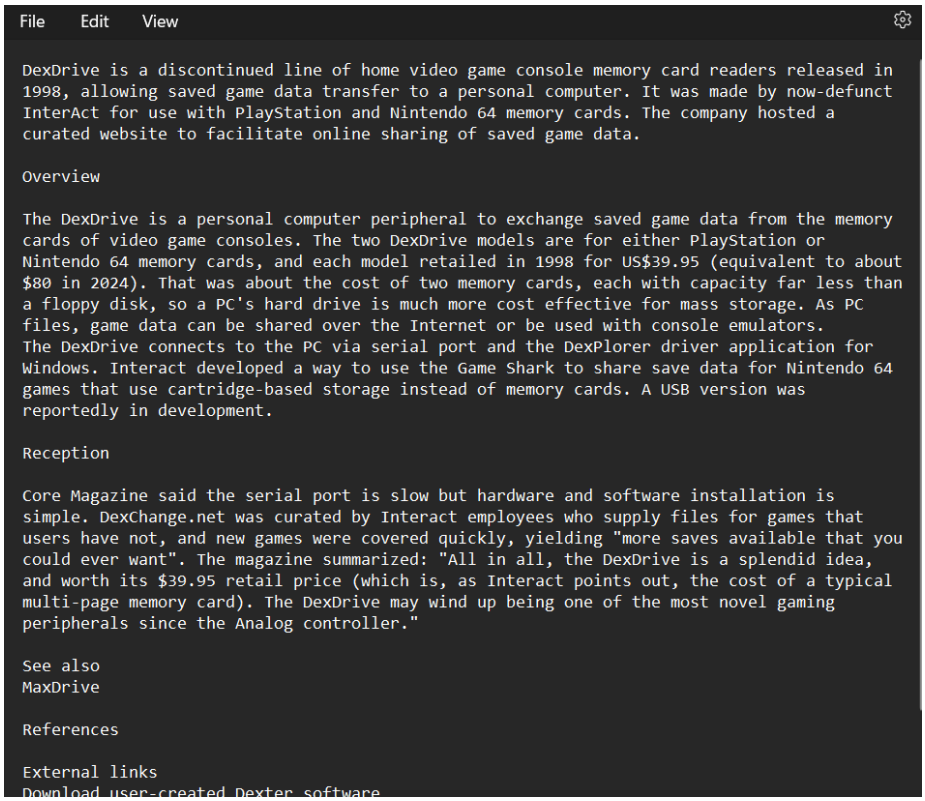

Рис. 2: пример текста

3 Закон Ципфа

Для анализа статистических свойств корпуса был рассмотрен закон Ципфа, описывающий распределение частот слов в естественных языках. Для этого на основе результатов токенизации и стемминга были подсчитаны частоты всех уникальных словоформ корпуса. Слова были отсортированы по убыванию частоты, после чего для каждого слова был определён его ранг.

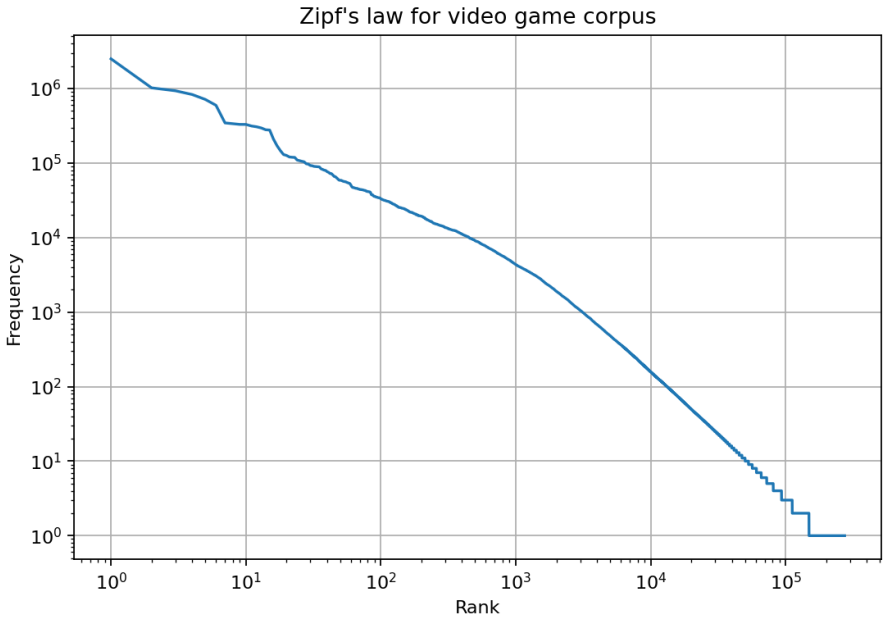

Рис. 3: закон Ципфа

rank	word	freq
1	the	2502562
2	and	1020405
3	of	932727
4	to	830274
5	in	713531
6	game	595295
7	wa	345416
8	for	336695
9	as	330005

Анализ графика показывает, что распределение слов по частотам близко к линейному в логарифмических координатах, что соответствует закону Ципфа. Отклонения наблюдаются для наиболее частотных слов и в хвосте распределения, что характерно для реальных текстовых корпусов.

4 Примеры существующих поисковых систем

Для анализа особенностей и ограничений современных поисковых систем были рассмотрены примеры поиска информации с использованием встроенного поиска Википедии и поисковой системы Google. В качестве запросов использовались термины, связанные с тематикой видеоигр.

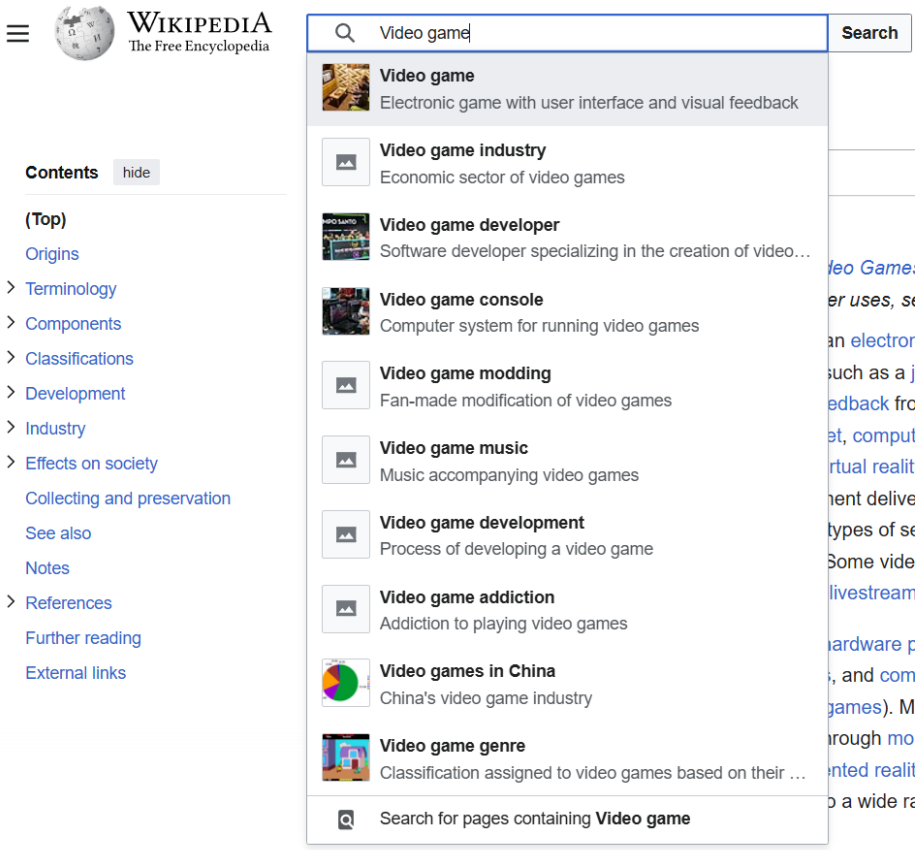

Рис. 4: поиск википедия

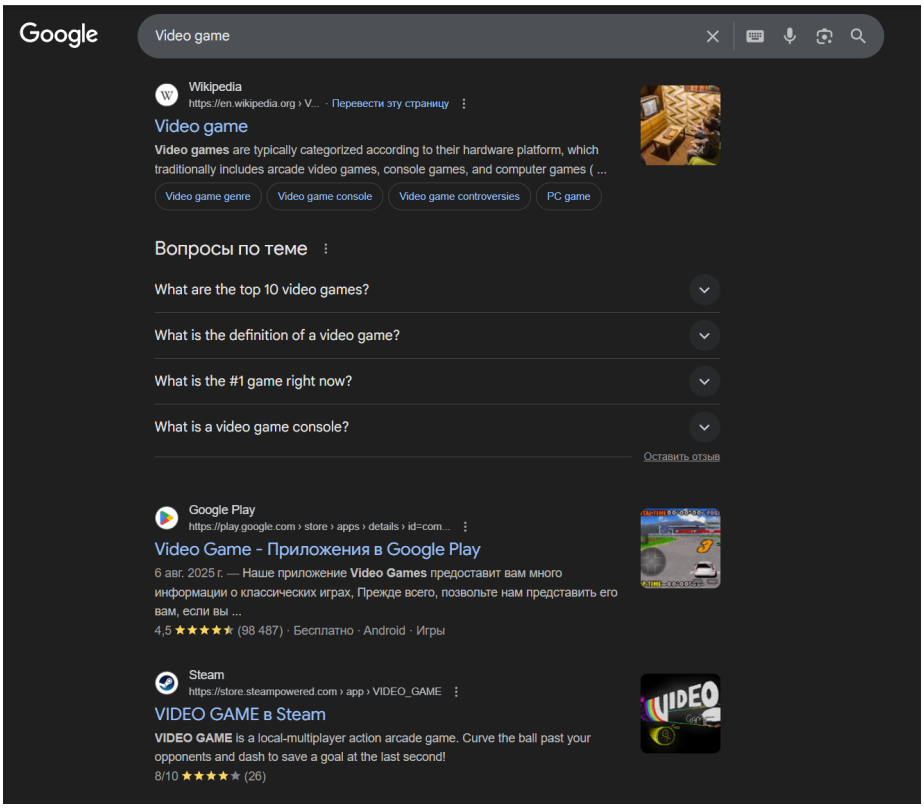

Рис. 5: поиск гугл

Анализ полученных результатов показывает, что существующие поисковые системы ориентированы на массового пользователя и популярные источники. При этом поиск по узкоспециализированным запросам часто требует дополнительного уточнения, а структура выдачи может содержать нерелевантные результаты или служебную информацию.

5 Индексация и поиск

Для обеспечения быстрого поиска по корпусу документов был реализован булев инвертированный индекс. Инвертированный индекс сопоставляет каждому терму список идентификаторов документов, в которых данный терм встречается. Такая структура позволяет эффективно выполнять операции логического поиска без полного перебора всех документов.

В процессе индексирования каждый документ обрабатывается независимо. Для каждого документа формируется набор уникальных термов, которые затем добавляются в соответствующие постинг-листы. Результаты индексирования сохраняются на диске в виде словаря терминов и бинарно-

го файла постинг-листов, что позволяет повторно использовать индекс без необходимости пересчёта.

```
> head index/dict.tsv
term    df      offset  len
0-0     8       0       32
0-0-7   1       32      4
0-00-713655-2   1       36      4
0-00-717558-2   1       40      4
0-00-720907-x   3       44      12
0-007-24622-6   2       56      8
0-02-935671-7   1       64      4
0-049-28039-2   1       68      4
0-06-083305-x   1       72      4

> wc -l index/dict.tsv
274105 index/dict.tsv

> ls -lh index/dict.tsv index/postings.bin index/maxdoc.txt
-rwxrwxrwx 1 user user 5.9M Dec 26 12:25 index/dict.tsv
-rwxrwxrwx 1 user user    6 Dec 26 12:25 index/maxdoc.txt
-rwxrwxrwx 1 user user  46M Dec 26 12:25 index/postings.bin

> cat index/maxdoc.txt
30000
```

6 Пример работы поисковой системы

Для демонстрации работы реализованной поисковой системы были выполнены несколько булевых запросов с использованием логических операторов `AND`, `OR` и `NOT`. В результате выполнения запросов система возвращает список идентификаторов документов, удовлетворяющих условиям запроса.

```
Loaded terms: 274104
Universe docs: 1..30000
Enter queries. Ctrl+D to exit.

nintendo
RESULTS 8905
1
...
30000
END

10-year AND NOT 10-year-old
RESULTS 46
```

```
3
...
29891
END

10-year-old OR 10-year
RESULTS 81
3
...
28337
29891
END

(10-minute OR 10-yard) AND 10-year
RESULTS 0
END
```

7 Статистика работы системы

Для оценки эффективности реализованной системы была измерена производительность основных этапов обработки данных. В частности, были зафиксированы время построения индекса и среднее время обработки одного поискового запроса.

```
    > /usr/bin/time -p ./build_index --stems stems --out index

Processed docs: 500, pairs: 337955
Processed docs: 1000, pairs: 565153
...
Processed docs: 30000, pairs: 11974986
Index built.
Docs processed: 30000
maxDoc: 30000
Output: index/dict.tsv, postings.bin, maxdoc.txt
real 39.78
user 8.13
sys 3.67
```

8 Заключение

В ходе выполнения данной работы были изучены основные принципы построения поисковых систем, включая сбор и подготовку корпуса документов, анализ статистических свойств текста, индексирование и реализацию булевого поиска. Реализованная система демонстрирует корректную рабо-

ту логических операций поиска и может быть расширена для поддержки более сложных методов ранжирования и анализа текстов.