Semester Short Report

ZHAW ~ School of Engineering

Intelligent Information Systems 1

Andras Marosi
(marosand)
marosand@students.zhaw.ch

05.12.2025

Contents

Introduction

This is a small experiment in the computer science field of information retrieval. The students were given 50 queries and 20’000 documents (consisting of mostly scientific documents about physics and physics related subject). The goal is to construct an interesting information retrieval problem and solution. Since there is no strictly defined criterium, with which the results could be judged, the students are free to be creative with their approaches and evaluations.

Methods

The idea behind this algorithm is to take advantage of the fact that in fields of science the content baring words tend to be longer than 3 (the linguists would say letters; the programmers would say characters) in order to cut out words that are not relevant to the search query (such as ”and”, ”or”, ”not”, etc.). There are exceptions of course (ax, ant, sun, key, box, etc.) but for the purposes of this experiment those are ignored.

The algorithm used in this report works using the following steps:

1. Both the document and the query are split up into words. A word is considered a word if it is separated by whitespace.
2. Words that are shorter than 3 are dropped from this list of words.
3. The word list can be sorted by the longest words first but doesn’t have to.

4. Each query is compared to each document in the following way:
    (a) Only words are compared that have relatively similar length. In this report the max absolute difference of 3 was taken.
    (b) For each of those word pairs the Levenshtein distance is calculated and added to a sum.
    (c) Those scores are stored in an associative array. Each document ID is associated with its score.
5. Finally, the 10 documents with the lowest scores are returned.

The source code for the project can be found here:
https://github.com/an-Iceberg/SSR

In order to not be an ambiguous verbal description of the algorithm here is a code snippet in Python showcasing the equivalent functionality:

```python
for query in queries
    similarity_scores = dict()
    for document in documents
        score = 0
        for query_word in query.longest_words()
            for doc_word in document.longest_words()
                if abs(len(query_word) - len(doc_word)) <= 3
                    score += levenshtein(query_word, doc_word)
        similarity_scores[document] = score
```

Results

... at first i could not believe my eyes.

The document returned by this algorithm are completely useless. They are very short, oftentimes shorter than the query itself. In the collection of documents a bunch of short documents (3-7 words) are included (presumably to test how well the search algorithm can deal with such unusual documents). Only those documents were returned by the above algorithm.

Conclusion

This experiment has been a colossal failure. This algorithm is compltetely useless.