Information Retrieval – Second Homework

Ondřej Měkota

January 4, 2020

Introduction

As a information retrieval framework I have chosen Whoosh [1]. It is written in Python and it provides a nice Python API. On the other hand, it is probably not as fast and well maintained as some popular frameworks such as ElasticSearch.

1 Run-0: baseline

I use the most basic setting of Whoosh for the baseline: only a `RegexTokenizer` (a regular expression based tokenizer). By default Whoosh does a boolean search (using AND between terms in query) this I changed to using OR between terms.

2 Run-1: constrained

The main thing in the constrained system is usage of BM25 algorithm with fine-tuned parameters B and K1. Apart from that, I use lemmatisation for Czech and stemming for English. For both languages, lowercasing, stopword removal has been utilized. Additionally for Czech, diacritics has been removed and words with length at most one have also been removed.

Stemming, lowercasing and stopword removal is pre-bundled in Whoosh as `StemmingAnalyzer` – used for English.

After many experiments with pseudo relevance feedback, I have not been able to achieve higher mean average precision than without it. I have tried extracting 5 to 30 *best words* from 5 to 20 highest scoring documents and

[1] `http://whoosh.readthedocs.io/`

constructing a query from them. Results from search with the new query either replaced the results of the original query or they were appended (without duplicities) at the end. All those methods decreased MAP (by about $0.02 - 0.05$).

3 Results

Results (MAP and P_10) are as follows:

Run	Czech	English
Run-0	0.0366	0.0764
Run-1	0.3342	0.4002

Table 1: Mean average precision of each run

Run	Czech	English
Run-0	0.0560	0.1040
Run-1	0.3360	0.4600

Table 2: Precision of the first 10 documents (P_10)

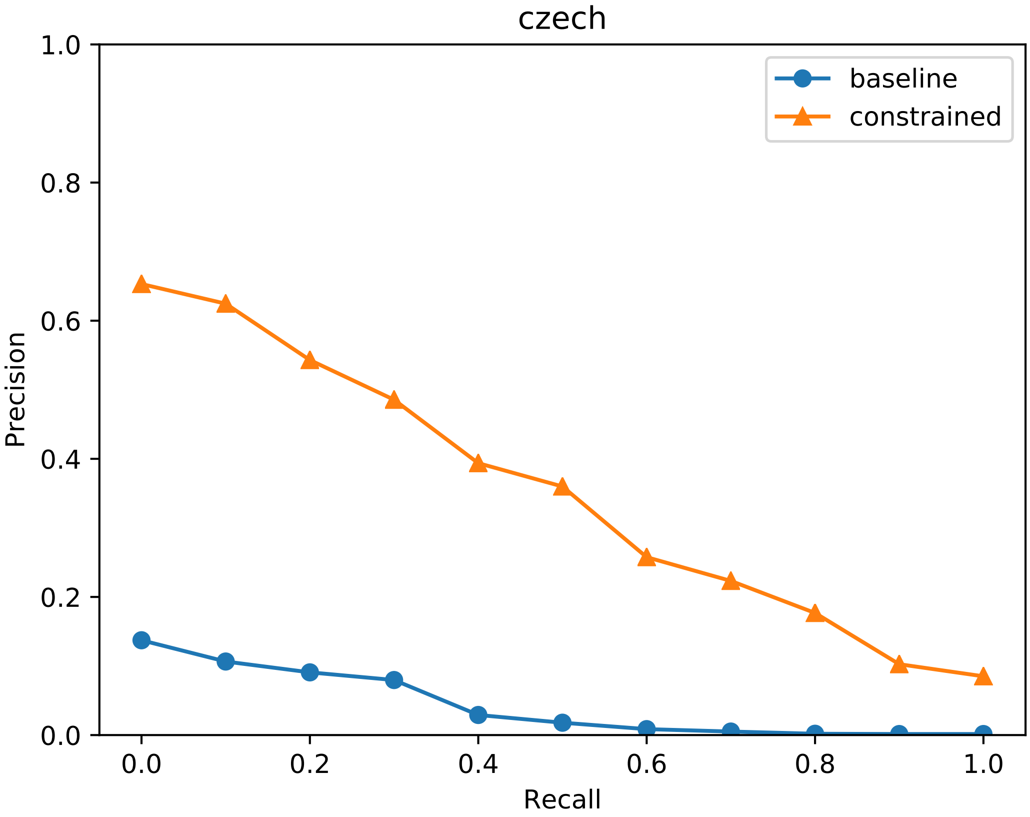

Figure 1: Precision-recall curve for the Czech dataset.

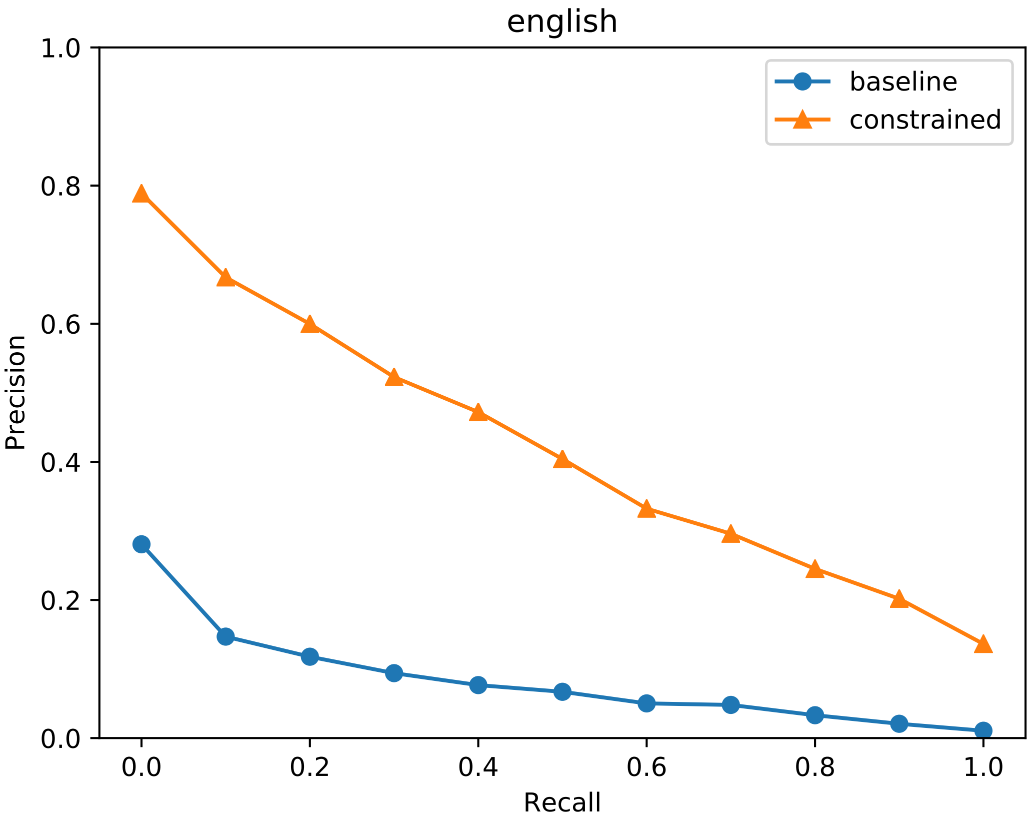

Figure 2: Precision-recall curve for the English dataset.

4 Details

4.1 Parameters

For run-1 parameters of BM25 have been tuned. Final values are in table 3

Parameter	Czech	English
B	0.45	0.31
K1	0.80	1.45

Table 3: Parameters of BM25

4.2 Instructions

There is a `README` file with instructions on how to run the retrieval system.

Computer has to have `xmllint` installed (all faculty computers in lab have it) and Python 3. Required packages are in file `requirements.txt` which can be installed by `pip3 install -r requirements.txt` Also morphodita files need to be downloaded.

All this can be done by running `make build`

The interface of the `run` program requires two additional arguments, that is `-document_path` which should be the path to the folder with document files, it can be blank if the list of documents (given by parameter `-d`) contains the whole (relative) path from `run` file to the individual documents. And `-data_prefix` which is the path to the folder in which all data files are (`A1` in my case – the extracted archive which was supplied with the assignment).

Conclusion

I think that the results are fairly good, especially since for English it is quite simple and it relies heavily on the framework.

I was quite surprised that the pseudo relevance feedback did not work. I may have been doing it wrong. But after trying all methods that I could think of (different numbers of documents, terms, tuning B and K1 parameters for each combination, etc.) and still having inferior results, I decided not to use it.