Vector Space Model

David Burian

December 3, 2022

1 Instalace

Moje řešení je implementováno v pythnu s verzí `3.10.2`. Všechny potřebné balíčky jsou k nahlédnutí v `requirements.txt` souboru.

Hlavní program je `main.py` a pouští se dle zadání:

```
python main.py -q <topics> -d <docs> -r <run> -o <out-file>
```

Pro mé vlastní pohodlí jsem si ještě napsal Makefile. Ta předpokládá, že:

- Celá složka se zadáním a daty je v `DATA_DIR`
- Program `trec_eval` je k nalezení na cestě `TREC_EVAL_BIN`

Obě proměnné jsou nastavitelné v prvních řádcích Makefile. Shrnutí co Makefile umí:

```
# Vygeneruje .res soubory pro defaultní run (run-0), oba jazyky a oba datové sety
make res

# Vygeneruje .res soubory pro run-1, oba jazyky a oba datové sety
make res run=run-1

# Vygeneruje .res soubory pro run-1, oba jazyky a train data
make res run=run-1 mode=train

# Vygeneruje .res soubor pro run-1, cs a train data
make res run=run-1 mode=train lan=cs

# Vygeneruje patřičné .res soubory (pokud je potřeba) a evaluuje je pomocí trec_eval
make {eval} run=run-1 lan=en
```

2 Experimenty

2.1 run-0

Základní řešení se chová dle zadání:

1. Z dokumentů vytáhne všechen text
2. Rozdělí text všemi následujícími znaky: `\t \n[](),.?!;:`

3. Uloží do inverted indexu počet výskytů
4. Z každé query si načte `<title>` a stejně ho rozdělí na slova
5. Vypočítá podobnost pomocí cosínu

run id	MAP	P@5	P@30	P@100	P@500
run-0_cs	0.1158	0.1920	0.0773	0.0324	0.0107