Reproducible IR needs an (IR) (Graph) Query Language

Chris Kamphuis and Arjen P. de Vries

iCIS | Data Science
Radboud University

Problem

Different implementations of the same ranking function can produce very different effectiveness scores

Problem

Different implementations of the same ranking function can produce very different effectiveness scores

System	MAP	P@5
Indri	0.246	0.304
MonetDB and VectorWise	0.225	0.276
Lucene	0.216	0.265
Terrier	0.215	0.272

Effectiveness scores BM25 ClueWeb12[1]

[1] Mühleisen et al. (2014)

Problem

Different implementations of the same ranking function can produce very different effectiveness scores

System	MAP@1000
ATIRE	0.2902
Lucene	0.3029
MG4J	0.2994
Terrier	0.2687

Effectiveness scores BM25 .GOV2[2]

[2] Arguello et al. (2015)

Problem

Different implementations of the same ranking function can produce very different effectiveness scores

System	AP	P@30	NDCG@20
Anserini	0.2531	0.3102	0.4240
ATIRE	0.2184	0.3199	0.4211
ielab	0.1826	0.2605	0.3477
Indri	0.2338	0.2995	0.4041
OldDog	0.2434	0.2985	0.4002
PISA	0.2534	0.3120	0.4221
Terrier	0.2363	0.2977	0.4049

Effectiveness scores BM25 Robust04[3]

[3] Clancy et al. (2019)

Reasons for differences

Investigating why results differ is not easy

Reasons for differences

Investigating why results differ is not easy

- Different preprocessing?

Reasons for differences

Investigating why results differ is not easy

- Different preprocessing?
- Different hyperparameter settings?

Reasons for differences

Investigating why results differ is not easy

- Different preprocessing?
- Different hyperparameter settings?
- Different function for IDF?

Reasons for differences

Investigating why results differ is not easy

- Different preprocessing?
- Different hyperparameter settings?
- Different function for IDF?
- Should documents without at least one keyword match be scored?

Reasons for differences

Investigating why results differ is not easy

- Different preprocessing?
- Different hyperparameter settings?
- Different function for IDF?
- Should documents without at least one keyword match be scored?
- Wrong implementation?

Reasons for differences

Investigating why results differ is not easy

- Different preprocessing?
- Different hyperparameter settings?
- Different function for IDF?
- Should documents without at least one keyword match be scored?
- Wrong implementation?

Components

Data management, processing, algorithms are all build on top of each other!

Use a database

Split data management from query processing

By representing the data in a database

- Easier to see differences in document representation
- Ranking functions need to be expressed precisely

Use a database

A relational database has limitations
When adding meta-data, entity information etc. the relational model is inconvenient for documents.

Use a database

A graph database to represent more complex data
Solution: Use a graph database where expressing queries that deal with more complex data structures are more easily expressed.