Search Engine Design

Prepared for Information Retrieval

Prepared by: Danielle Souza Da Silva, Georgia Dean, Adam Toth

March 18, 2022

Queen Mary University of London

Introduction and Problem Formulation

Introduction

Okapi BM25 is one of the most popular retrieval models and often considered the most effective – it is the benchmark to beat. It has many commonalities with the TF-IDF (term frequency – inverse document frequency) terms weighting algorithm, however, BM25's formulation of these factors differs by additional, tuneable parameters accounting for the verbosity and scope of documents and their lengths.

Indeed, one of the most notable differences between the traditional TF-IDF models is that BM25 adjust its final ranking scores to varying document lengths found in a document collection. While BM25 works well for managing varying document lengths, it does not account for the structure of a document.

Robertson et al. (2004) proposed an extension to BM25, called BM25F, which would weight terms present in multiple fields of a document. The idea is that different sections of structured documents may be more predictive of relevance than others (Robertson & Zaragoza, 2009). For example, a query match in a specific paragraph could be expected to provide stronger evidence of possible relevance than an equivalent match on the title.

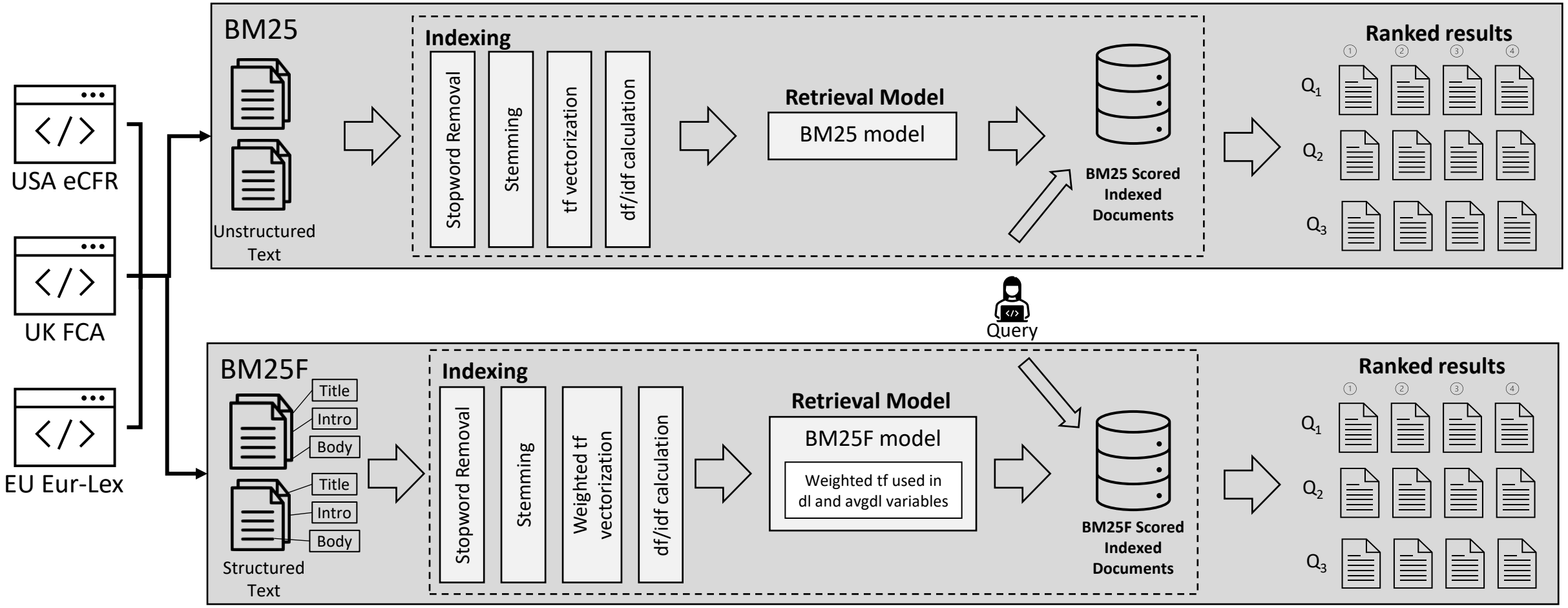
Problem Formulation

This experimental search engine will use the BM25F model to improve retrieval results of regulatory text. Regulation has been selected as the dataset due to its largely consistent structure and unique approach to topic inclusion. Typically, these documents introduce highly relevant concepts in the opening sections and generally do not repeat them in the body. The goal of using BM25F is to weight these known sections higher to improve the retrieval results. The challenge here is how to define these weights.

The minimum viable product (MVP) for this search engine will leverage the BM25F retrieval model using manually chosen weights for the given sections. A more advanced solution, however, requires the weights be optimized using gradient descent or grid-search (Svore and Burges, 2009). Once the MVP is successfully delivered, we will adapt the BM25F model to include these more complex strategies for weight optimization.

Our evaluation strategy will use the performance of the BM25 model as the benchmark. Performance will be measured using precision, recall and F1 calculations. The MVP BM25F and the potential advanced solution will be evaluated against these metrics.

Architecture



Model Description

Queen Mary University of London

BM25 (Best-match Okapi model):

BM25 extends the scoring function for the binary independence model to include document and query term weights. [W.B. Croft, D. Metzler, T. Strohman, 2015].

$$\textbf{document length}: \quad dl := \sum_{i \in V} tf_i; \quad tf: \ term\ frequency,$$

Average doc. length: $avdl$

Soft Length normalisation component: $B := \left((1-b) + b\frac{dl}{avdl}\right), \ 0 \le b \le 1;$

When b = 1 ⟹ full document-length normalisation will be performed; when b = 0 ⟹ document-length normalisation switched off.

Classic **BM25** term-weighting and document-scoring function:

$$\boldsymbol{w_i^{BM25}(tf) = \frac{tf}{k_1\left((1-b) + b\frac{dl}{avdl}\right) + tf} \cdot w_i^{RSJ}}$$

Note that: In the absence of relevance feedback information, w_i^{RSJ} will be replaced by w_i^{IDF}.

Robertson/Sparck Jones weight: $\boldsymbol{w_i^{RSJ} = log\frac{(r_i+0.5)(N-R-n_i+r_i+0.5)}{(n_i-r_i+0.5)(R-r_i+0.5)}}$,

Formula close to approximation to classical idf: $\boldsymbol{w_i^{IDF} = log\frac{N-n_i+0.5}{n_i+0.5}}$

[**N** — Size of the judged sample; $\boldsymbol{n_i}$ — Number of documents in the judged sample containing $\boldsymbol{t_i}$; **R** — Relevant set size (i.e., number of documents judged relevant); $\boldsymbol{r_i}$— Number of judged relevant docs containing $\boldsymbol{t_i}$]

BM25F:

The simplest extension method of BM25 to include weighted fields is to calculate a weighted variant of total term frequency. This implies having a similarly weighted variant of the total document length. (*Please note: 'stream' will be used to denote a field of text below.)*

$$\boldsymbol{tf_i} \rightarrow \boldsymbol{\widetilde{tf_i}} = \sum_{s=1}^{S} v_s tf_{si}, \qquad where\ tf_{si}: frequency\ of\ term\ i\ in\ stream\ s$$

$$and\ streams\ s = 1, \dots, S$$

$$\boldsymbol{dl} \rightarrow \boldsymbol{\widetilde{dl}} = \sum_{s=1}^{S} v_s sl_s, \qquad where\ v_s: stream\ weights;\ sl_s: stream\ lengths$$

$$\boldsymbol{avdl} \rightarrow \boldsymbol{\widetilde{avgl}} = average\ of\ \widetilde{dl}\ \text{across documents}$$

To allow different streams to have different characteristics (i.e. in relation to verbosity) the formula can be rearranged to include a stream-specific part (e.g. k1, b, w_i^{RSJ}). To allow b to be stream-specific, the BM25F model is presented as:

$$\widetilde{tf_i} = \sum_{s=1}^{S} v_s \frac{tf_{si}}{B_s}$$

$$B_s = \left((1-b_s) + b_s\frac{sl_s}{avsl_s}\right), \qquad 0 \le b \le 1$$

$$\boldsymbol{w_i^{BM25F} = \frac{\widetilde{tf_i}}{k_1 + \widetilde{tf_i}} w_i^{RSJ}}$$

Note that: In the absence of relevance feedback information, w_i^{RSJ} will be replaced by w_i^{IDF}

Reference: Robertson, S. and Zaragoza, H., 2009. The Probabilistic Relevance Framework: BM25 and Beyond. *Foundations and Trends® in Information Retrieval*, 3(4), pp.333-389.

Tools

Python will be used to implement the search engine components. The potential libraries that may be utilised during implementation and their description of use are listed below.

Note that a range of Python's built-in functions will be applied throughout implementation, e.g., the summation function may be used for Ranking step.

Python library Name	Description of use in proposed implementation
Beautiful Soup	It will be used to extract and inspect data from the web. It will also be used to build the dataset(s) of interest.
Scikit.learn	It will be used to implement the retrieval model's in-functions and other functions needed for evaluation step. It will also be used during the vectorisation and tokenisation step.
re (Regular Expressions)	It will be used to search for patterns in text. It'll mainly be used in the data cleaning step.
NLTK	This library will be used to perform stemming across the dataset.
ElasticSearch	It hasn't been decided yet whether the group will use this library due to the nature of the retrieval models chosen (they can be implemented using other libraries already mentioned). The group may decide to use this library if it is convenient for the retrieval model's implementation.
FastText (gensim.models)	This library may be used to capture relationships between word terms and documents and can also help the group evaluate initial performance of retrieval models.
NMSLIB	This library may be utilised to implement search index that performs at greater speed.
Pyterrier	This library may be used to tune the retrieval model's parameters, so that the models can be adapted to the dataset chosen for effective retrieval performance.

Dataset and Evaluation Strategy

Dataset

The data will be made up of regulatory texts from the United States, United Kingdom and European Union focusing specifically on financial regulation.

Jurisdiction	Regulations
United States	E-CFR (Title 12, Title 17, Title 31)
United Kingdom	FCA Handbook
European Union	Eur-Lex (Selected regulations in the subject)

Query relevant document pairs will be developed by the research team using domain expertise. There will be an emphasis on topics that are of high importance but not mentioned at high frequencies.

Evaluation Strategy

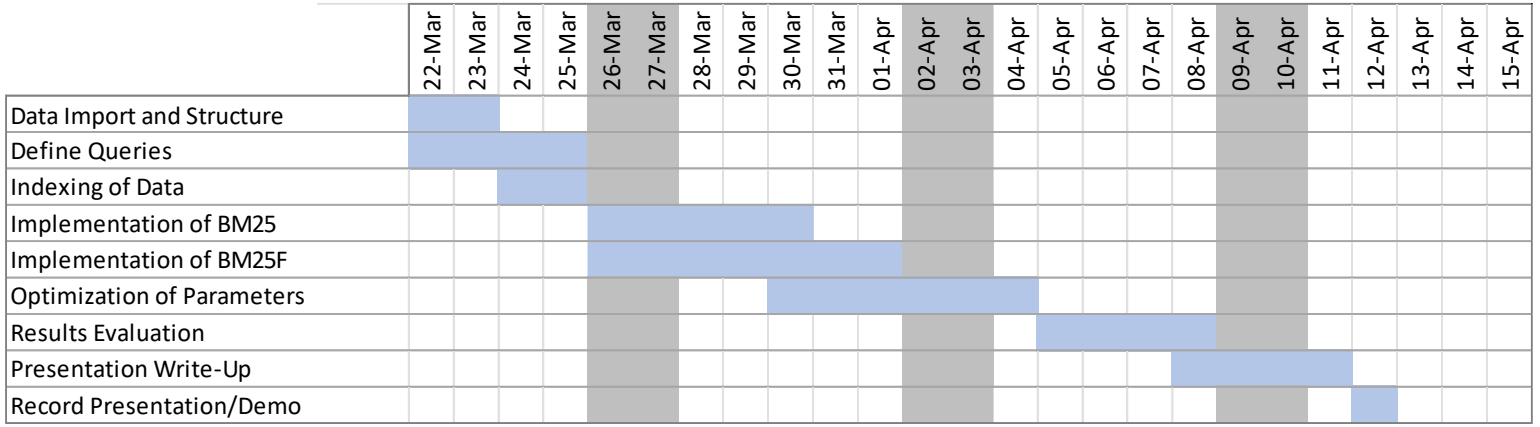
To evaluate the performance of the models will use the precision, recall and F1 metrics and compare against the benchmark BM25 Model

Precision is important to ensure that the retrieved results do not have many irrelevant documents, requiring the searcher to go through many irrelevant docs.

Recall, however, is *very* important in the regulatory compliance field because there is a low risk appetite to miss any relevant regulations.

F1 is a standard metric that captures the trade-off between precision and recall and will be included in the analysis as well.

Timeline and Task Division



Task	Team Member	Description	Due Date
Data Import and Structure	GD	Bring in and prepare data, both structured and unstructured	23 March
Define Queries	GD	Define 15-20 queries and relevant document pairs	23 March
Indexing of Data	DD, AT	Using team coding, index data to be scored-Including but not limited to, stop word removal, stemming, tf/idf calculations	25 March
Implementation of BM25	DD, AT	Using team coding, implement BM25 model on the unstructured document data.	30 March
Implementation of BM25F	GD, DD, AT	Using team coding, implement BM25F model on structured document data	1 April
Optimization of Parameters [optional]	DD, AT	Optimize BM25F weight parameters and hyper parameters for both models (potentially using gradient descent)	4 April
Results Evaluation	GD	Calculate precision, recall and F-scores to determine quality of retravel results	8 April
Presentation Write-Up	GD, DD, AT	Write presentation of	11 April
Record Presentation/Demo	GD, DD, AT	All team members will record and present search engine and results	12 April

References

W.B. Croft, D. Metzler, T. Strohman, 2015 (n.d.). *Search Engines Information Retrieval in Practice*. [online] Available at: https://www.cse.iitk.ac.in/users/nsrivast/HCC/search%20engines.pdf.

Garcia, E., 2016. A Tutorial on the BM25F Model. [online] Available at: <https://www.researchgate.net/publication/308991534_A_Tutorial_on_the_BM25F_Model> [Accessed 13 March 2022].

Robertson, S. and Zaragoza, H., 2009. The Probabilistic Relevance Framework: BM25 and Beyond. *Foundations and Trends® in Information Retrieval*, 3(4), pp.333-389.

Robertson S., Zaragoza H., Taylor M.J., 2004. Simple BM25 extension to multiple weighted fields. *Conference: Proceedings of the 2004 ACM CIKM International Conference on Information and Knowledge Management, Washington, DC, USA, November 8-13, 2004.*

Svore, K. and Burges, C., 2009. A machine learning approach for improved BM25 retrieval. *Proceeding of the 18th ACM conference on Information and knowledge management - CIKM '09*,.