Team: CTRL Freaks (Group 12)

CA Part 2

Raaghav Sawhney
23364370
MAI
Trinity College Dublin
rsawhney@tcd.ie

Milena Nastin
21365848
MCS
Trinity College Dublin
ruszyncm@tcd.ie

Shreya Shah
25337017
MSc. CS - DS
Trinity College Dublin
shshah@tcd.ie

Qiming Nie
20336894
MCS
Trinity College Dublin
nieq@tcd.ie

Ritik Saxena
25336681
MSc. CS - DS
Trinity College Dublin
rsaxena@tcd.ie

ABSTRACT

Relevance models and Query expansions are techniques used to increase recall in an ad-hoc information retrieval system. In this paper, we implement a custom variant of the RM3 model that focuses on entity biased mining for the expansion of the query. We also filter out negative constraints and irrelevant text from a query to find out the most suitable documents for it.

KEYWORDS

Information Retrieval, Query expansion, RM3, TREC, Pseudo Relevance Feedback.

1 INTRODUCTION

Information Retrieval (IR) systems aim to match a user's information need with relevant documents, but real-world queries are often short and ambiguous. To overcome this vocabulary mismatch, modern IR systems frequently incorporate relevance models. Relevance models are used to determine P(w|R), the probability of observing a word w given the class of documents R relevant to the user's need.[1]

One of the best known performing relevance models is the RM3 model[4]. It is useful for both relevance and pseudo-relevance feedback (PRF). PRF expands the original query using terms from the top-ranked initial results. This reduces vocabulary mismatch, improves recall, and often yields substantial gains in performance, although poor initial results can still cause query drift [2].

Underpinning the relevance feedback is the Okapi BM25 model [3] which utilises the occurrence of terms, term saturation and document length. They provide a high quality source for candidate terms for the subsequent model. We utilise the Lucene text search engine for its indexing capabilities and the English analyser to normalize the text data and reduce vocabulary mismatch.

To assess our model, we conduct experiments on the TREC Dataset (FBIS, FT, FR94, LATimes). All performance metrics are computed using the trec_eval suite and the key measures to be looked at are the MAP (to ensure overall ranking quality) and the P@20 (to evaluate the system's ability to generate highly relevant content).

2 RELEVANT WORK

Classical relevance feedback and PRF aim to improve ad-hoc retrieval by automatically expanding the user's query with terms taken from the top ranked documents. This works under the assumption that early-ranked documents are more likely to be relevant, which can improve recall but also may cause query drift when documents are noisy or off-topic. Earlier studies emphasised careful tuning of parameters, like how many documents to use, how many expansion terms to keep, and how strongly expansion should be mixed with original query, all to balance recall gains with ranking stability [1].

Some early work highlights the importance of refining the set of feedback documents using PRF rather than only adjusting term-weight schemes[2]. This research demonstrated that blindly using the top-ranked documents often introduces query drift, and proposed 2 alternatives: manually designed boolean and proximity based filters to select cleaner feedback sets, and a fully automatic method that leverages term co-occurrence statistics to estimate word correlation and identify better candidate documents.

RM3 has become one of the most reliable and influential PRF baselines on TREC-style benchmarking. RM3 builds a relevance model over the feedback documents and then interpolate it with the original query to help preserve the user's intent while simultaneously exploiting evidence from

the collection. When correctly tuned, RM3 consistently improves MAP across test collections[4].

3 EXPERIMENTAL SETUP

3.1 DOCUMENT COLLECTION

Since the whole dataset is written in English, we utilised the standard English Analyser to process the text. It utilises a four step procedure for text cleaning, which includes breaking text into individual words (tokenisation), and lowercasing them. It next removes the stop-words (the, and, or, is) which appear frequently but carry very little semantic meaning. Finally, it applies the porter stemming algorithm which removes the suffixes based on a fixed set of rules.

COLLECTION NAME	DOCUMENT COUNT
Financial Times(FT)	209566
Federal Register(FR)	55152
LA Times	131166
FBIS	130365
Total	526249

THE TREC DATASET

3.2 TOPICS

Each query is represented in the form of a topic, which contains three distinct fields: title, description and the narrative. The title is a 2-3 word query, the description encapsulates the domain of the query and the narrative is a comprehensive paragraph describing the specific relevance criteria, including what should be rejected.

To construct the initial retrieval query, we create a boolean query with the narrative boost factor of 0.5, the description 1.3, and the title 3.0 (primary anchor). We filter out negative constraints, (the sentences containing ‘not relevant’, ‘irrelevant’) from the narrative because our underlying retrieval model relies on term frequency matching and lacks semantic negation handling, which would otherwise be treated as positive relevance signals.

4 IMPLEMENTATION

4.1 THE ANCHOR QUERY

The anchor query utilizes the BM25 model to generate initial feedback of relevant documents for term mining. BM25 is an evolution of the traditional TF-IDF (term frequency, inverse document frequency), because it addresses the critical limitation of the aforementioned model. BM25 via its saturation (k1=1.2) parameter ensures that the increasing frequency of a term in a document has diminishing contributions to the score. It also contains a length normalisation parameter (b=0.75), which helps provide a robust baseline probabilistic model for identifying relevant documents by making the term frequency comparable in relation to the document length. The formula for calculating BM25 score is:

$$SCORE(D,Q) = \sum_{q \epsilon Q} IDF(q).\frac{f(q,D).(k1+1)}{f(q,D)+k1.(1-b+b.\frac{|D|}{avgdl})}$$

Where: f(q,D) is the term frequency of q in document D; |D| is the length of document D in words; avgdl is the average document length in the collection; k1 and b are tuning parameters.

4.2 THE ENTITY BIASED MINING

After executing the initial anchor retrieval, we isolate the top 20 documents to serve as the pseudo-relevant feedback set, under the assumption that the top ranked results contain a high density of terms related to the topic. To optimize the quality of the expansion vocabulary (and also to save compute resources), we restrict the term extraction process to 200 tokens (25x faster than if we search 5000 words) of each feedback document. This truncation can be classified as a ‘lead bias’ which estimates that the critical entities and the important terms are concentrated in the opening paragraphs.

The standard RM3 approach uses the formula:

$$P(w|D) = \frac{count}{length}$$

Where count is the word frequency and the length refers to the document length. This leads the model to select the terms with the highest frequency in the distribution but can lead to ‘noise’ if the sample size is limited in our case. Therefore, we utilised a custom variant of standard RM3 to better suit our case which helps us identify terms which are rare and discriminative. We use the formula:

$$S = (log(\frac{N}{df}) + 1) * score$$

Where : S is the word weight in the expansion; N is the term frequency; df is the document frequency and score is BM25 score for the document.

This formula incorporates an inverse document frequency term to penalize common tokens. We also integrate a heuristic of multiplying those terms which are capitalized other than the first word of a sentence with a 1.25 multiplier (the entity bias) and other terms with 1 to bias the model to select terms which might be of greater relevance.

4.3 THE FINAL SEARCH

With the terms now appropriately weighted, we select the words with the top 40 weights to form the basis for our expansion. The standard RM3 utilises query expansion of the form:

$$QF = (1 - \alpha) * QO + \alpha(FT)$$

Where: QF is the final query; QO is the original query; FT are the feedback terms and α is a hyperparameter.

We boost the feedback by a factor of 0.5, while adding it to the query. The effective α is unclear as the original query has multiple weights associated within it (3, 1.3, 0.5). In the most extreme scenario, the overall weighting of the original query would be 4.8, leading alpha to be approximately 0.1. While the usual values of α range from 0.4 to 0.6, the number of terms we use in our expansion have a high aggregate score contribution to the query even though we have a conservative estimate of α.

This final query constructed after the expansion is utilized again in conjunction with the BM25 model to select the top 1000 documents with the parameters specified which are then returned as the top documents.

4.4. OPTIMIZATION

Parameter optimization is executed by a systematic parameter sweep. We optimize six different critical parameters: title boost, description boost, narrative boost, top docs number, weight multiplier and the query expansion number.

The analyzer and model iterations are performed in isolation, with no other parameters being set. This allows for a determination of an unbiased baseline.The parameters were optimised sequentially utilising a greedy strategy in the order mentioned above. The default value is determined by prioritizing MAP, Precision and DCGP in the order provided.

5 EVALUATION

5.1 SETUP

We evaluated our retrieval model within the specific context of the TREC News dataset described in Section 2. For each query topic, our Lucene system generates a standard TREC-format run file. Each entry records the topic ID, document ID, ranking position, and retrieval score. We compared performance against the official qrels and measured results using standard TREC evaluation metrics. Our primary metric is MAP (Mean Average Precision), which reflects the overall ranking quality of relevant documents. We also report P@20 and nDCG@20 to capture precision and ranking quality at the top of the list.

5.2 COMPARATIVE EXPERIMENTS

5.2.1 EFFECT OF ANALYZERS

In our first set of experiments, we fixed the retrieval model to BM25 and varied only the analyzer to assess its impact on retrieval performance. We compared two configurations, with the relevant details summarized in the table below.

System	MAP	P@20	nDCGP@20
KstemAnalyzer + BM25	0.22	0.35	0.38
EnglishAnalyzer + BM25	0.3044	0.482	0.55335

Under identical query specifications (title + description), EnglishAnalyzer demonstrated the best performance across both metrics. Compared to KstemAnalyzer, it achieved significant improvements in MAP, P@20 and nDCG@20. This highlights the importance of stopword removal and stemming strategies for this news retrieval task. Based on this finding, we uniformly adopted the EnglishAnalyzer as the default configuration in subsequent experiments and in the final submitted system.

5.2.2 EFFECT OF RETRIEVAL MODELS AND QUERY FORMULATION

After configuring the analyzer as the “English Analyzer,” we compared several retrieval models and query formulation strategies. The table below illustrates the effectiveness of three typical models under our default query construction approach (title + description + filtered narrative).

System	MAP	P@20	nDCGP@20
BM25	0.3044	0.482	0.55335
LMDirichlet	0.2791	0.432	0.4827
BM25 + IB Hybrid	0.3183	0.486	0.5561

When using the BM25 model, MAP is 0.3044, P@20 is 0.482, and nDCG@20 is 0.553. Using the LMDirichlet model, all three metrics decrease. This indicates that under the current query and parameter settings, BM25 is better suited for this news retrieval task. BM25 + IB Hybrid performed best among the tested models. This indicates that introducing an IB (divergence-from-randomness) component in specific fields can slightly improve ranking quality.

In additional experiments (not listed in the table), we further varied query construction while keeping the retrieval model fixed. We compared queries using only the title against combined queries using title, description, and filtered narrative, while adjusting weights for different fields. Results consistently showed that the “rich query” using title + description + filtered narrative achieved significant improvements in MAP and precision over the top-ranked results compared to using title alone.

Although the BM25 + IB Hybrid model showed slight advantages over pure BM25 across metrics in the no-feedback setting, its additional gains became unstable when combined with pseudo-correlation feedback or RM3 style expansion. It amplified feedback noise, leading to increased overall MAP volatility. This resulted in the BM25 + IB Hybrid hybrid model performing worse than pure BM25. Furthermore, as a classic baseline model, BM25 aligns more readily with baseline results provided by the course and reports in relevant literature, facilitating interpretation. Based on stability, reproducibility, and ease of analysis and comparison, we ultimately selected BM25 as the foundational retrieval model for all pseudo-correlation feedback and RM3 style configurations.

With the results of parameter sweeping we confirmed the most important parameters were expansion number and title boost, which brought quite a significant improvement in the score compared to other boost values.

5.3 BEST CONFIGURATION AND DISCUSSION

Based on the above comparisons and experiments, we ultimately selected the following configuration: EnglishAnalyzer + BM25, employing a weighted query based on title/description/filtered narrative, combined with entity-enhanced RM3-style pseudo-relevance feedback. Under this configuration, the system achieved a MAP of 0.3651, P@20 of 0.54, and nDCG@20 of 0.6005 on the official trec_eval benchmark. Compared to the initial baseline (BM25 retrieval using only titles), this represents a relative MAP improvement of nearly 60%.

Those choices were backed by not only the improvement in the implementation that we were actively working on but also by the parameter sweeping, which allowed us to agree that the values we chose previously were the best choice available.

These results demonstrate that, on one hand, a well-designed analyzer and query construction form the foundation for improving IR system performance. On the other hand, pseudo-relevance feedback without proper term filtering can significantly degrade MAP. By integrating document scores, IDF, and entity information into a refined PRF design, overall ranking quality can be markedly improved while maintaining top-ranking precision.

6 CONCLUSION

Our approach provided an entity-biased relevance model which tackles the vocabulary mismatch problem in ad-hoc information retrieval. We successfully integrated entity recognition without the need of semantics. The experiment shows that a conservative expansion strategy can increase recall with a multi field anchor query while reducing query drift.

However, there are limitations to our model. First, we introduce entity recognition from capitalization as a proxy for the named entity recognition in machine learning models. Our models don't capture context inherently since they are counting models rather than contextual. Our optimization technique is primarily static and is based on the queries initial performance. As compared to classic models, relevance models also have significantly longer response times. Lastly, our negation of constraints relies on simple string matching which can’t detect more complex negation scopes accurately.

Future work in this space could include looking at word embeddings to introduce context, and also a pre-computed thesaurus to remove non-entities from query expansion.

REFERENCES

[1] Victor Lavrenko and W. Bruce Croft. 2001. Relevance Based Language Models. In Proceedings of the 24th Annual International *ACM SIGIR Conference on Research and Development in Information Retrieval* (New Orleans, Louisiana, USA) (SIGIR ’01). Association for Computing Machinery, 120–127. https://doi.org/10.1145/383952.383972

[2] Mitra, M., Singhal, A., & Buckley, C. (1998). Improving automatic query expansion. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 206–214). ACM. https://doi.org/10.1145/290941.290995

[3] Robertson, S., & Zaragoza, H. (2009). The probabilistic relevance framework: BM25 and beyond. *Foundations and Trends in Information Retrieval, 3*(4), 333–389. https://doi.org/10.1561/1500000019

[4]Yannakoudaikis, H. (2018). *Lecture 7: Relevance feedback and query expansion* [Lecture slides]. University of Cambridge.