Relevance Feedback Based Query Expansion Model using Condorcet Ranking Approach

Kompal Aggarwal[1] ,Rajender Nath[2]
1 Department of Computer Science & Applications
Kurukshetra University, Kurukshetra
2Department of Computer Science & Applications
Kurukshetra University, Kurukshetra.

Abstract
The relevant results can not be fetched using keyword based search because they can not detect the exact meaning of the expression or term and relationship exist between them during web search**.** Pseudo-Relevance Feedback (PRF) is a query expansion method used for better performance of information retrieval systems. These methods are used to remove redundant and irrelevant terms from the top retrieved feedback documents for a given user query. This paper proposes an architecture for query expansion term reweighting using BM25, Condorcet and Rocchio algorithm(BMCQE).The experimental results demonstrates that the proposed BMCQE approach achieve a significant improvement over related state-of-the-arts approaches.

Key Words: Information Retrieval, Relevance Feedback, Rocchio classification

1. Introduction

The major challenge in the Information Retrieval (IR) system is to extract the documents that are relevant according to the user needs. Most of the users on the Web cannot find the results which they require. An IR model specifies how a query and a document are represented and how the document relevance to a user query is defined. Relevance feedback is one of the techniques used for improving retrieval effectiveness.

The PRF based Query Expansion (QE) is one of the most successful and reliable techniques to solve this problem. In this technique some top documents which are retrieved in the first iteration are used for the purpose of expanding the original user query. To solve the above problem, there is requirement of automatic PRF based QE techniques so that the original user query can be reformulated in an automated manner. Also all terms of top retrieved feedback documents are not required for the QE. Some of the QE terms may be irrelevant or redundant. Some terms can even misguide the result, when irrelevant QE terms are larger than relevant terms. QE selection targets to remove irrelevant and redundant terms from the term pool which is formed from top retrieved feedback documents. One of the challenge is to improve the performance of each of the QE terms selection methods. The multiple QE terms selection methods can be combined for better performance by considering the advantage of the strength of the individual method.

BM25 (BM stands for Best Matching) is probabilistic retrieval framework based ranking function and used by search engines to rank the documents which matches with the given query according to their relevance.

2. Related Work

Rocchio's [1] algorithm was relevance feedback model. The basic theory was to move query vector towards relevant documents centroid and far from irrelevant documents centroid. The new query vector was formed from concatenation of initial query and terms representing the feedback documents . [2] provided a detail study on the approach used for relevance feedback in information access systems. In [3] authors provided a model that considers both positive and also the negative feedback. Authors in [4] used collaborative filtering for pseudo relevance feedback. [5] used the concept of probability distribution of terms both in relevant and in the whole collection that provides the frequency of the terms that makes the probability of relevant documents divergent. [6] provides the ranking algorithms based on various ranking principles which are used in probabilistic IR model. [7] presented method for query reweighting which was based on the user's relevance feedback for improving the performance of retrieval system. [8][9] deals with Condorcet ranking aggregation algorithm based upon the concept of the majority. In this algorithm the pair comparison was performed and winner candidate beats all other candidates. A candidate which is non-ranked loses its score than all other candidates having ranked. All the candidates more than one which are unranked tie with each other.

3. Problem Formulation

An efficient BM25 method for selecting initial top relevant documents has been used by [10]. All the terms which are unique and from top *n* retrieved documents are selected to form term pool. Then, the author presents different ranking algorithm to rank the terms in term pool. The Borda rank aggregation methods is applied for aggregation of different ranked lists of expansion terms selected by different ranking algorithm called Borda Based Query Expansion

(BBQE) .Then the author apply semantic similarity approach to filter out the irrelevant and duplicate expansion terms in context to user query called Borda and Semantic Based Query Expansion (BSBQE)., The query is expanded by using the terms after applying the Rochio approach.

It has following demerits (i) The keywords extracted after applying Borda count aggregation method may not always be unique because the method may result to produce terms having same rank as keywords because pairwise comparison between terms is not being performed (ii) Thede terms preferred by majority of the documents may not be selected as keyword rather the term having maximum rank is the winner and selected as keyword.

4. Proposed Relevance Feedback Based Query Expansion Model Using Condorcet Ranking Approach (BMCQE)

To address the above mentioned demerits a BMCQE method is proposed. For best formation of keywords, the terms are retrieved from each of the top n documents using BM25 and perform pair wise comparison of keywords. Condorcet algorithm is applied for aggregating the terms retrieved. The main steps of this method are given below:

Step 1: Apply Okapi-BM25 function for retrieving top relevant documents for the given user query.

Step 2: Retrieve top n relevant documents from the documents retrieved.

Step 3: The top m keywords are retrieved from each selected top n documents using BM 25 methods and formed keyword pool. Find the occurrence of each keyword in their respective document and return < docid, keyword, count> .

Step 4: Apply Condorcet rank aggregation method to extract top final keywords from the keyword pool formed in step 3.Call Procedure Condorcet().

Step 5: Top *m* terms or keywords obtained from step (4) are used to expand the user query using Rocchio approach.

Procedure Condorcet() Repeat the substeps I to III for all unique keywords i.e.k1,k2.....ki (1<=i<=M) retrieved from top n documents .

(I)Perform pair-wise comparison of all unique keywords retrieved from top n documents and represent in table form of size M*M having M Rows for each unique keyword and M Columns for each unique keyword in format keyword<win : lose : tie>. All unique Keywords(M) i.e k1,k2....ki (1<=i<=M) in each row have values for each column i.e. Kj<win : lose : tie>(1<=j<=M and i<>j) as follows:

(a) If a keyword (ki) count is more than kj count in k documents out of all n documents then win score of ki with respect kj is k.

(b) If a keyword (ki) count is less than kj count in j documents out of all n documents then lose score of ki with respect to kj is j.

(c)If there is a tie between keywords ki and kj in all n documents then tie score is 1 otherwise tie score is 0.

(II) After the construction of pair-wise comparison matrix in substep(I) of all unique keywords obtained from different top n documents, the pair-wise analysis is done having format <keyword, Win score, Lose Score, Tie score>.The values are assigned to win score, lose score, and tie score of the keywords as follows:

(a) If a keyword(ki) is ranked *p* times ahead of every other keyword then win score of ki is p.
(b) If a keyword(ki) is ranked p times after every other keyword then lose score of ki is p.
(c) If there is *p* times tie between ki and any other keyword then tie score of ki is p.

(III) To rank final keywords from substep(II), the win and lose scores of keywords are used.

(a) If a keyword(ki) has more win score than another keyword win score in substep(II), then ki will win over another one.
(b) If their win score are equal, then keywords lose scores are considered, and a keyword that has a less lose score wins.
(c) If both, win and lose scores of keywords are equal, then the keywords are tied.

Some high ranked keywords selected by Condorcet scheme are used for query expansion.

The detailed steps of the proposed method are depicted in flow chart form as in Figure1 and Figure 2.

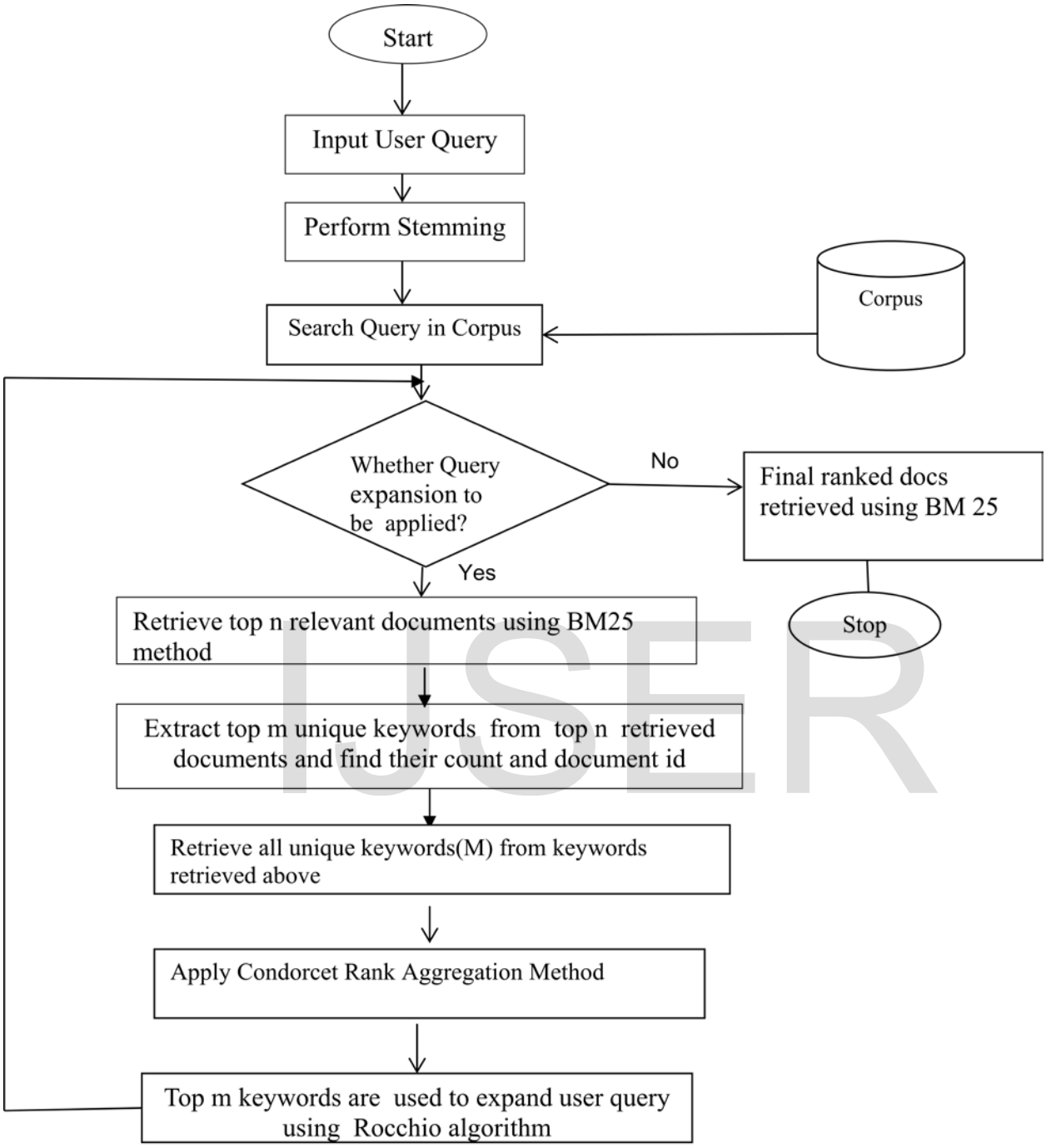

Figure 1: Flow Chart of Proposed BMCQE Approach

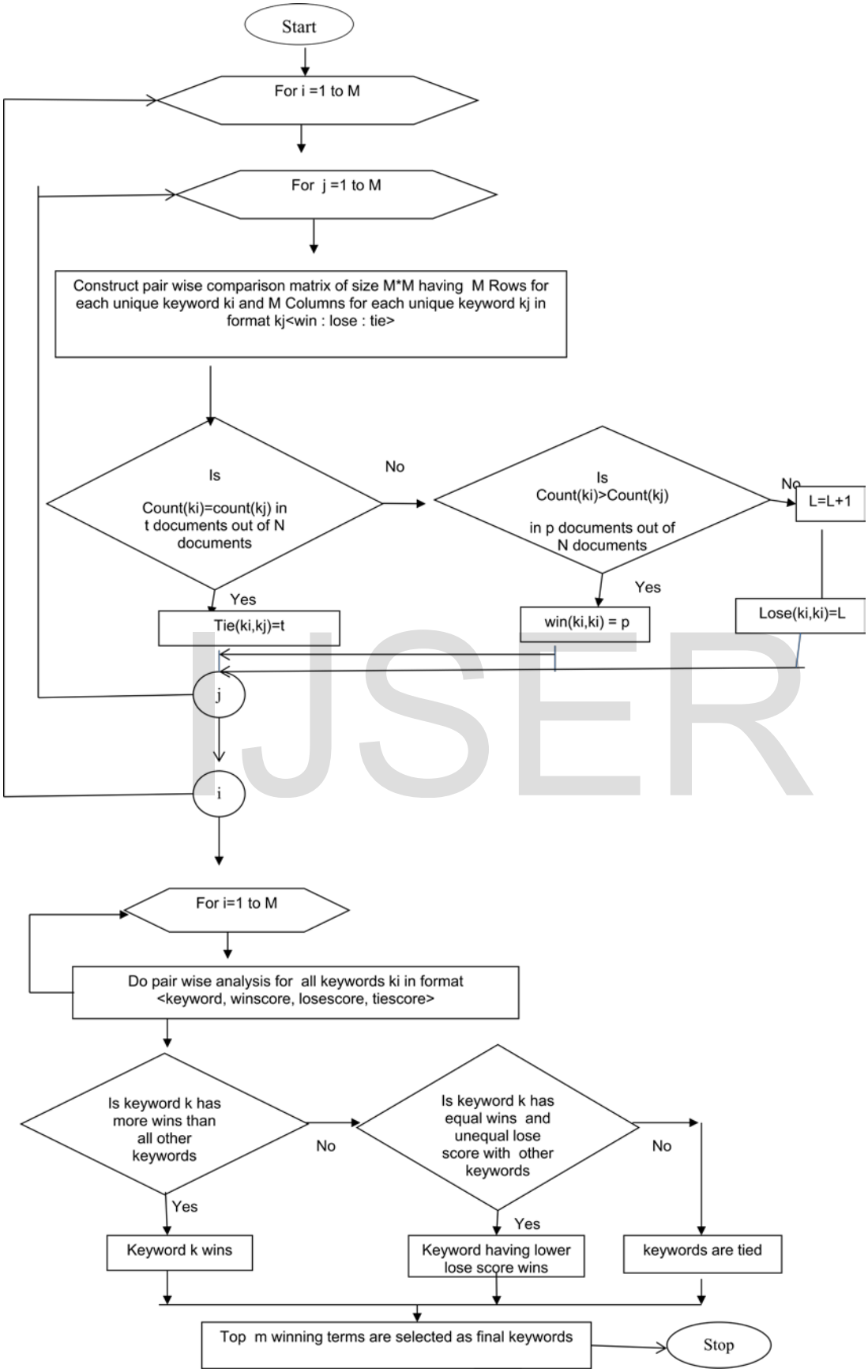

Figure2: Flowchart of Condorcet Rank Aggregation Method

5. Experimental Setup

For checking effectiveness efficiency, the proposed BMCQE Model is programmed in python 3.0. In this experiment, a corpus of 3204 documents by CACM [11] is taken. There is a corpus.txt in which each document id begins with a # and the following lines are the document contents that are already stemmed.

The five queries taken as input for experiment are given in table 5.1

Table 5.1: Input Queries

• Portable operating
• Parallel algorithm
• Applied stochastic process
• Perform evaluation and model of computer system
• Parallel process in information retrieval

Let the initial input query is portable operating. The Proposed BMCQE Model returns top relevant documents using BM25 for this query.

Table 5.2: Top Relevant documents using BM25

Query	Query after stemming	Document score	BM 25
Portable operating	Portabl oper	3127	12.648879702
		2246	11.4125974757
		1930	9.54090934979
		3196	8.95596644965
		2593	5.77381726478
		2555	3.4118394933
		103	3.36067935759
		1334	3.25615977373
		1680	3.24172877928
		1591	3.23277418446

Then the proposed BMCQE Model asks from the user to apply for relevance feedback or not.

If answer is 'yes', then the proposed model ask for the required number of relevant documents i.e. value of n is to be entered.

Enter how many relevant documents to take: 5.

Then the proposed model display top five relevant documents along with their keyword and its occurrence in that document in format <keyword, count>.

Document No: 3127
[<'system',5>,<'Program',4>,<thoth',4>]
Document No: 2246
[<'Languag',7>,<'system',5>,<'thi',5>]
Document No: 1930
[<'random',9>,<'gener',9>,<'number',8]
Document No: 3196
[<'random',9>,<'gener',9>,<'number',8]
Document No: 2593
[<'random',9>,<'gener',9>,<'data',9>]

Then the Condorcet algorithm is applied on keywords retrieved document wise to extract final keywords required for query expansion. The proposed model displays the following output:

The keywords retrieved after Condorcet approach are: random, gener, data

Finally the query is reformulated using final keywords retrieved above using Rocchio approach and the proposed model display the following modified query:

Revised Query: oper gener random data portabl

Re-run this modified query on corpus and return the new results according to the feedback. The top relevant documents retrieved after applying BMCQE model are:

Table 5.3: Top Relevant Documents using BMCQE

Document No.	Document score
1930	18.317304732
3127	12.648879702
2246	11.4125974757
1750	10.7508041389
1951	9.22136913547
2593	9.10810427762
3196	8.95596644965
856	8.63093132459
2176	8.63024606105
2516	8.3603302309

Evaluation Parameters

Recall (*R*) and Precision (*P*) are parameters that are used to evaluate the performance of information retrieval system and are calculated as

$$\text{Recall}(R) = \frac{\text{Set of Relevant documents retrieved}}{}$$

$$\text{precision}(P) = \frac{\text{Set of all Relevant documents} \cap \text{Set of Relevant documents retrieved}}{\text{Retrieved documents set}}$$

6. Results & Discussions

The Table 5.4 shows the retrieval performance of proposed query selection approach BMCQE in terms of both average precision and recall on corpus dataset. The proposed approach is compared with **BBQE** [10]. The experimental work shows that the performance of proposed approach achieved improvement over Okapi-BM25 model and BBQE methods.

Table 5.4: The comparison of Proposed Model with BM25 and BBQE for CACM dataset

Methods	Top 3 Retrieved documents		Top 5 retrieved documents		Top 10 retrieved documents	
	Average Precision	Average recall	Average precision	Average recall	Average precision	Average recall
Okapi-BM25	0.1045	0.1247	0.1342	0.1686	0.1432	0.1856
BBQE	0.1854	0.2635	0.1945	0.2836	0.2176	0.2952
Proposed	0.2054	0.3442	0.2150	0.3888	0.2386	0.4024

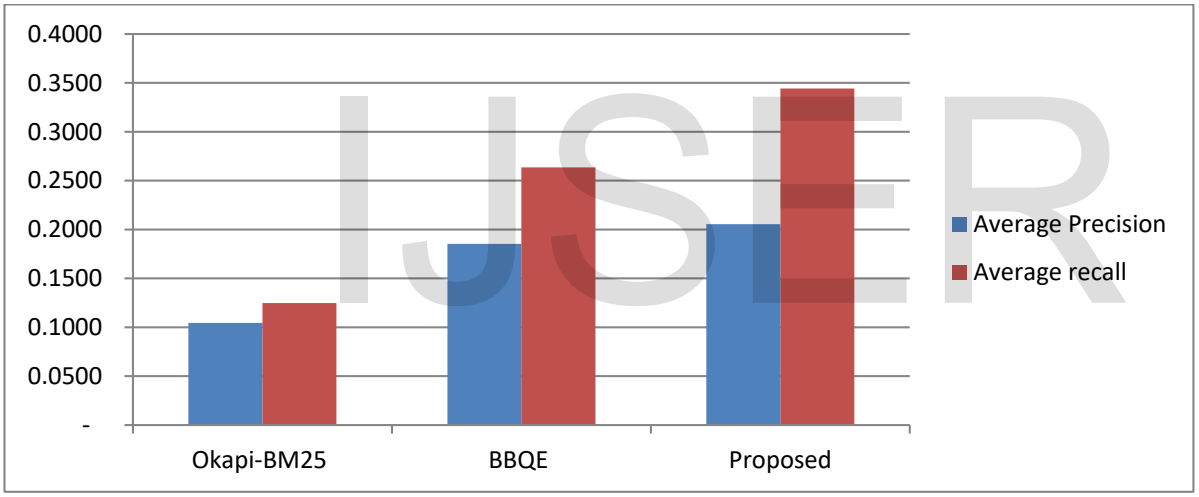

Figure 5.1: Performance of proposed approach over Okapi-BM25 model and BBQE method with top 3 retrieved documents

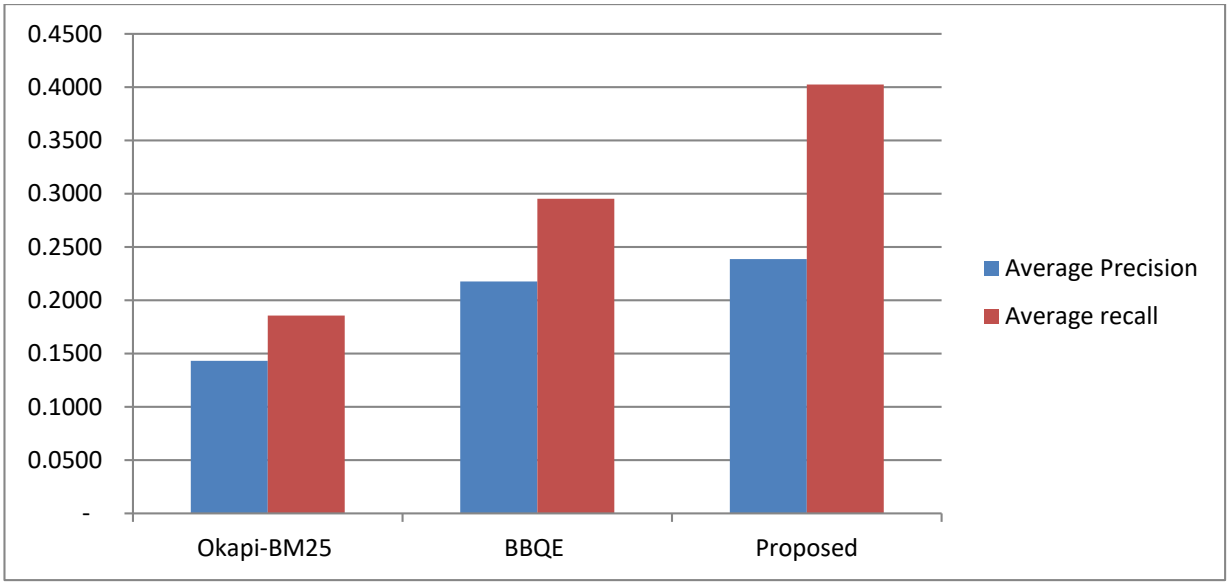

Figure 5.2: Performance of proposed approach over Okapi-BM25 model and BBQE method with top 5 retrieved documents

7. Conclusion

For best query results, search engines have to works in faster and effective manner. For this, each crawler need to identify the requirement of the user. This can be taken as a profile i.e. experience of the user on search results. In this work, an attempt has been made for reformulating the query to exclude non-relevant documents and include the relevant documents for the user query using Condorcet and Rocchio approach. The parameters in Rocchio must be analyzed and tuned to make the algorithm perform better. By the experiments on corpus of 3204 documents these parameters are tuned accordingly. It has been observed that the proposed algorithm performs better than the existing methods like BM25 and BBQE. Still there are some limitations of relevance feedback as it will not work in case of vocabulary mismatch and computing cost, which can be taken as future work for this framework.

References:

[1] Rocchio J.J. Relevance Feedback in Information Retrieval The SMART Retrieval System -experiments in automatic document processing,1971, Chapter 14, pp 313-323.

[2] Ruthven, I. and Lalmas, M. A Survey on the use of Relevance feedback for Information access Systems". Knowledge Engineering Review, 2003, pp. 95-145.

[3] Lee C., Lee G. G. Information gain and Divergence-based Feature selection for machine learning-based text categorization. Information Processing & Management. 2006;42(1):155–165.

[4] Zhoua D., Truranb, M., Liua J., Zhanga S. "Collaborative pseudo-relevance feedback", Expert Systems with Applications, Volume 40, Issue 17, 1 December 2013, Pages 6805–6812.

[5] V. Karyotis, T. Kasrinogiannis, G.Androulidakis, C.Malavazos, M.Lazaridis : Victory- Hypertech-D-WP4-V9-D4.4 Relevance Feedback Algorithms

[6] Vishwa Vinay, I. J. Cox, Milic-Frayling, Ken Wood, Evaluating Relevance Feedback Algorithms for Searching on Small Displays. ECIR 2005, Pages 185-199 .

[7] Stephen Robertson and Hugo Zaragoza . The probabilistic Relevance Framework: BM25 and Beyond, Foundation and Trends in Information Retrieval,2009,Vol.3,pg 333-389.

[8] Amaia Portugal (2011), Contributor Enhancing Europeana, the great European digital library, CORDIS-Community research and Development Information Service ,January 2011.

[9] Carpineto C and Romano G. A survey of Automatic Query Expansion in Information Retrieval. ACM Computing Survey 2012; 44(1): 1-50

[10] Jagendra Singh, Aditi sharan, Relevance Feedback Based Query Expansion Model Using Borda Count and Semantic Similarity Approach, Computational Intelligence and Neuroscience,2015

[11] Test Collections, http://ir.dcs.gla.ac.uk/resources/test_collections/ August 2017