Evaluating the Quality of Query Refinement Suggestions in Information Retrieval

Ramesh Nallapati and Chirag Shah
Center for Intelligent Information Retrieval
Department of Computer Science
University of Massachusetts
Amherst, MA 01003
{nmramesh, chirag}@cs.umass.edu

ABSTRACT

Automatic suggestion of alternative terms to refine a user's query is an effective technique to help the user quickly narrow down to his(her) specific information need. However, evaluating the effectiveness of these suggestions has remained quite subjective, with a vast majority of the past work relying on expensive user studies.

In this work, we look at this problem from the IR perspective. We propose two objective measures that evaluate the quality of Query Refinement (QR) suggestions, based on the degree to which the documents retrieved by the QR suggestions, when used as queries, capture the overall sub-topical structure underlying the topic of the original query. The first measure, known as Maximum Matching Averaged Mean Average Precision (MM-AMAP) requires labeled documents for the sub-topics underlying the query's topic. The second measure which we call Distinctness and MAP based F1 (DMAP-F1) requires only labeled documents that are relevant to the original query.

We also define a series of simple QR suggestion techniques, each of which is intuitively better than the previous ones and evaluate them using our measures on TDT3 and TDT4 corpora. Our experiments show that our evaluation metrics numerically capture our intuitive expectations on performance, thus informally validating our measures.

Further, we also show that the second metric DMAP-F1, that does not require sub-topic judgments, is consistent in results as well as statistically highly correlated with the first metric. This allows us to perform extensive evaluations of the quality of QR suggestion techniques on standard TREC collections in the future.

Categories and Subject Descriptors

H.3.4 [**Information Storage and Retrieval**]: Systems and Software—*Performance evaluation (efficiency and effectiveness)* ; H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*Selection process*

Submission to CIKM 2006. Please do not distribute.

General Terms

Measurement, Performance, Experimentation

Keywords

Evaluation measures, query refinement, terminological feedback

1. INTRODUCTION

In today's age of information explosion, it is no longer sufficient for IR systems to merely present a ranked list of documents relevant to a user's query. Considering the amount of available information and the complexity of information needs, it is not realistic to expect the user to patiently sift through the traditional ranked list that runs into thousands of documents for most queries. Hence, in addition to the ranked list, the user needs further assistance from the system in narrowing down the search and quickly discovering the documents pertaining to his(her) specific information need. As Henninger and Belkin rightly put it: "Information retrieval systems must not only provide efficient retrieval, but must also support the user in describing a problem that s/he does not understand well." [3]

Query Refinement (QR) suggestions, also called *terminological feedback* in the literature, is an effective way to assist the user in quickly locating the relevant documents. In this technique, the user is presented with a few alternative suggestions for refining the original query. The user is expected to choose one of the suggestions, which will then be appended to the original query and a new list of retrieved documents corresponding to the refined query are presented to the user. The technique of QR suggestions is best understood through the illustrative examples shown in table 1, which we obtained from popular search engines. For example, when a user types in a query "computer science", the popular search engine *ask.com* provides the user with several alternatives such as "research", "careers", etc. If the user is a graduating student of computer science and is looking for jobs, (s)he may choose the suggestion "careers", upon which the search engine issues the new query "computer science careers" and presents a new set of results to the user that are more relevant to this specific need.

Note that QR suggestions are probably meaningful for only 'topical' queries like the ones cited in table 1. This technique may not be applicable for a known-item finding query such as "Microsoft research" or "MIT home page", etc. In the rest of this work, we assume that we are dealing with topical queries for which providing QR suggestions is meaningful. No assumption is however made on the sub-topical structure underlying a query's topic. The topic could have a hierarchical structure instead of a flat one, but the QR suggestions reveal only the sub-topical structure at the next level

#	Query	Refinement suggestions	Source
1	Computer Science	Research; Careers; Definition Topics; Jobs; History; Projects	www.ask.com
2	Diabetes	Treatment; Tests/diagnosis; Symptoms; For patients Causes/risk factors; For health professionals; Alternative medicine	www.google.com
3	Java	virtual machine; applet; tutorial; applications; Indonesian; island; juice; language	www.yahoo.com

Table 1: illustrative examples for query refinement suggestions

each time. In the above example, if the sub-topic “jobs” has a further topical structure beneath it such as “software”, “hardware”, “administrative”, etc., this could be revealed by the new QR suggestions when the user chooses the QR suggestion “computer science jobs” at the first level. Thus, each time when a QR suggestion is chosen by the user, (s)he is descended to the next level in the topic hierarchy. This is similar to a tree-search making it quick and efficient for the user to locate the specific sub-topic that (s)he is interested in.

1.1 QR suggestions and Query expansion

It is important to note the distinction of QR Suggestions from automatic query expansion [8, 9, 4] which is a technique to expand the query with related words to boost retrieval performance. While QR suggestions aim at narrowing down the focus of the user’s search by presenting specific aspects of the original topic to the user, query expansion aims at improving the overall recall of the relevant documents in the ranked list. While in QR, the user is expected to choose one of the alternative refinement suggestions, query expansion requires no user intervention. Additionally, query expansion addresses synonymy problem of queries well, but is not as effective in addressing the polysemy problem while QR suggestions can effectively address both the problems. For example, query expansion may be able to retrieve documents that contain *automobile* for a query on *cars*, thus handling the problem of synonymy. But when the query is polysemous such as *java*, it may add both *coffee* and *programming* as expansion terms to *java*. In contrast, QR suggestions can distinguish between *coffee* and *programming* aspects of *java* by providing them as separate suggestions.

1.2 Objectives and Motivation

In this work, we are primarily concerned with measuring the quality of Query Refinement suggestions. Some work has also been done on evaluating QR suggestions, but mostly involving expensive user studies. For example, Anick [1] used query logs of users to demonstrate that QR suggestions can be useful to the users. He also studied the effectiveness of QR suggestions by examining certain user indicators such as percentage of sessions ending in a click in the ranked list, whether or not a QR suggestion is selected, etc. However, these evaluation measures can be quite expensive since it involves user interaction.

Another work on evaluation that has similar objectives to the present work is that of Zhai *et al* [10] which evaluates the ability of an IR system to retrieve documents that cover many different sub-topics under a given query’s topic. Their evaluation generalizes the traditional precision and recall metrics by accounting for intrinsic sub-topicality as well as redundancy in documents. This work differs from ours in the subject of interest: while the former work evaluates the ability of the ranked list to cover all sub-topics within a query’s topic, we are interested in measuring the ability of QR suggestions to cover all sub-topics of the query’s topic.

A closely related problem to sub-topic retrieval, sometimes called ”aspect retrieval”, is investigated in the interactive track of TREC, where the purpose is to study how an interactive retrieval system can best support a user in gathering the information about different aspects of a topic [7]. Again, this work consisted of user studies rather than any objective evaluation metric.

As far as evaluation of the quality of QR suggestions is concerned, we are not aware of any work that proposes an objective evaluation metric that does not involve expensive and time-intensive user studies. We believe an objective measure is very vital for the research community not only for repeatability of experiments but also for comparison of various techniques proposed for QR suggestions and further development of newer techniques.

The rest of the paper is organized as follows: We present our new evaluation metrics in section 2. Section 3 lists a few simple QR suggestion techniques that we considered for evaluation while section 4 presents the results of our experiments. In section 5, we present some discussion on the limitations of the new measures and map out directions for future work.

2. EVALUATION MEASURES

In this work, we propose two objective measures for evaluating the quality of QR suggestions. The first measure, known as Maximum Matching Averaged Mean-Average-Precision (MM-AMAP) requires labeled documents for the sub-topics underlying the query’s topic. The second measure which we call Distinctness and Mean-Average-Precision based F1 (DMAP-F1) requires only labeled documents that are relevant to the original query.

Our evaluation measures are based on the idea that QR suggestions will be most effective when the suggestions reveal the underlying sub-topical structure of the query’s topic. For example, in the example query “Computer Science” in table 1, the QR suggestions “jobs” “departments”, “research areas”, “companies” reveal the sub-topic structure of the broader “computer science” topic. The user can narrow down his(her) search by choosing one QR suggestion S, say “jobs”, from the set of QR suggestions $\mathbf{S}$, upon which the system retrieves a set of documents $\mathbf{D}_S$ by appending S to the original query, as in “computer science jobs”.

Following this intuition, the key idea behind our evaluation measures can be described as follows:

The quality of the QR suggestions can be measured objectively by the their retrieval effectiveness w.r.t. the sub-topic it represents, when used as queries.

Thus, the optimality of a QR suggestion S for a given query Q, can be measured by quality of the corresponding retrieval set $\mathbf{D}_S$ of the expanded query, quantified by Mean Average Precision (MAP) w.r.t. its sub-topic T_S represented by S. This is illustrated graphically in figure 1.

Since we do not know a priori which QR suggestion the user would click on, we assume the user has equal probability of choosing each QR suggestion. As a simple evaluation measure, one can maximize the expected retrieval effectiveness over all the sub-

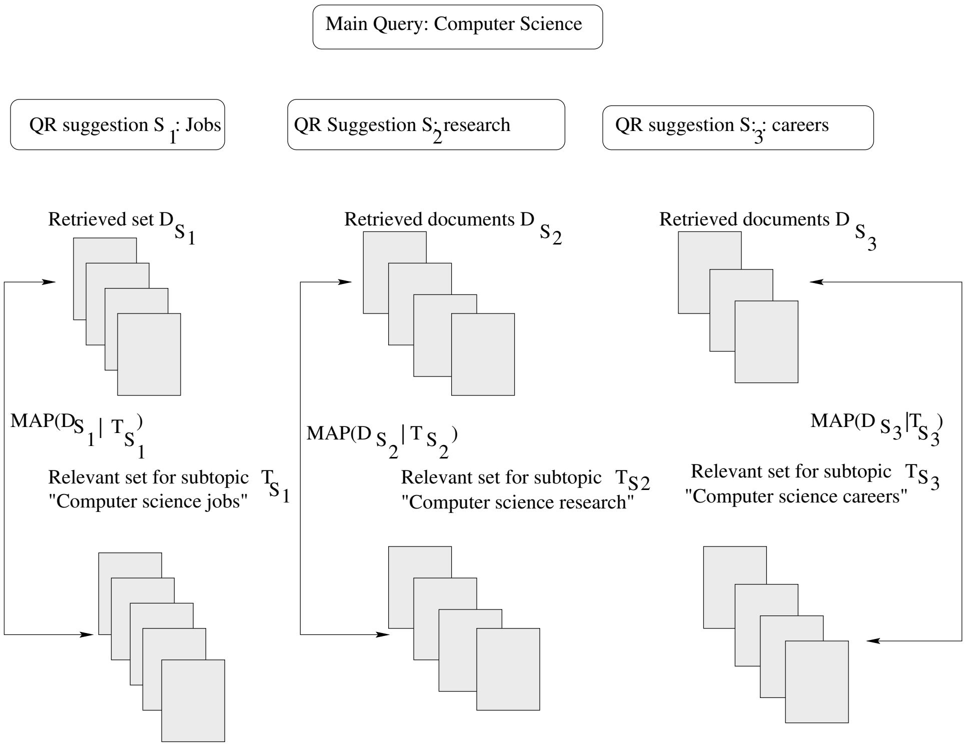

Main Query: Computer Science
QR suggestion S 1: Jobs
QR Suggestion S 2: research
QR suggestion S 3: careers
Retrieved set D S 1
Retrieved documents D S 2
Retrieved documents D S 3
MAP(D S 1 | T S 1)
MAP(D S 2 | T S 2)
MAP(D S 3 |T S 3)
Relevant set for subtopic T S 1
"Computer science jobs"
Relevant set for subtopic T S 2
"Computer science research"
Relevant set for subtopic T S 3
"Computer science careers"

Figure 1: **Evaluating QR suggestions by using them as queries and measuring the IR effectiveness of the corresponding retrieved set of documents w.r.t. respective sub-topics**

topics as shown below:

$$\text{AMAP} = \frac{1}{|\mathbf{S}|}\sum_{S\in\mathbf{S}} MAP(\mathbf{D}_S|T_S) \qquad (1)$$

where $MAP(\mathbf{D}_S|T_S)$ indicates the mean average precision of the retrieved document set $\mathbf{D}_S$ w.r.t. the sub-topic of the suggestion S. Thus, the new evaluation measure AMAP (Averaged MAP) is the average of the MAPs of the each QR suggestion w.r.t. its corresponding sub-topic. To compute this measure, it is evident that we need relevance judgments for the sub-topics underlying the query.

2.1 Maximum Matching Averaged MAP

An assumption that the above evaluation measure makes is that the correspondence between each QR suggestion S and its sub-topic T_S is known. In reality, this information is not available. Besides, the suggestions automatically generated by the system may not even capture the exact sub-topic structure underlying the user's query. Complicating the matters further is the fact that the number of QR suggestions generated by the system $|\mathbf{S}|$ may not be equal to the actual number of sub-topics $|\mathbf{T}|$ underlying a given query. We propose a greedy bipartite matching algorithm which computes AMAP by assigning retrieval sets $\mathbf{D} = \{\mathbf{D}_{S_1}, \cdots, \mathbf{D}_{S_m}\}$ corresponding to the QR suggestions $\mathbf{S} = \{S_1, \cdots, S_m\}$ to true set of topics sub-topics $\mathbf{T} = \{T_1, \cdots, T_n\}$.

The algorithm is described in table 2 and is graphically illustrated in figure 2 and works as follows. We first define a complete weighted bipartite graph between U, the set of nodes denoting the retrieval sets $\mathbf{D}$ and V, the set of nodes representing the sub-topics $\mathbf{T}$ where the weight of each edge e_{ij} is given by the average precision of the retrieval set $\mathbf{D}_i$ w.r.t. the sub-topic $\mathbf{T}_j$ (step 1 in table 2. Next, we iteratively pick each edge e_{uv} that has the maximum weight and assign the corresponding retrieval set $\mathbf{D}_u$ to the topic $\mathbf{T}_v$. We remove all other edges that connect to these nodes each time. The greedy assignment is complete when there are no edges remaining in the graph. We then sum the assigned edge weights and divide the sum by the larger of the initial number of retrieval sets or sub-topics to obtained the evaluation measure which we call Maximum Matching AMAP (MM-AMAP). The larger value is chosen to penalize the system if it provides too few or too many suggestions than the number of actual sub-topics.

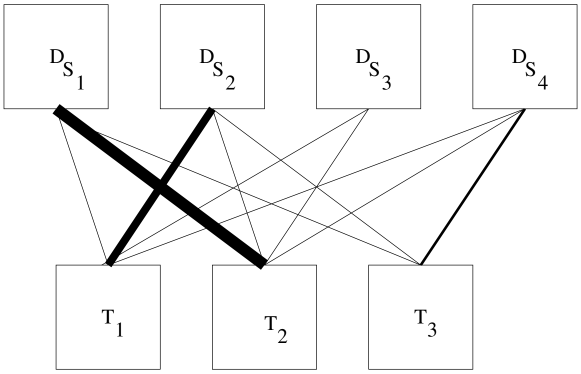


Figure 2: **Maximum Matching AMAP algorithm: the edge thickness represents its weight**

2.2 Distinctness and MAP based F1 (DMAP-F1)

```
Define a fully connected weighted bipartite graph $(U, V, E)$ where
$U = \{\mathbf{D}_{S_1}, \cdots, \mathbf{D}_{S_m}\}$ and $V \equiv \mathbf{T} = \{T_1, \cdots, T_n\}$ and
$E = \{e_{ij} = (\mathbf{D}_{S_i}, T_j) \mid \mathbf{D}_{S_i} \in U \;\&\; T_j \in V,\ W(e_{ij}) = MAP(\mathbf{D}_{S_i}|T_j)\}$.
MM-AMAP $\leftarrow 0$
while $E \neq \{\}$
    $e_{uv} = \arg\max_{e_{ij} \in E} W(e_{ij})$
    MM-AMAP $\leftarrow$ MM-AMAP $+ W(e_{uv})$
    $E \leftarrow E - \{e_{uj} | T_j \in V\} - \{e_{iv} | D_{s_i} \in U\}$
MM-AMAP $\leftarrow$ MM-AMAP$/\max(m, n)$
```

Table 2: A greedy Maximum matching algorithm for evaluating QR suggestions

Most of the standard TREC collections do not have any judgments for sub-topics underlying the queries. Hence MM-AMAP cannot be used as an evaluation technique on these collections. In this subsection, we present a new measure relaxes the requirement for sub-topic judgments. This new measure is based on the following premise: Since retrieved sets of documents $\mathbf{D} = \{\mathbf{D}_{S_1}, \cdots, \mathbf{D}_{S_m}\}$ corresponding to a QR suggestions $\mathbf{S} = \{S_1, \cdots, S_m\}$ are expected to represent distinct sub-topics, they should have as little overlap between them as possible. Additionally, since all the sub-topics are part of query's main topic, each retrieval set should also capture the main topic as much as possible. This premise allows to us to evaluate the quality of QR suggestions as follows:

1. For each retrieval set $\mathbf{D}_{S_i}$, compute MAP($\mathbf{D}_{S_i}|\mathcal{T}$), the MAP w.r.t. the query's main topic $\mathcal{T}$.

2. Compute AMAP $= \frac{\sum_i \text{MAP}(\mathbf{D}_{S_i}|\mathcal{T})}{|\mathbf{S}|}$, where $|\mathbf{S}|$ is the total number of QR suggestions.

3. Compute the distinctness ratio at rank $DR(R)$ between the retrieval sets as the ratio of the number of documents that occur in top R documents in exactly one of the retrieval sets to the total number of unique documents in all the retrieval sets put together. The mean distinctness ratio MDR is then given by:

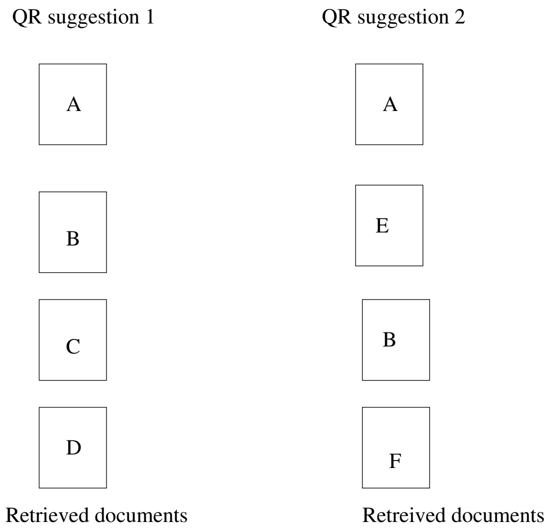
$$\text{MDR} = \frac{\sum_{i=1}^{10} DR(100 * i)}{10} \tag{2}$$

   An example computation of $DR(R)$ is shown in figure 3. MDR averages the distinctness ratio at top 100 documents through top 1000 documents in steps of 100 documents. This measure is inspired by mean average precision (MAP) and accounts for the fact that maintaining distinctness at the top of the ranked lists is more important than at the bottom.

4. Return DMAP-F1 $= \frac{2 \times (\text{AMAP}) \times (\text{MDR})}{(\text{AMAP}) + (\text{MDR})}$.

Thus DMAP-F1 captures retrieval effectiveness of each retrieval set w.r.t. to the main topic $\mathcal{T}$ as well the distinctness of each retrieval set w.r.t. one another.

Note that distinctness of retrieval sets does not necessarily guarantee that each retrieval set captures a unique sub-topic. This situation is illustrated in the set-representation of retrieval sets D_{S_1} and D_{S_2} and sub-topics T_1 and T_2 in figure 4. In this example scenario, although the retrieval sets are distinct (no overlap) and they together cover the query's main topic T, they fail to capture the exact sub-topic structure T_1 and T_2 of the main topic. This is clearly a shortcoming of the evaluation metric. However, in our experiments, we demonstrate that it works well empirically as shown by the strong correlation between MM-AMAP and DMAP-F1.



Distinctness ratio at rank 2 = $\frac{\#\text{(B,E)}}{\#\text{(A,B,E)}}$ = 0.67

Distinctness ratio at rank 4 = $\frac{\#\text{(E,F,C,D)}}{\#\text{(A,B,C,D,E,F)}}$ = 0.67

Figure 3: **An example computation of distinctness ratio**

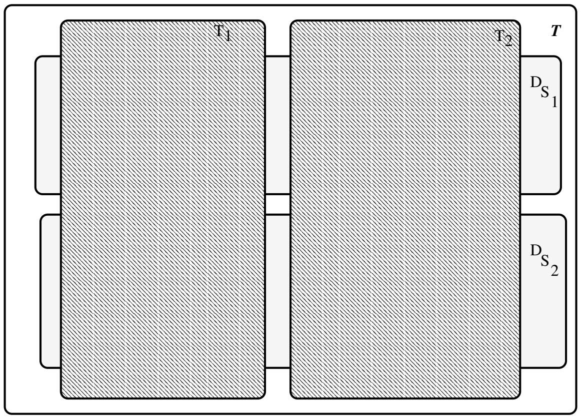


Figure 4: **Potential problem in the DMAP-F1 evaluation: The retrieval sets corresponding to QR suggestions may capture clusters different from the actual topical structure but may never be noticed by the evaluation**

3. TECHNIQUES

Several techniques have been proposed to generate QR suggestions automatically. The early work on QR suggestions can be traced back to Anick and Tipirneni [2] where they use different terms that the query words occur within certain syntactic constructions such as “adjective, noun, noun”, etc. as QR suggestions. Other techniques also include suggesting hyponyms (e.g.: birds of prey / falcons), morphological variants (e.g.: norse myth/ norse myths), acronyms (e.g.: USA / United States of America), etc. [1].

In a work that is similar to the technique of QR suggestions, Sanderson and Croft [6] came up with a technique based on subsumption relationships between terms to derive a concept hierarchy for each query. Lawrie’s work on hierarchical summarization [5] also comes quite close to our objective here. She provided a language modeling based framework to choose good topic words and then create hierarchy that provides a summary of the underlying collection of documents.

In this work, all the techniques we considered in our experiments extract terms from the top ranking documents from the initial retrieval based on the user’s original query. Our term selection is primarily based on statistical term weighting and ignores all linguistic information.

The main reason for considering simple techniques is to provide a sanity check to our evaluation measures. The techniques we present below are very intuitive and each technique overcomes potential flaws in the earlier techniques in the order presented below. Thus we expect intuitively sounder techniques to perform better on our evaluation measures. This could serve as an informal verification of the soundness of the evaluation measures. We would like to emphasize that the main contributions of this paper are the evaluation measures and not the techniques. In the future we would like to experiment with other existing techniques for QR suggestion using our new evaluation measures.

1. **TF-IDF:** In this technique, we sort the terms in the top 1000 documents from the original query by their TF-IDF weights given by:

$$\begin{aligned} TF(t,D) &= \frac{TF_{Raw}(t,D)}{TF_{Raw}(t,D) + 0.5 + 1.5 * \frac{\text{DocLen}(D)}{\text{AvgDocLen}}} \\ TF(t) &= \frac{1}{N_{1000}(t)} \sum_{D \in \text{ top 1000 Docs}} TF(t,D) \\ IDF(t) &= \log\left(\frac{N+1.0}{N(t)+0.5}\right) \\ \text{TF-IDF}(t) &= TF(t) \times IDF(t) \end{aligned} \qquad (3)$$

   where $N_{1000}(t)$ is the number of documents that the term t occurs in the top 1000 documents, $N(t)$ is the total number of documents in the collection that contain t, $TF_{Raw}(t,D)$ is the number of times t occurs in the given document D, $N(t)$ is the total number of documents in which the term occurs and N is the collection size.

   Given parameters n_t, the number of terms in each QR suggestion, n_s, the number of suggestions and the sorted array of terms $tvec$, the set of query refinement suggestions is given by: $S_i = \{tvec[(i-1)*n_t+1], \cdots, tvec[i*n_t]\}$. In other words the top n_t terms are used as the first QR suggestion, the next n_t words are used for the next QR suggestion and so on.

2. **C-TF-IDF:** The above technique makes no attempt to capture the sub-topic structure of the query’s topic in generating feedback suggestions. This technique rectifies the drawback by first clustering the top 1000 documents. We use a simple online clustering algorithm that determines cluster membership using TF-IDF weighted cosine similarity and a threshold T, where the clustering is done in the rank order of the documents. These clusters are assume to represent the sub-topic structure of the query’s topic. Given the parameter n_t as above, n_t top ranking TF-IDF weighted terms are extracted from each cluster as a QR suggestion. Thus there are as many QR suggestions as there are number of clusters, which in turn depends on the cluster threshold value n_t.

   Note that we used online clustering because it is one of the most computationally efficient clustering techniques and is therefore a suitable technique in this scenario where response time to the user is expected to be as low as possible. Additionally, a threshold based clustering allows different number of clusters for different queries(topics) depending on the inter-document similarity of the topic, providing higher flexibility than a clustering algorithm that fixes the number of clusters *a priori*.

3. **C-TF-IDF-ICF:** This technique aims to improve the distinctness of the retrieval sets of the QR suggestions by adding an extra-weight to terms which we call the Inverse Cluster Frequency weight as shown below:

$$\text{C-TF-IDF-ICF}(t) = TF - IDF(t) * \log\left(\frac{N_c+1}{N_c(t)+1}\right) \qquad (4)$$

   where N_c is the number of clusters and $N_c(t)$ is the number of clusters the term occurs in.

   Similar to the IDF weight, it down-weights terms that occur in all the clusters and prefers terms that are unique to the given cluster.

4. **C-TF-IDF-ICF-RW:** The previous algorithms do not differentiate between terms that occur in documents at the top of the ranked list and those that occur at the bottom of the rank list. This algorithm is same as the last one except in that it adds an extra rank weight (RW) equal to the okapi score of the document in the original retrieval, that scores terms from high ranking documents higher than the ones from the low ranked ones. In other words, the new weight of a term in a document TF-RW(t,D) is computed from $TF(t,D)$ formula in equation 3 as follows:

$$\text{TF-RW}(t,D) = TF(t,D) \times \text{okapi-score}(D,Q) \qquad (5)$$

   The objective is to eliminate terms that may have high TF-IDF weights but may be unrelated to the query’s topic.

5. **Upper-bound:** In this case, each query is provided with exactly as many QR suggestions as the number of actual sub-topics. In addition, the sub-topic descriptions provided by TDT annotators are used as QR suggestions. This is clearly an artificial scenario, but provides an estimate of the performance of the best possible system.

4. EXPERIMENTS

4.1 Data

The standard TREC data collections do not contain judgments for sub-topics. The Topic Detection and Tracking [1] corpus on the

[1] http://www.nist.gov/speech/tests/tdt/

other hand, has this desirable property of a two level topic structure and hence we used this in our experiments.

The TDT corpora contain news stories from multiple sources such as audio, video and news wire from multiple sources such as CNN, New York Times, ABC News etc., and multiple languages such as English, Mandarin and Arabic. When the source is non-English, machine translation output to English is available. When the source is audio or video, manual transcription feeds or automatic speech recognition output are used. We used manually transcribed, machine translated sections of the TDT3 and TDT4 corpora which contain 101,765 and 98,245 documents respectively.

The top level of the two-level hierarchical topical structure of TDT corpus is called Rules of Interpretation (ROI) categories which contains broad categories like “Acts of War”, “Celebrity news”, “Elections”, etc. Under each ROI category there are several topics and labeled documents on these topics are made available. For example, under the ROI category “Acts of violence and war”, there are topics such as “Bogota Plane hijacking”, “Palestinian child killed in cross-fire”, “Car bombings in Spain” etc. We excluded the “Miscellaneous” ROI category from each corpus since it contains unrelated topics and is thus not a good representation of a topical structure.

We considered each ROI category as our query’s main topic and the topics under each ROI as our sub-topics under the query. Henceforth, we will refer to each ROI category as ROI topic and the TDT topics as our sub-topics. When a query is issued on the ROI topic “Acts of violence and “War”, the QR suggestions are expected to reveal the sub-topic underneath it such as “Bogota Plane hijacking”, “Palestinian child killed in cross-fire”, etc.

In all, we have 10 ROIs from TDT3 corpus and 11 ROIs from TDT4 corpus. Under the ROIs we chose to use, there are 80 sub-topics in TDT3 and 65 sub-topics in TDT4 corpus that have judged documents. The number of sub-topics per each ROI topic ranges from 2 to 21 in TDT3 corpus and from 2 to 17 in TDT4 corpus.

The TDT corpus is primarily built for event based organization of news stories. Although it contains topic judgments which we can use for evaluation, it doesn’t contain any queries for each ROI. The ROI titles such as “Acts of violence and war” are too general to be used as queries since relevant documents may not contain those exact words. hence we generated artificial queries for each ROI by extracting top 8 TF-IDF terms from the union of judged relevant documents of all sub-topics under each ROI topic.

4.2 Results

We indexed the collection using the *Lemur* software. We did stopping using a standard stop-list and stemming using K-stemmer. We built *Lemur* APIs for all our algorithms listed in section 3. We used the okapi retrieval method for basic retrieval for our our original query as well as the retrieval for QR suggestions (needed for evaluation). The okapi TF-IDF weight for a query term t in a document D is given as follows:

$$TFIDF(t) = \frac{TF_{Raw}(t,D)}{TF_{Raw}(t,D) + 0.5 + 1.5 * \frac{\text{DocLength}(D)}{\text{AvgDocLength}}} \times \log\left(\frac{N+1.0}{N(t)+0.5}\right) \quad (6)$$

For all our algorithms, we did a two-fold cross validation: We optimized the parameters of each algorithm by maximizing the objective function (MM-AMAP) on one corpus and its corresponding set of queries and tested the algorithms on the other corpus and its corresponding queries, with the parameters set at these optimal values. We then switched the training and test sets and repeated the same process. The results of the two test sets are then merged to obtain 21 data points corresponding to 21 ROI queries.

The results of our experiments are presented in table 3. Results show consistent trends between the evaluation measures as well as a strong statistical correlation in terms of Pearson correlation coefficient (that ranges between -1 and +1; values ¿ 0 indicate positive correlation). Also notice that each successive algorithm performs better than the previous one on both measures, informally validating our measures since each successive algorithm overcomes the flaws in the previous one.

5. CONCLUSIONS

In this paper, we presented two IR based, objective two evaluation measures to estimate the quality of query refinement suggestions. While the first one MM-AMAP relies on the availability of sub-topic judgments for the query’s main topic, the other measure eliminates this requirement by estimating the distinctness of the retrieval sets w.r.t. one another. The correlation between the two measures is established not only by the Pearson test, but also by the consistency of results between the two measures.

One of the limitations of the current work is the relatively small number of queries (21) that we performed the experiments on. This is mainly due to the non-availability of sub-topic judgments in standard research collections. The second evaluation metric DMAP-F1 address precisely this issue and eliminates the need for sub-topic judgments, allowing us to perform experiments on larger number of queries using TREC collections and judgments in the future.

Another important limitation of the current evaluation measure is that it fails to take into account user experience. A QR suggestion can be effective in retrieving documents on a sub-topic when used as a new query, but it is not of much use if the user does not comprehend it. For example, given a query “information retrieval”, a QR suggestion such as “spider robot” may be effective in retrieving documents on the sub-topic “automatic crawling and indexing” and hence may be rated high by our evaluation metric. But it is not clear if this suggestion would help user understand the subject sub-topic. Hence one of our future plans is to measure the correlation between our measures and user satisfaction and also to develop predictors of user satisfaction using objective evaluation metrics.

We also intend to perform more extensive experiments comparing other published QR suggestion techniques in the future.

Acknowledgments

We are extremely thankful to Prof. James Allan and Prof. Bruce Croft for their invaluable suggestions on this work. This work was supported in part by the Center for Intelligent Information retrieval and in part by the Defense Advanced research projects Agency (DARPA) under contract number HR0011-06-C-0023 and in part by NSF grant #IIS-0527159. Any opinions, findings and conclusions or recommendations expressed in this material are the author(s)’ and do not necessarily reflect those of the sponsor.

6. REFERENCES

[1] Peter Anick. Using terminological feedback for web search refinement - a log-based study. In *Proceedings of the 26th annual international ACM SIGIR conference*, pages 88–95, 2003.

[2] Peter Anick and Suresh Tipirnenit. Interactive document retrieval using faceted terminological feedback. In *Proceedings of the 32rd Hawaii International Conference on System Sciences*, 1999.

Table 3: Performance comparison of various algorithms on both the evaluation measures over 21 queries on TDT3 and TDT4 corpora: bold faced numbers indicate statistically significant difference of the corresponding algorithm w.r.t. the algorithm in the row above, in terms of a two-tailed paired T-test at 95% confidence. The upper bound results for DMAP-F1 are not available at the time of submission, but will be made available upon acceptance.

Technique/Evaluation	**MM-AMAP (%)**	**DMAP-F1 (%)**	Pearson Corr.
TF-IDF	13.06	34.12	0.48
C-TF-IDF	14.20	37.87	0.21
C-TF-IDF-ICF	14.55	**43.65**	0.39
C-WTF-IDF-ICF	14.61	**46.32**	0.53
Upper-bound	20.75	N.A.	N.A.

[3] Scott Henninger and Nicholas Belkin. Interface issues and interaction strategies for information retrieval systems. In *Proceedings of the Human Factors in Computing Systems Conference (CHI)*. ACM Press, New York, 1996.

[4] V. Lavrenko and W.B.Croft. Relevance-based language models. In *Proceedings of the 24th annual international ACM SIGIR conference*, pages 120–127, 2001.

[5] Dawn Lawrie. Language Models for Hierarchical Summarization (Ph.D. dissertation). Technical report, UMass Amherst, 2003.

[6] Mark Sanderson and Bruce Croft. Deriving concept hierarchies from text. In *Proceedings of the 22rd annual international ACM SIGIR conference*, 1999.

[7] E. Voorhees and D. Harman, editors. *Proceedings of Text REtrieval Conference (TREC-10)*. NIST Special Publications, 2001.

[8] E. M. Voorhees. Query expansion using lexical-semantic relations. In *Proceedings of the Seventeenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 61–69, 1994.

[9] Jinxi Xu and W. Bruce Croft. Query expansion using local and global document analysis. In *Proceedings of the Nineteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 4–11, 1996.

[10] Cheng Zhai, William W. Cohen, and John Lafferty. Beyond independent relevance: Methods and evaluation metrics for subtopic retrieval. In *Proceedings of the 26th annual international ACM SIGIR conference*, pages 10–17, 2003.