Scribes: Gilly Leshed, N. Sadat Shami

Outline

1. Review: Relevance Feedback
2. Interactive Query Expansion (IQE)
3. AQE vs. IQE
4. Active Relevance Feedback
5. Evaluation of Relevance Feedback
6. Motivation for Implicit Relevance Feedback

1 Review: Relevance Feedback

Explicit Relevance Feedback (RF) is the process in which the user marks documents initially retrieved by an IR system as relevant or not. The system then uses this information to retrieve a better set of results for the user's query. In the previous lecture we discussed the following ways to use RF information:

- *Term Re-weighting*: Weights of terms that appear in the query are adjusted as a result of the RF information, and the document scores are re-calculated. For instance, a query term that appears in a judged relevant document will most likely receive increased weight.
- *Query Expansion (QE):* A process in which terms are added to the query as a result of the examination of judged relevant documents. We examined how, methodologically, QE can be applied to the VSM, probabilistic model, and language model. These methods assume that the system *automatically* expands the query given the RF information. We can therefore refer to these techniques as *Automatic Query Expansion* (AQE).

  However, AQE can be confusing to users, as they do not necessarily understand how the system works: the underlying mechanism of RF and AQE is not transparent to users [1].

2 Interactive Query Expansion (IQE)

To clarify the concept of query expansion, instead of applying it "behind the back" of users[1], we can ask the user's permission to add terms to the query, basing the suggested terms on the RF information. This is the idea of *Interactive Query Expansion* (IQE), of which the process is:

- Given RF, re-rank terms and list them in descending order
- Ask user which terms from the list to add to the query

2.1 IQE Advantages

- User more in control of the process
- It makes more use of the user's knowledge
- Additional feedback, beyond document relevance feedback, is provided about terms

2.2 IQE Disadvantages

[1] The following exercise could be applied to check if a certain IR system is applying AQE: (1) type a query; (2) given the results, guess which terms could have been added to the query; (3) type the query with the conjectured terms. If the results of (1) and (3) match, it is possible that the system is applying AQE.

- Extra work for the user
- Poor user choice of which terms to add could produce inferior results.
  From a user-centered perspective we need to consider what it means for a user to make a "poor choice". The user could (1) misunderstand the meaning of terms, or (2) misjudge which terms would be useful for the system to enhance the number of relevant documents retrieved. Additionally, as mentioned in a prior lecture, perhaps the interface used for searching/formulating a query does not follow the mental model of certain users. For example, a user can easily judge whether a document is relevant or not, but most users have problems formulating effective query terms.

3 AQE vs. IQE

Two complementary ways can be considered to examine which approach is superior, AQE or IQE: user preference and system performance.

3.1 User Preference: Fowkes and Beaulieu, 2000

The degree to which users prefer AQE or IQE depends on how well they perceive the system is performing in either approach. Fowkes and Beaulieu [3] found that AQE seemed more effective to users for simpler search tasks whereas IQE appeared more productive for more complex search tasks.

The concept of delegating the control back to the user in IQE can also play a role in user preference. Users might evaluate the results of IQE as more valuable than AQE because it keeps them in the loop.

3.2 System Performance: Ruthven, 2003

In a paper that won the Best Paper Award in SIGIR 2003, Ruthven described a way to compare the performance of AQE with that of IQE [4]. He looked at three AQE concepts, namely, collection independent, collection dependent, and query dependent.

3.2.1 Simulating IQE

First, Ruthven describes a method in which IQE was simulated, over a total of 99 queries in three separate collections:

For each query,

1. The documents were initially ranked using VSM.
2. A list of 15 possible expansion terms was obtained from user-judged relevant documents.
3. All 32678 combinations of expansion terms were created, simulating all possible IQE decisions that a user could make.
4. Each combination of expansion terms was separately added to the original query and the documents were re-ranked using VSM.
5. For each of the 32678 versions of query expansion results, recall-precision performance measures were calculated.

These 32678 possible IQE versions were then sorted by performance, providing the best performing IQE decision at the top and the worst performing IQE decision at the bottom.

3.2.2 Query Expansion vs. No Query Expansion

One way to compare AQE with IQE is to show which of them improves more over the "no query expansion" results, in terms of the percentage of queries that were improved by these strategies. On average, AQE was more likely to improve the results of a query than to harm it, with the query dependent approach performing better and the collection independent approach performing worst. However, the best performing IQE decision for each query improved the queries in a much higher percentage than all AQE

approaches. Ruthven suggests that this only represents a potential benefit, as there is no guarantee that the user can easily select the best performing set of terms to add to the query.

3.2.3 AQE vs. IQE

To address the question of the potential of IQE decisions to perform better than AQE, Ruthven compared how many IQE decisions actually performed better than AQE. In all, depending on the corpus, only 9-12% of possible IQE decisions performed significantly better than the corpus independent approach, which can be considered as the most realistic approach. This implies that it may be hard for users to select the set of terms that would expand the query to produce superior results.

In a pilot study using three human participants, users were presented with 15 expansion terms for each query. For each term, users were asked whether adding it to the query would be useful or not in enhancing the retrieval of relevant documents. Among the three users and the three different corpora, the results ranged from 32-73% in detecting the "good" expansion terms, those terms that improve the results when added to the query.[2]

Ruthven also suggests that good expansion decisions are dependent upon the instructions on how to operate IQE. He shows that some simple instructions, such as 'select more terms', 'trust the system', and 'use semantics' would not necessarily result in making better expansion decisions.

4 Active Relevance Feedback: Shen and Zhai, 2005

Another issue in explicit RF is how to choose the documents to be presented to the user for relevance feedback, such that the system will learn most from the RF information. Shen and Zhai [6] suggest that the traditionally selected top-K documents lack diversity, as they are all selected based on similarity to the query. As a result, the system does not gain much from presenting the top-K documents to the user for RF. For example, providing RF about two distinct documents is of more value to the system than providing RF about two identical documents.

Adding documents that are not from the top-K ranked list should be done carefully, as it increases the likelihood of presenting non-relevant documents to the user. The implications could be:

- A decrease in learning from RF *by the system* if too many documents are judged as non-relevant.
- The *user experience* with the system could be degraded.

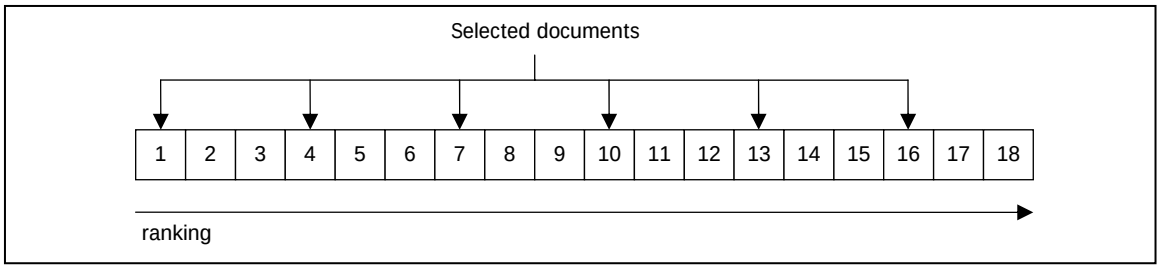
Shen and Zhai proposed Gapped Top-K and Cluster Centroid as two approaches for selecting documents, which do not necessarily display documents from the top-K ranked list. Both approaches assume that K documents to be presented are selected from the top-N documents, $N \geq K$ (top-K is obtained when N=K).

4.1 Gapped Top-K

Given $N=G \times K$, where G is a small positive integer, the list is sorted in a descending order by rank. The first document is selected. The next selected document is G documents away, and so on. In this way, K documents are selected, each located at least G documents away from each other.

In the following example, N=18, G=3, and K=6:

[2] Note that one should be wary in establishing conclusions from a user study with 3 participants, as statistical significance was not measured.



4.2 K Cluster Centroid

The top-N documents are partitioned into K clusters according to similarity among the documents. From each cluster, the centroid document is selected, such that it maximizes the average similarity between the chosen document and other documents in the cluster. This method allows selecting the most representative document from each cluster. Another choice that can be employed is selecting from each cluster the document with the highest ranking. The clustering method ensures diversity in the selection of documents for the RF process, assuming diversity in the N documents being clustered.

4.3 Results

Shen and Zhai show that both Gapped Top-K and K Cluster Centroid algorithms outperform the traditional top-K algorithm, with fewer judged relevant documents. They conclude that diversity in the presented documents is a desirable property for RF systems.

5 Evaluation of Relevance Feedback: Chang, Cirillo, and Razon, 1971

Evaluating RF systems using the conventional precision/recall scheme is biased because the RF system can place the documents judged by the user as relevant at the top of the list regardless of how the system updated the scores of these documents. This idea is known as the *ranking effect*. Chang et al describe three techniques that ameliorate the *ranking effect* weakness of RF evaluation, described in the following sections [2].

5.1 Residual Ranking

This technique examines how well the system does before RF against how well it does after RF. The effectiveness measure reflects the number of *newly retrieved relevant documents* as a result of the RF process - at every iteration, the evaluation takes into consideration only the remainder of the document collection, excluding the documents that were judged by the user.

Some issues of residual ranking:

- In small collections, ranking only the residual collection could be problematic as the set of judged documents changes relative to the original ranking. In very large collections this could be less of an issue, since the number of judged documents is considerably small. However, this could still be an issue even if the corpus is large. For instance, consider the case in which |R| = 1, and the one relevant document is retrieved before RF and marked in the RF process. As a result, after RF precision = 0.
- This method penalizes systems that do well on early iterations, since they don't have much to improve on in subsequent iterations.

5.2 Modified Freezing

The idea of this technique is to *freeze the rank of the judged relevant documents*, so that no matter where these documents are placed after the RF, their ranking for the evaluation is considered as if they were in

their position before the RF[3]. The freezing technique fixes the before vs. after comparison problem identified in the residual ranking technique, as we compute performance measures on the entire collection in all iterations.

One issue with the freezing method is that at each iteration a higher proportion of documents are frozen. As a result, slow systems that perform poorly in early iterations but better in later iterations are evaluated by the freezing method as performing more poorly due to the higher percentage of frozen documents [5].

One question that we need to ask ourselves in relation to RF evaluation is whether we are interested in evaluating overall performance, or the degree to which the system is learning from the RF process.

5.3 Split Corpus

The third technique includes splitting the collection randomly into two disjoint halves: the *test group*, upon which the query is modified through RF, and the *control group*, upon which the evaluation is carried out. This method fixes the problems identified in the previous techniques.

The main issue with this technique is splitting the collection: a random split will not necessarily end up with the relevant documents evenly split between the two groups. Also, the relevant documents in the test group are not necessarily representative of those in the control group [5]. This could result in a decline in the user experience with the RF system.

6 Motivation for Implicit Relevance Feedback

In this lecture we discussed various issues in RF, including the user (mis)understanding of how the system works, diversity of judged documents, and evaluation of the results. While these issues question the whole idea of RF, previous studies have shown that RF methods could be beneficial for IR systems by leading to better performance.

Still, the user experience with an RF system is troublesome, especially by asking the user to exert extra work only for the sake of one query's results. Implicit RF, which will be discussed in the next lecture, attempts to overcome this issue. The idea is to receive RF from observing the users' behavior without asking them to explicitly provide the relevance information.

References

[1] M. Beaulieu, H. Fowkes, N. Alemayehu, M. Sanderson. Interactive Okapi at Sheffield - TREC-8. *Proceedings of the 8th TREC conference (TREC-8)*, 689-698, 1999.

[2] Y. K. Chang, C. Cirillo, and J. Razon. Evaluation of feedback retrieval using modified freezing, residual collection & test and control groups. *The SMART retrieval system - experiments in automatic document processing*. G. Salton (ed). Ch 17. 355-370. 1971.

[3] H. Fowkes and M. Beaulieu. Interactive searching behaviour: Okapi experiment for TREC-8. *Proceedings of 22nd BCS-IRSG European Colloquium on IR Research. Electronic Workshops in Computing*. Cambridge. 2000.

[4] I. Ruthven. Re-examining the Potential Effectiveness of Interactive Query Expansion. *Proceedings of the Twenty-Sixth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 03)*. Toronto. 2003.

[5] I. Ruthven and M. Lalmas. A survey on the use of relevance feedback for information access systems. *Knowledge Engineering Review*. 18 (2). 95-145. 2003.

[3] Note that the *full-freezing* technique involves freezing the rank of all documents presented to the user, whereas in the *modified-freezing* technique only the ranks of the judged relevant documents are frozen.

[6] X. Shen and C. Zhai. Active Feedback in Ad Hoc Information Retrieval. *Proceedings of the Twenty-Eighth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 05).* Salvador, Brazil. 2005.

Question 1

In this class we considered two approaches to perform query expansion. In both approaches the user's query is expanded based on his or her input, provided as relevance feedback and/or term selection. Can you think of additional ways to perform query expansion?

Answer

The motivation behind relevance feedback, besides the fact that relevance is precisely the information current IR systems generally seek to capture, is that it is fairly easy for individual users to judge some documents as relevant or irrelevant for their query. We also know that it is hard for a system to generate good terms for query expansion automatically using techniques such as automatically generated thesauri for synonym generation [8]. Given that it is hard to automatically come up with terms for query expansion, could terms that thousands of users used for a particular query to find relevant documents be used to suggest terms to the user for expansion? This concept was first conceptualized by Malone et al. as 'social filtering' [5] and later adopted and implemented using different names such as 'collaborative filtering' [2, 4] and 'recommender systems' [6]. The basic idea behind this is to filter information based on aggregating individual judgments of quality through methods that learn about individual users and provide recommendations based on correlations of groups of users [5]. Perhaps there is similarity between the query terms different users used to find relevant documents. Once the user provides relevance feedback, the query term(s) used to obtain those relevant documents could be associated with a particular 'information need' and indexed. One apparent difficulty of this approach is in identifying these 'information needs', as they are psychological states that are associated with the desire to know the unknown, and that cannot be directly observed [9]. For the purposes of our algorithm, an 'information need' could be considered as a category or cluster of related terms. The strength of association between the query terms and the 'information need' category would dictate which other terms within that category will be presented for query expansion. Thus, when another user issues a query on the same 'information need', the terms other users found useful in finding relevant documents for that 'information need' could be suggested for inclusion in query expansion.

Like many machine learning approaches, this approach will hopefully improve over time, although as query terms and relevance information are added to the system it requires continually re-clustering the 'information need' categories in order to better define them until stabilization. There are two interface design goals that need to be incorporated to improve the user experience:

- Users should be told the origin of the suggested query expansion term(s).
- Users should be given some information about the power of the suggested query expansion term(s).

For example, with an original query of 'automobile manufacturer', users should be told that "9334 other users found using the query term 'car dealership' is helpful when you 'want to buy a car'".

This approach partly suffers from the main drawback of Interactive Query Expansion (IQE) in that the user has to exert additional effort in expanding the query beyond providing relevance information for documents. This leads us to Jonathan Grudin's question of 'who does the work and who gets the benefit' that is typically associated with interfaces of this nature [3]. However, we feel that it offers relatively more filtering value for relatively less filtering work. As the number of users grows, and correlations between different users allow them to be classified as a group of users seeking the same 'information need', the value of this approach to an individual user will grow.

Question 2

In class we talked about how the same query can represent different information needs. This is particularly problematic for relevance feedback. For example, relevance feedback for documents will differ substantially when the user wants to find information about 'Queen Elizabeth' the person compared to 'Queen Elizabeth' the ocean liner. Discuss a possible way of overcoming the problem of different information needs represented by the same query in order to improve relevance feedback.

Answer

A possible solution to this problem might lie in clustering documents within the search interface. Clustering is a classification method which logically organizes documents into categories according to their similarity. For example, the term 'mercury' returns the following clusters of documents on a clustering engine like vivisimo.com:

⇨Planet (18)

⇨Photo (16)

⇨Health (14)

⇨Dealer, New and used (13)

⇨Environment (13)

⇨Testing, Software (10)

⇨Manufacturer (7)

⇨Program (13)

⇨Solutions (10)

⇨God, Hermes (8)

We can obtain relevance feedback from documents within a cluster. The relevance feedback obtained would pertain to documents only in that particular cluster. In this way, we can capture the information need behind the user query. For example, if the user clicks on 'Planet' and expands that cluster, she will be presented with documents related to the planet Mercury. We would then be able to obtain relevance feedback on documents when the information need behind the user query is about finding information related to the planet Mercury. If the user clicks on 'Dealer, New and Used' and looks at documents after expanding that cluster, we would know that she is interested in the car Mercury. We would thus be able to obtain relevance feedback for documents when the information need behind the query is the car Mercury. This approach increases the amount of information that the system can receive from the user's feedback.

However, as Baldonado & Winograd mention, developing an understanding of a user's current information need is not simple [1]. Information needs are often fluid. Users move from one area of interest to another based upon what they find along the way. The concept of Ostensive Relevance presented in Ruthven and Lalmas's survey [7] attempts to mend this problem, by weighing heavier new RF information provided later in the RF process relative to old RF information from early stages of the process. Additionally, sometimes users do not have a clear information need in their search. They hope that the returned search results will help them to obtain a better information need. When users themselves are not certain about the purpose of their query, obtaining relevance feedback will be difficult.

References

[1] Baldonado, M.Q.W., and Winograd, T. SenseMaker: An information-exploration interface supporting the contextual evolution of a user's interests. Proc. CHI'97, ACM, 11-18.

[2] Goldberg, D. Nichols, D., Oki, B. M., and Terry, D. Using collaborative filtering to weave an information tapestry. *Communications of the ACM*, 35 (12), 61-70, 1992.

[3] Grudin, J., Social Evaluation of the User Interface: Who Does the Work and Who Gets the BENEFIT? *Proceedings of IFIP INTERACT '87: Human-Computer Interaction*, 805-811, 1987.

[4] Konstan JA, Miller BN, Maltz D, et al. GroupLens: Applying collaborative filtering to Usenet news *Communications of the ACM*, 40 (3), 77-87, 1997.

[5] Malone, T.W., Grant, K.R., Turbak, F.A., Brobst, S.A. and Cohen, M.D. Intelligent Information Sharing Systems. *Communications of the ACM*, 30, 5, 390-402, 1987.

[6] Resnick, P. & Varian, H. Recommender systems. *Communications of the ACM*, 40 (3), 1997.

[7] Ruthven, I., and Lalmas, M. A survey on the use of relevance feedback for information access systems. *Knowledge Engineering Review*, 18(1), 2003.

[8] Singhal, A. Modern Information Retrieval: A brief overview. *Bulletin of the IEEE Computer Society Technical Committee on Data Engineering*, 24(4), 2001.

[9] Saracevic, T. Relevance: A Review of and a framework for the thinking on the notion in information science. *Journal of the American Society for Information Science*, 26(6), 321-343, 1975.