

Query Expansion Powered by Wikipedia Hyperlinks

Carson Bruce, Xiaoying Gao, Peter Andreae, and Shahida Jabeen

School of Engineering and Computer Science
Victoria University of Wellington
PO Box 600, Wellington, New Zealand
{brucecars,Xiaoying.Gao,Peter.Andreae,shahidarao}@ecs.vuw.ac.nz
<http://www.victoria.ac.nz/>

Abstract. This research introduces a new query expansion method that uses Wikipedia and its hyperlink structure to find related terms for reformulating a query. Queries are first understood better by splitting into query aspects. Further understanding is gained through measuring how well each aspect is represented in the original search results. Poorly represented aspects are found to be an excellent source of query improvement. Our main contribution is the way of using Wikipedia to identify aspects and underrepresented aspects, and to weight the expansion terms. Results have shown that our approach improves the original query and search results, and outperforms two existing query expansion methods.

1 Introduction

Search engines enable us to sift through billions of web pages allowing us to find required information over a manageable list of documents ordered by their relevance to the query. Translation from a search goal to a search query is a difficult task for the user, requiring a specialized understanding of the search engine's mechanism coupled with a strong linguistic proficiency to best express the search goal. Search engines treat a query as a character string to be compared with character strings in the searched document set. This interpretation of a query is limited in light of the original information need of the user, resulting in a semantic gap between the search query and the search goal. Query Expansion (QE) is a solution that addresses this problem. The idea is to reformulate a query by adding new terms to the query so that more relevant documents can be retrieved.

Wikipedia has become the largest and most accessed web-based encyclopedia with 19.8 million articles (3.7 million in the English Wikipedia). Each article contains information relevant to a focused concept. Within each article are hyperlinks, 60 million of them in total, referencing other article concepts to support its own contents. Through these hyperlinks, each Wikipedia article is heavily linked with other articles, resulting in a huge hyperlink structure. These hyperlinks contain useful information in the form of anchor text – the terms or phrases in the article to which hyperlinks are attached. These features make Wikipedia a

promising lexical resource for the problem of query expansion. The main focus of our research is to use Wikipedia as a knowledge source for supporting query interpretation and reformulation.

Section 2 of the paper discusses related research on query expansion. Section 3 describes our proposed query expansion approach. Section 4 explains our method of evaluation along with the results and discussion and section 5 summarizes the outcomes of this research, possible limitations and the future work directions.

2 Related Work

Web users that have used a search engine are likely to have been exposed to some form of query expansion system operating at the back end or providing visible suggestions for users to reformulate their queries manually. There is a large set of query expansion solutions summarized through multiple information retrieval surveys [2, 11, 18].

2.1 WordNet and Wikipedia Based Query Expansion

Many query expansion systems [10, 9, 1, 19] utilize WordNet [8], a lexical database for the English language, to find query expansion terms. WordNet resembles a thesaurus, classifying words into sets of cognitive synonyms. This classification can be used to find the synonyms of terms within a query. By using these synonyms to reformulate a query has shown to be effective to improve recall, but sometimes hurt precision.

The works of [15, 12, 16, 22, 1, 9, 19, 10] focus on using Wikipedia as its source for finding and weighing terms. This has values over WordNet and other web source as the Wikipedia source is large, dynamic, clean, objective, and is revolved around articles focused upon single concepts. A single article provides a valuable source of terms so long as the concept expressed in the article holds relevance to the sense of the query. These works concluded that a key to success is to use Wikipedia as a source for selecting articles most closely related to the query.

Koru [15], a knowledge-based web search engine, uses Wikipedia Link based Measure [13] for query expansion. Wikipedia Link-based Measure is a method of measuring semantic relatedness between two words. Suppose each word is linked to a Wikipedia article, it uses the hyperlinks shared between the two Wikipedia articles to find the semantic similarity between the two terms, as shown by the formula below:

$$sr(a, b) = \frac{\log(\max(|A|, |B|)) - \log(|A \cap B|)}{\log(|W|) - \log(\min(|A|, |B|))} \quad (1)$$

In the above equation, A and B are the set of articles that link to articles a and b , and W is the entire Wikipedia. Koru's search was based on an interactive query expansion system which identified related Wikipedia topics that users could select after finishing their queries. These selected topics mapped to article

concepts are used to identify candidate terms for the expansion. This interactive application is valuable in that it allows direct feedback from the user to be used to disambiguate the query often seen as providing an optimal query expansion solution [18]. Limitations come in the behavior of the users, very rarely taking the effort to provide relevance feedback to the system, preferring to simply reformulate their own queries alone [21, 3].

Our research is a Wikipedia-based approach and we also use Wikipedia Link based Measure [13] in our system. Our system is unique in that it is based on an idea of identifying underrepresented aspect of a query for query extension, which is introduced in our previous research on AbraQ [17, 7]. The rest of this section details AbraQ.

2.2 Aspect-Based Query Expansion: AbraQ

AbraQ [17, 7], is a query expansion system using a unique step named query aspect identification in their solution. This step divides a query into various term combinations that represent individual concepts contained. These combinations are termed Query Aspects. For example, a query "black bear attacks" has two aspects: "black bear" and "attacks". This step creates additional meta-data, providing a greater source of information for the expansion system to understand the query.

The AbraQ system can be divided into three distinct stages. First, the query is split into aspects based on global document analysis. This is done in two steps: subsequences of the query are executed as queries to retrieve their hit counts (total respective document number); the count of the focused aspect is measured relative to the counts of the individual terms of the aspect and the counts of the possible permutations of the aspect.

With the aspects defined, AbraQ looks to identify how well the aspects are represented in the original search results so it is able to identify the most under-represented aspect. Identifying the underrepresented aspects is done by building and comparing vocabulary models for each of the aspects and for the entire query. The vocabulary is generated from the initial results retrieved by the aspects (for aspect vocabulary) and query (for query vocabulary), counting the frequencies of the terms appearing in the documents. A strong similarity between the query vocabulary and a single aspects vocabulary suggests that the aspect is already well represented in the search results. The focus of the second phase is to find out the biggest difference in vocabulary models for identifying the most underrepresented aspect of the query.

The final step of the AbraQ method is the term selection for query refinement. The vocabulary model of the underrepresented aspect is used as a source of terms to extend the original query. The new query is then re-evaluated to see if the aspect representation is improved. The goal of this step is to generate a query whose aspects are all equally represented.

The AbraQ system uses web documents retrieved through web search engines as its source for aspect identification, vocabulary building, and term weighting functions. This has its limitations as the document results can be noisy,

populated with messy spam and advertising advocating individual companies. This research chose to use Wikipedia as a better alternative.

3 Query Expansion Method

This section will first detail our core method (Aspect Representation), and then outline its three variations (Relatedness, TFIDF, Combined).

3.1 Core Method: Aspect Representation

Aspect Representation is our core method further divided into four major steps. Figure 1. explains these four steps along with the supporting measurements used in each of them.

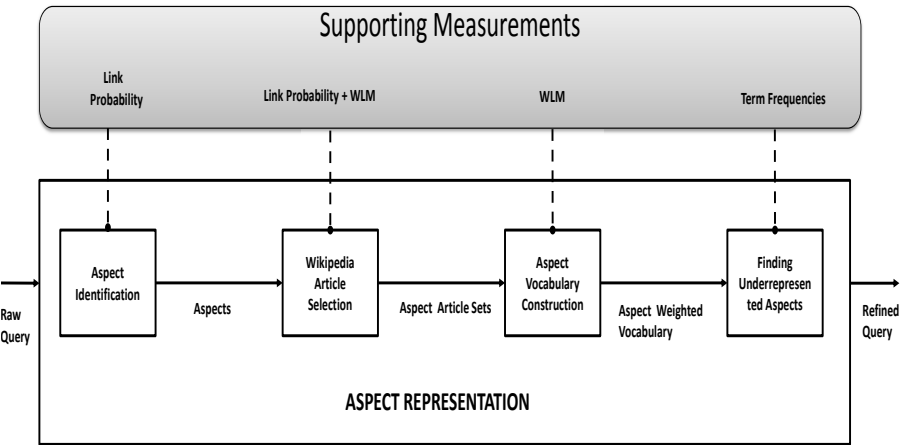


Fig. 1. Overview of Aspect Representation method along with the supporting measurements

An example of the whole QE process taking raw query “Discovery Channel Store” as input and generating reformulated query “Discovery Channel Store History” is shown in the Table 1.

Table 1. Process overview with an example query

Step	Output
1. Query	“Discovery Channel Store”
2. Aspect Identification	“Discovery Channel” + “Store”
3. Wikipedia Article Selection	[“Discovery Channel”, “Store”]
4. Aspect Vocabulary Construction	“history(Discovery Channel)”, “shops(Store)”
5. Finding Underrepresented Aspects	0.4(Discovery Channel), 0.6(Store)
6. Query Expansion	“Discovery Channel Store History”

Step 1: Aspect Identification. An aspect is a topic or concept which is a term or phrase appearing in the query that preserves the original ordering. Aspect Identification is the initial stage of our method, taking the original query as its input to give a set of query aspects. This added information helps support all subsequent stages of the system aiming to put boundaries between the single concepts that may exist in a multi-concept query. A query term can be used in more than one aspect. Identifying aspects within the query allows us to better source articles for our expansion terms than trying to source articles from the whole query alone.

We identify aspects based on their Link Probability [14] weighting. Link Probability is a proved feature for measuring 'keyphraseness'. It estimates the probability of a term to be selected as a hyperlink in a new document by counting the number of documents where the term was already selected as a hyperlink divided by the total number of documents where the term appeared. This probability can be interpreted as, the more often a term is selected as a hyperlink among its total number of occurrences, the more likely it will be selected again. We first identify each possible aspect of our query and list them in order of link probability weight. From this list we finalize a set of query aspects by following a set of selection rules:

- Aspects are selected from highest value through to the lowest.
- An aspect is ignored if it is a subset of an already selected aspect.
- No aspects with a weighting of 0 should be added unless they contain terms that are yet to be covered by the selected aspects.
- Aspect Identification is complete when each term of the query has been covered by an aspect.

Step 2: Wikipedia Article Selection. This phase creates a single set of Wikipedia articles with close connection to our query and its aspects defined by the previous stage. This article set is used as a source for extracting terms for query reformulation and building aspect vocabularies used in the following stage. Wikipedia structure continues to help us through this stage with each aspect likely to map to a single Wikipedia article. For each aspect we aim to add at least one article to our set. Each aspect is disambiguated individually using Link Probability measure, giving a list of possible articles and their weights of disambiguation confidence. We introduce a cut-off threshold to our selection criteria relative to the highest confidence appearing over all the query aspects. All articles that have a confidence greater than half of the maximum measure are included in our article set.

We also utilize the ability to disambiguate aspects in pairs through the Wikipedia Link Based measure [13] providing added context for better disambiguation relative to the other query aspects. The same threshold mechanics are applied to the Wikipedia Link based measurement and its disambiguation confidences.

Step 3: Aspect Vocabulary Construction. This stage aims to build a weighted vocabulary for each aspect using the article set produced during the

previous stage. All terms appearing in the selected articles are vocabulary candidates weighted by their relation to their corresponding aspects. For each aspect, a vocabulary is produced by adding to it each candidate term from the article set along with the score based on Wikipedia Link Based Measure [13]. This produces a weighted list of terms, named an aspects vocabulary, sorted by their semantic relation to the aspect.

Step 4: Finding Underrepresented Aspects. The final step is to select the best expansion term by supporting the least represented aspect of the query. We build the vocabulary of the entire query by counting term frequencies of all terms in the first 10 documents of the initial Bing search results. We select first 50 highest weighted terms and normalize them so the weights of each term add to give 1. Then the vocabulary of each aspect is compared with the query vocabulary. Each aspect is given a score by multiplying the terms weightings in the aspect vocabulary by their frequency weighting in the query vocabulary. The aspect that produces the lowest score is determined to be the least represented with its highly weighted terms in this aspect's vocabulary assigned as the final output for query expansion.

3.2 Variations to Core Method

The following three methods are devised to deviate in some way from the core method for performance comparison. The three methods share the same Aspect Identification step (Step 1) and Wikipedia Article Selection step (Step 2) with the Core, but their step 3 is changed and they don't have step 4. Their Aspect Vocabulary Construction step (Step 3) changes its priorities from creating a vocabulary for each of the aspects to creating a complete vocabulary for the entire query that provides an output of weighted terms to conclude the query expansion method. In other words, they only build the query vocabulary and weight the terms in the vocabulary for output. The difference between the three methods is the way they weight the terms in the vocabulary.

Relatedness. This method differs the least from the core method. The query vocabulary is built by taking each term existing in the article selection and weighting it equally against all the query aspects. The weighting measure is found by taking the dot product of the terms semantic relation to each of the aspects using the Wikipedia Link Based Measure (WLM) [13], as shown below,

$$w(t) = \prod_{x=1}^n (WLM(t, A_x)) \quad (2)$$

where n represents the number of query aspects and WLM is the way of assigning scores to a term based on its relatedness to the corresponding aspect.

TF.IDF This method uses the TF.IDF weighting to build a vocabulary that represents the entire query. Term frequencies are taken from the Wikipedia Article Selection step (Step 2) with term document occurrences based upon the total collection of Wikipedia articles. The formula used to calculate TF.IDF weighting is given by: The inverse documents frequency is used to measure the general importance of a term to the entire documents collection.

Combined. Combined method involves intersecting our Relatedness and TF.IDF methods where vocabulary construction is determined by terms Relatedness (WLM) and TF.IDF weightings as shown below.

$$combined(t) = w(t) \times tfidf(t) \quad (3)$$

4 Evaluation

For evaluation, we compared the performance of our query expansion core method with that of alternate query expansion methods. Each system processed a set of 150 queries producing a reformulated query. This query is then searched over a web collection to retrieve a ranked set of documents. The retrieval lists for each query are compared with a set of relevance judgments allowing us to calculate evaluation measures. Averages of system queries give the grounds for us to compare the query expansion methods and validate our own.

4.1 Test Data

Three main elements for the performance evaluation of our query expansion method are: the set of queries, a static web document collection and the set of relevance judgment. We use all the 150 testing queries from 2009, 2010, and 2011 TREC Web Track [4, 5, 6], each containing a brief topic and subtopics description. Queries from the 2011 TREC Web Track were aimed to be harder than the previous year's queries, providing our evaluation with a range of query difficulties. The ClueWeb09 dataset at <http://www.lemurproject.org/clueweb09.php/> is our static web collection containing over 1 billion web pages.

Relevance judgments were taken from the TREC Web Track workshop, consisting of manually labeled document set for each query by a panel of experts from their respective years. Judgments were based on the topics of the query. Indri search engine [20] is used to retrieve the documents.

4.2 Measurements

Traditional measures such as Precision and Recall are not suitable for our task. For example, an extended query which increases the first relevant judgement from rank 5 to rank 1, does not change precision or recall, but is understood to be an improvement of retrieval. We develop our own evaluating criteria which consider the ranking/index of each retrieved document.

The documents retrieved by each query expansion method for each query are classified under one of three categories: Relevant, Non-Relevant, or Non-Judged.

Relevant documents indicate that the retrieval fits the description of the queries topic or subtopic, with Non-Relevant documents straying from the query sense. Non-Judged documents are documents that have neither label nor part of the judgments of the Web Track workshops. We can not ignore these Non-Judged documents because they change the ranking of other relevant or non-relevant documents. It is too harsh to treat them as Non-relevant because that is not true. So we value these Non-Judged documents higher than Non-Relevant documents and give them a weighting based upon the fraction of relevant judgments to total judgments existing for that query.

We created two metrics to evaluate our systems, the first describes the performance of a query (original query or an extended query using a single system) and the second describes the performance of an extended query using a single system relative to the performance of the original query. Results will focus on the second metric where we are able to compare the improvements between systems. The first single query measurement is scored with a maximum value of 5050, evaluating the first 100 documents retrieved. Each document is scored based on its index in the retrieval and its label. A documents index score begins at a value of 100 for the first document down to a value of 1 for the hundredth document. This index score is multiplied by its label score to give the document score as shown below. Label scores range in values from 0 to 1 with Relevant documents getting a score of 1 and Non-Relevant documents a score of 0. The Non-Judged documents get a label score relative to the ratio of total relevant to total judgments that exist for the query. This measurement combines a sense of precision and recall measurements. Adding all the document scores of a query together gives the system its query score.

$$queryScore(|D|) = \sum_{idx=0}^{99} (100 - idx) \times labelScore(d_{idx})$$

The second metric presents the performance of the target system relative to the performance of the unaltered query. For each system we select query pairs with the original query and its extended query and both queries have to be meaningful. Meaningfulness is enforced by a threshold to include the query based on the number of Non-Judged document labels of a query. For a query to be included its retrieved document set is required to have at least 25 of the first 50 documents to have a Relevant or Non-Relevant judgment. Using the meaningful query pairs, we can define the performance of an entire system being the average ratio of change of the system relative to the original queries, defined as follows.

$$relativePerformance(extended|Q|, original|Q|) = \frac{1}{n} \sum_{x=0}^n 100 \times \frac{2 \times queryScore(extended[Q_x])}{queryScore(original[Q_x]) + queryScore(extended[Q_x])} - 1$$

4.3 Results and Discussion

This section presents the comparison results of our four methods introduced in this paper and two exiting solutions based on previous research in query expansion. The two alternative solutions are outlined as follows:

Wikipedia Local TF.IDF. Uses the default Wikipedia search interface to retrieve a list of 10 articles to build a query vocabulary. Candidate terms come from the contents of the articles using a TF.IDF measurement to weight them, selecting the highest weighted terms for expansion.

Bing Local TF.IDF. Uses the Bing web search to retrieve a list of 10 documents to build a query vocabulary. Candidate terms come from the contents of the documents using a TF.IDF measurement to weight them, selecting the highest weighted terms for expansion.

Because AbraQ doesn't handle single aspect queries, we are unable to compare the performance of our system with it.

Table 2. Number of queries tested per system

System	Number of Querys evaluated		
	1 Term	2 Terms	5 Terms
Aspect Representation	34	26	16
Combined	31	23	19
TF.IDF	29	27	13
Relatedness	33	23	19
Web Local TF.IDF	43	33	27
Wikipedia Local TF.IDF	33	30	25

We computed relative performance metric over all query expansion solutions posed with the 150 queries from TREC 2009, 2010, and 2011. Each of the six systems, four introduced in this paper and two comparison solutions emulating previous work, is measured through three different sizes of expansion terms. The three sizes are a single term, two terms, and five terms resulting in a total of 18 systems. Table 2 shows the number of queries tested with each expansion size. They are grouped by the number of terms added by the method. With our relative performance metric resulting in varying sizes of query test sets, the average number of queries tested for each system was 26.9 with a maximum of 43 and a minimum of 13, as shown in Table 2.

The following three figures show the comparison results. The y-axis gives the values of our performance metric for each of the methods of the x-axis. Results across the single term expansion, as shown in Figure 2, are in a similar range with the highest performance coming from the Aspect Representation method at 8.6% and the lowest from the Web TF.IDF method at 5.4%.

Increasing the number of terms from one to two reveals a reduced performance to all methods except two, the Combined and Web TF.IDF methods, indicated by Figure 3. The performance of the Combined method at 11.3% comes close to doubling the next method of Wikipedia TF.IDF at 6.8%. This is unexpected as the Combined method is basically a concatenation of the TF.IDF and Relatedness methods which are the two lowest performing.

The general reduction in performance when expanding a query with two terms, contrasted with the Combined methods best performance coming from the two

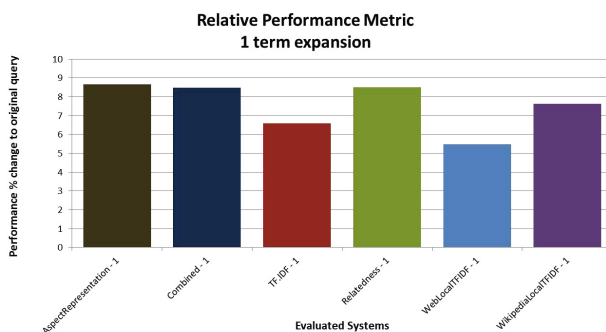


Fig. 2. Performance comparison of six QE methods on one term query expansion

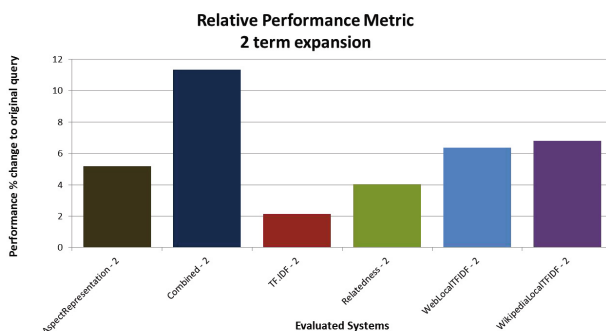


Fig. 3. Performance comparison of six QE methods on two term query expansion

term expansion, highlights the influence of the expansion term quantities. Analysis of a few method expansions shows growing the number of terms steers the retrieval to either provide more matches or produce one of two forms of drifting. The first kind of drift involves the change of topic to something completely detached from the original sense. The second form of drift involves the query sense narrowing too finely on a subset of the query sense. With a lower number of query terms present, their weighting effect on the query is greater, as more are added, their presence reduces. This, coupled with the two forms of drifting, could account for the inconsistent results occurring through the two term expansion evaluation. Our core method could have utilised a dynamic number of expansion terms by re-evaluating the representations of aspects through alternate numbers of expansion terms to search for an optimal number of terms.

Extending the number of terms to 5, in figure 4, shows the largest performance jumps. Aspect Representation at 5 terms performs better than all other methods at 23.6%. The Relatedness method is similar to the Aspect Representation method and trails closely at 21.7%. The third performer of the 5 term methods is the Web TF.IDF method with a strong performance of 17.7% doubling the fourth equal methods of Combined and Wikipedia TF.IDF at 8.3%.

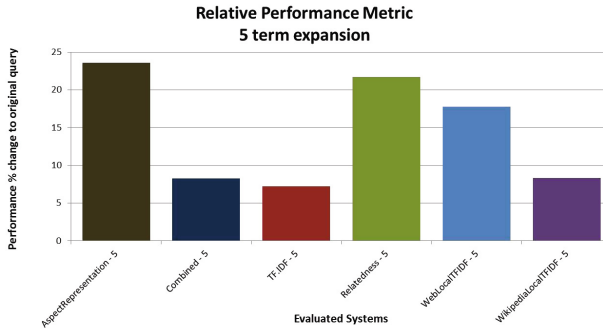


Fig. 4. Performance comparison of six QE methods on five term query expansion

5 Conclusion

This paper presented a query expansion system with Wikipedia as the source for finding related terms. Our system is tested on a large dataset and all four methods show performance improvements. The experiments show that the Aspect Presentation method outperforms other methods. When the aspects of a query are identified well, the vocabularies produced by our Aspect Representation method provide strong candidate terms for a query expansion system.

The limitations of Wikipedia as a sole knowledge base have become apparent through this research. Wikipedia's ability to identify query aspects is powerful because of the close similarities between Wikipedia articles and query aspects although its encyclopedic style is not broad enough to platform the diversity of web search goals. Future work would focus on complementing Wikipedia with alternative sources and forming methods to evaluate each source's performance relative to the query so to avoid the limitations of individual sources.

References

- [1] Bernhard, D.: Query expansion based on pseudo relevance feedback from definition clusters. In: Proceedings of the 23rd International Conference on Computational Linguistics: Posters, COLING 2010, pp. 54–62. Association for Computational Linguistics, Stroudsburg (2010)
- [2] Bhogal, J., Macfarlane, A., Smith, P.: A review of ontology based query expansion. *Inf. Process. Manage.* 43, 866–886 (2007)
- [3] Brenes, D.J., Gayo-Avello, D.: Stratified analysis of aol query log. *Inf. Sci.* 179, 1844–1858 (2009)
- [4] Clarke, C.L., Craswell, N., Soboroff, I.: Overview of the trec 2009 web track (2010)
- [5] Clarke, C.L., Craswell, N., Soboroff, I., Cormack, G.V.: Overview of the trec 2010 web track (2011)
- [6] Clarke, C.L., Craswell, N., Soboroff, I., Voorhees, E.M.: Overview of the trec 2011 web track (2012)

- [7] Crabtree, D.W., Andreae, P., Gao, X.: Exploiting underrepresented query aspects for automatic query expansion. In: Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2007, pp. 191–200. ACM, New York (2007)
- [8] Fellbaum, C. (ed.): WordNet An Electronic Lexical Database. The MIT Press, Cambridge (1998)
- [9] Jiang, X.: Query expansion based on a semantic graph model. In: Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information, SIGIR 2011, pp. 1315–1316. ACM, New York (2011)
- [10] Klyuev, V., Haralambous, Y.: Query expansion: Term selection using the ewc semantic relatedness measure. CoRR, abs/1108.4052 (2011) (informal publication)
- [11] Manning, C., Raghavan, P., Schütze, H.: Introduction to information retrieval. Cambridge University Press (2008)
- [12] Meij, E., de Rijke, M.: Supervised query modeling using wikipedia. In: Proceeding of the 33rd International ACM SIGIR conference on Research and Development in Information Retrieval, SIGIR 2010, pp. 875–876. ACM, New York (2010)
- [13] Milne, D., Witten, I.H.: An effective, low-cost measure of semantic relatedness obtained from wikipedia links. In: Proceedings of AAAI 2008 (2008)
- [14] Milne, D., Witten, I.H.: Learning to link with wikipedia. In: Proceeding of the 17th ACM Conference on Information and Knowledge Management, CIKM 2008, pp. 509–518. ACM, New York (2008)
- [15] Milne, D.N., Witten, I.H., Nichols, D.M.: A knowledge-based search engine powered by wikipedia. In: Proceedings of the Sixteenth ACM Conference on Conference on Information and Knowledge Management, CIKM 2007, pp. 445–454. ACM, New York (2007)
- [16] Nguyen, D., Callan, J.: Combination of evidence for effective web search. In: The Nineteenth Text REtrieval Conference Proceedings TREC 2010 (2010)
- [17] Robertson, G., Gao, X.: Improving abraq: An automatic query expansion algorithm. In: Proceedings of the 2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology, WI-IAT 2010, vol. 1, pp. 653–656. IEEE Computer Society, Washington, DC (2010)
- [18] Ruthven, I., Lalmas, M.: A survey on the use of relevance feedback for information access systems. *Knowl. Eng. Rev.* 18, 95–145 (2003)
- [19] Santamaría, C., Gonzalo, J., Artiles, J.: Wikipedia as sense inventory to improve diversity in web search results. In: Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, ACL 2010, pp. 1357–1366. Association for Computational Linguistics, Stroudsburg (2010)
- [20] Strohman, T., Metzler, D., Turtle, H., Croft, W.B.: Indri: a language-model based search engine for complex queries. In: Proceedings of the International Conference on Intelligent Analysis, Technical report (2005)
- [21] White, R.W., Marchionini, G.: Examining the effectiveness of real-time query expansion. *Inf. Process. Manage.* 43, 685–704 (2007)
- [22] Xu, Y., Jones, G.J., Wang, B.: Query dependent pseudo-relevance feedback based on wikipedia. In: Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2009, pp. 59–66. ACM, New York (2009)