Performance Evaluation of State-of-the-Art Ranked Retrieval Methods and Their Combinations for Query Suggestion

Suthira Plansangket and John Q. Gan
School of Computer Science and Electronic Engineering, University of Essex, Colchester, Essex, U.K.

Keywords: Query Suggestion, Query Expansion, Information Retrieval, Search Engine, Performance Evaluation.

Abstract: This paper investigates several state-of-the-art ranked retrieval methods, adapts and combines them as well for query suggestion. Four performance criteria plus user evaluation have been adopted to evaluate these query suggestion methods in terms of ranking and relevance from different perspectives. Extensive experiments have been conducted using carefully designed eighty test queries which are related to eight topics. The experimental results show that the method developed in this paper, which combines the TF-IDF and Jaccard coefficient methods, is the best method for query suggestion among the six methods evaluated, outperforming the most popularly used TF-IDF method. Furthermore, it is shown that re-ranking query suggestions using Cosine similarity improves the performance of query suggestions.

1 INTRODUCTION

Internet search engines play the most important role in finding information from the web. One of the great challenges faced by search engines is to understand precisely users' information need, since users usually submit very short (only a couple of words) and imprecise queries (Fonseca et al., 2003). Most existing search engines retrieve information by finding exact keywords. Sometimes, users do not know the precise vocabulary of the topic to be searched and they do not know how search algorithms work so as to produce proper queries (Delgado et al., 2009).

One solution to these problems is to devise a query suggestion module in search engines, which helps users in their searching activities. (Kelly et al., 2010) pointed out that query suggestions were useful when users ran out of ideas or faced a cold-start problem. (Kato et al., 2013) analysed three types of logs in the Microsoft's search engine Bing and found that query suggestions were often used when the original query is a rare query or a single-term query or after the user has clicked on several URLs in the first search result page.

1.1 Query Expansion and Reformulation

Query expansion is a technique to expand the query with related words and is widely used for query suggestion. It aims to improve the overall recall of the relevant documents (Nallapati and Shah, 2006; Baeza-Yates and Ribeiro-Neto, 2011). Query reformulation or dynamic query suggestion is more complex than query expansion, which forms new queries using certain models (Nallapati and Shah, 2006; Costa et al., 2013; Kato et al., 2012). This paper mainly addresses query expansion.

1.2 Explicit and Implicit Feedback

Relevance feedback plays an important role in query suggestion. There are two major categories of relevance feedbacks. Explicit feedback is provided directly by users, which is expensive and time consuming. On the other hand, implicit feedback is derived by the system (Baeza-Yates and Ribeiro-Neto, 2011). The system derives the feedback information from several sources of features, such as log files, web documents, and ontology. This paper focuses on query suggestion methods based on implicit relevance feedback.

There are many studies on query suggestion using log files (Huang et al., 2003; Fonseca et al., 2003; Baeza-Yates et al.; 2004, Boldi et al, 2008; Mei et al., 2008; Cao et al., 2008; Boldi et al., 2009; Kato et al., 2011; Kruschwitz et al., 2013; Zanon et al., 2012; Liao et al., 2014). Various ontologies have been applied to create knowledge-driven models for

Plansangket S. and Q. Gan J..
Performance Evaluation of State-of-the-Art Ranked Retrieval Methods and Their Combinations for Query Suggestion.
DOI: 10.5220/0005018401410148
In *Proceedings of the International Conference on Knowledge Discovery and Information Retrieval* (KDIR-2014), pages 141-148
ISBN: 978-989-758-048-2

generating query suggestions, such as WordNet (Gong et al., 2005; Wan et al., 2012), Wikipedia (Hu et al., 2013), ODP and YAGO (Suchanek et al., 2007; Hoffart et al., 2011; Biega et al., 2013; Suchanek et al., 2013). Query suggestions can also be generated from query related features extracted from web documents returned by search engines (Delgado et al., 2009). There are some studies on query suggestion that combined query log and web search results (Yang et al., 2008) or combined query log and ontology (Song et al., 2012).

1.3 Ranked Retrieval Models

In ranked retrieval models, the system returns an ordered list of top matching documents with respect to a query. Typical ranked retrieval methods include Jaccard coefficient, Cosine similarity and TF-IDF (Jurafsky and Martin, 2008), which will described in more detail in the next section.

In information retrieval, ranked retrieval methods are used to order relevant documents with respect to a query. Similarly, highly relevant query suggestions are preferable to appear first in query suggestions (Manning et al., 2008). Therefore, it is reasonable to adapt ranked retrieval methods for query suggestion.

This paper investigates the state-of-the-art ranked retrieval methods, namely TF-IDF, Jaccard coefficient, and Cosine similarity, and adapts and combines them for query suggestion. These methods extract query related items or features from the titles and snippets of the top eight documents returned from Google search and rank them using different concepts as query suggestions. This paper conducts comprehensive performance evaluation of these methods using multiple criteria emphasizing different perspectives.

2 METHODS

2.1 Query Suggestion Methods

2.1.1 TF-IDF

Term frequency – inverse document frequency (TF-IDF) (Baeza-Yates and Ribeiro-Neto, 2011) is the most popular term weighting scheme in information retrieval. The TF-IDF score of a term in a set of documents is calculated as follows:

$$tfidf_i = \sum_{j=1}^{N} w_{i,j} \quad (1)$$

$$w_{i,j} = \begin{cases} (1 + \log f_{i,j}) \times \log \frac{N}{n_i}, & if\ f_{i,j} > 0 \\ 0, & otherwise \end{cases} \quad (2)$$

where $f_{i,j}$ is the frequency of term i in document j, n_i is the number of documents in which term i appears, N is the total number of documents.

TF-IDF has been used to measure word relatedness (Yih and Qazvinian, 2012). Therefore, it can be applied to identify terms in the documents returned from Google search, which are mostly relevant to the original query, as query suggestions.

2.1.2 Jaccard Coefficient

Jaccard coefficient (Jurafsky and Martin, 2008) is a measure of overlap of two returned documents D_1 and D_2, which are represented as vectors of terms and may not have the same size.

The Jaccard coefficient for a length-normalized model is calculated as follows:

$$Jaccard(D_1, D_2) = \frac{|D_1 \cap D_2|}{|\sqrt{D_1 \cup D_2}|} \quad (3)$$

where $\cap$ represents intersection and $\cup$ union. In this paper, D_1 and D_2 are bags of words which contain query suggestion candidates that are selected from words which appear in at least two returned documents. In mathematics, the notion of multiset or bag is a generalization of the notion of set, in which members are allowed to appear more than once. The intersection or union of multisets is a multiset in general (Blizard, 1989).

If a query suggestion candidate is from more than two returned documents, its Jaccard coefficient can be extended as

$$Jaccard(D_1, D_2, \ldots D_M) = \frac{|D_1 \cap D_2 \cap \ldots D_M|}{|\sqrt{D_1 \cup D_2 \cup \ldots D_M}|} \quad (4)$$

In this paper, for each query suggestion candidate, M documents that contain this suggestion term are identified, and then Jaccard coefficient is calculated as the score to rank this candidate.

Jaccard coefficient has been used to measure the similarity between search texts (Zanon et al., 2012). (Kulkarni and Caragea 2009) used this method to compute semantic relatedness between two concept clouds.

2.1.3 Cosine Similarity

The vector space model using Cosine similarity (Jurafsky and Martin, 2008) is one of the most commonly used methods to rank returned documents according to their proximity (similarity of vectors) to the query.

In this paper, Cosine similarity is used to measure the similarity between a query suggestion candidate and the original query. For length-normalized vectors, Cosine similarity is simply a dot

product:

$$\cos(\vec{q}, \vec{s}) = \vec{q}.\vec{s} = \sum_{i=1}^{B} q_i s_i \quad (5)$$

where q_i is the term frequency of the original query in returned document i, s_i the term frequency of a query suggestion candidate in returned document i, and B the number of documents in which both the original query and the query suggestion candidate appear.

2.1.4 Query Suggestion Methods to Be Investigated

By adaptation and combination of the TF-IDF, Jaccard coefficient, and Cosine similarity methods, six query suggestion methods as shown in Table 1 are investigated. They use different methods for feature (or term) extraction and ranking and will be evaluated using multiple performance criteria.

In the proposed combinations of methods, the query suggestions are selected from the top ten TF-IDF scores or Jaccard coefficient scores, depending on which scores are more important or reflect better relevance. After that, these suggestions may be re-ranked in descending order by Cosine similarity scores.

Table 1: Query suggestion methods to be investigated.

No.	QS methods	Feature extraction and ranking (selection)	Suggestion re-ranking
1	Tfidf	TF-IDF score	-
2	Tfcos	TF-IDF score	Cosine similarity score
3	Jac	Jaccard coefficient score	-
4	Jacos	Jaccard coefficient score	Cosine similarity score
5	Tfjac	TF-IDF score and Jaccard coefficient score	-
6	Tfjacos	TF-IDF score and Jaccard coefficient score	Cosine similarity score

Our initial experiment found that the TF-IDF method was capable of producing suggestions relevant to the user's original query whilst Jaccard coefficient was good to rank the suggestions. Therefore, the Tfjac method was proposed in this paper, which selects terms from the combination of the top ten candidate words from the TF-IDF method and the Jaccard coefficient method. The process starts with finding duplicate words from both methods. If the number of these words is less than ten, more candidate words from the Jaccard coefficient method are added. If the number of terms is still less than ten, more candidate words from the TF-IDF method are added till ten query suggestions are selected.

For the Tfjacos method, the selection process is the same as the Tfjac method; however, the selected query suggestions are re-ranked in descending order by their Cosine similarity scores.

2.2 Evaluation Methods

2.2.1 Mean Reciprocal Rank (MRR)

MRR (Dybkjaer et al., 2007; Otegi et al., 2011) is a statistic measure suitable for query suggestion's ranking evaluation. For query j, the reciprocal rank of a good query suggestion i, RR_{ji}, is the multiplicative inverse of the rank of this suggestion in the list of potential query suggestions made by a query suggestion method, r_{ji} (0 if no such query suggestion in the list), *i.e.*,

$$RR_{ji} = \frac{1}{r_{ji}} \quad (6)$$

MRR is the average of the reciprocal ranks of all the good suggestions for all the queries:

$$MRR = \frac{1}{q}\sum_{j=1}^{q}\frac{1}{Q_j}\sum_{i=1}^{Q_j} RR_{ji} \quad (7)$$

where Q_j is the number of good suggestions for query j, q is the number of queries. For a query, its good query suggestions are determined partly by users' judgement and partly by the Google query suggestions in the experiment in this paper. More detailed explanation is given in Section 3.

2.2.2 Mean Average Precision (MAP)

MAP (Manning et al., 2008; Otegi et al., 2011) is an average precision across multiple queries and rankings. MAP assumes that users are interested in finding many relevant query suggestions and highly relevant suggestions should appear first in the list of suggestions.

Let the rank of the ith relevant query suggestion in the potential query suggestions made by a query suggestion method for query j be r_{ji}. The precision of the ith suggestion is defined by

$$P_{ji} = \frac{number\ of\ relevant\ suggestions}{number\ of\ suggestions\ examined} = \frac{i}{r_{ji}} \quad (8)$$

For an irrelevant suggestion, the precision is set to 0. MAP is defined as the average precision of all the query suggestions for all the queries:

$$MAP = \frac{1}{q}\sum_{j=1}^{q}\frac{1}{Q_j}\sum_{i=1}^{Q_j} P_{ji} \quad (9)$$

where Q_j is the number of relevant query

suggestions for query j and q is the number of queries.

2.2.3 Discounted Cumulated Gain (DCG)

MAP allows only binary relevance assessment (relevant or irrelevant), which does not distinguish highly relevant suggestions from mildly relevant suggestions. Discounted cumulated gain (DCG) (Jurafsky and Martin, 2008; Manning et al., 2008) is a metric that combines graded relevance assessments effectively.

Cumulative Gain (CG) of the Q_j query suggestions for query j is defined by

$$CG_j = w_1 + w_2 + \cdots w_{Q_j} \quad (10)$$

where w_i is rating or weighting factor of the rank of the ith suggestion. Discounted Cumulative Gain (DCG) is defined by using a discount factor $1/(log_2 i)$:

$$DCG_j = w_1 + \frac{w_2}{log_2 2} + \frac{w_3}{log_2 3} + \ldots \frac{w_{Q_j}}{log_2 Q_j} \quad (11)$$

The average DCG (AvgDCG) over q queries is

$$AvgDCG = \frac{1}{q}\sum_{j=1}^{q} DCG_j \quad (12)$$

2.2.4 Precision at 10 (P@10)

Precision is the ratio of the number of relevant suggestions to the total number of irrelevant and relevant suggestions. This is a simple performance criterion and is often used as a baseline evaluation method.

Precision@10 (Okabe and Yamada, 2007, Otegi et al., 2011) is the precision for the top ten query suggestions, which is calculated as follows:

$$P@10 = \frac{number\ of\ relevant\ suggestions\ among\ top\ 10}{10} \quad (13)$$

2.2.5 Integrated Evaluation and User Evaluation

The above four evaluation criteria emphasize different aspects of the performance. MRR is used to measure the performance of ranking, whilst P@10 is used to measure the performance of generating relevant query suggestions. MAP and DCG can measure the performance of both ranking and producing relevant suggestions. Integrating the evaluation results from these four methods may lead to more comprehensive evaluation.

In order to check whether the evaluation using the above criteria is acceptable by real users, user evaluation will be conducted as well. Questionnaires are used to obtain users' evaluative feedback, which ask the participants to select a top suggestion respectively from the query suggestions made by each query suggestion method for each of the eighty test queries and then rank the six top suggestions made by the six query suggestion methods for each test query from one to six.

3 EXPERIMENT

3.1 Experimental Design

(He and Ounis, 2009) proposed an entropy measure which estimates how the occurrences of a query term spread over returned documents. The higher the entropy is, the more a returned document is related to the query. Their results show that the entropy in the top five returned documents is very high, and it decreases rapidly in the remaining documents. Therefore, it has been decided that in this experiment query suggestions are created from analysing the top eight Google search returned documents. That would be enough to generate highly relevant or good suggestions to the original query from these documents. Each document is pre-processed as follows. First of all, not the whole document, but only the title and snippet content in each document are considered. After that, all HTML tags are removed and all contents are separated into tokens. Thirdly, since the most selective terms for query suggestions should be nouns (Baeza-Yates and Ribeiro-Neto, 2011; Bordag, 2008), only nouns are considered for suggestions.

A simple search engine using the Google API and the six query suggestion methods described in the previous section have been implemented in this experiment. From the analysis of titles and snippets of the top eight Google returned documents using the query suggestion methods, query suggestions are generated by each method for each query. For evaluation purposes, eighty test queries were selected from eight popular search topics (categories), as shown in Table 2. Each category contains ten queries consisting of one to three words that are commonly known and convenient for user evaluation.

It is important to know whether a query suggestion is truly good or not in the performance evaluation. In this experiment, highly relevant, mildly relevant and irrelevant suggestions for each test query were judged by two approaches in order to reduce subjective bias to expected results and make the experimental results more reliable. Fifty percent of the decisions were based on the suggestions by the Google search engine, which has been widely

recognized, and another fifty percent of the decisions were made by users who were three PhD students in this experiment.

Table 2: Categories of test queries.

Category	Description	Number of queries
1	Movies	10
2	Food	10
3	Traveling	10
4	Shopping	10
5	Sports	10
6	Arts	10
7	Flowers	10
8	Animals	10
Total		80

3.2 Experimental Results and Evaluation

The experimental results are shown in the following tables, where an asterisk indicates that the related score differs significantly from the best one with the p value ≤ 0.05. The method for statistical significance test is t-test.

3.2.1 MRR Results

The results of evaluation using MRR are given in Table 3, which show that the best query suggestion methods are Tfjac and Jacos followed by Tfjacos, and the ranking score of Tfidf is significantly lower than those of the best methods.

Table 3: MRR results.

QS methods	MRR scores	Rank
Tfidf	0.2934*	6*
Tfcos	0.3254	4
Jac	0.3211	5
Jacos	0.3846	1
Tfjac	0.3846	1
Tfjacos	0.3687	3

3.2.2 MAP Results

The results of evaluation using MAP are given in Table 4, which show that Tfjac is the best method for generating query suggestions in terms of ranking and producing relevant words. However, its score is not significantly different from the others.

Table 4: MAP results.

QS methods	MAP scores	Rank
Tfidf	0.9544	4
Tfcos	0.9519	5
Jac	0.9485	6
Jacos	0.9695	2
Tfjac	0.9712	1
Tfjacos	0.9609	3

3.2.3 DCG Results

The results of evaluation using DCG are given in Table 5, which show that Tfjac is the best method for ranking and producing highly relevant suggestions followed by Jacos and Tfjacos.

Table 5: DCG results.

QS methods	DCG scores	Rank
Tfidf	8.0339	6
Tfcos	8.0898	5
Jac	8.2880	4
Jacos	8.5578	2
Tfjac	8.5880	1
Tfjacos	8.4120	3

3.2.4 P@10 Results

Table 6: P@10 results.

QS methods	P@10 scores	Rank
Tfidf	0.9145*	6*
Tfcos	0.9147*	5*
Jac	0.9524	1
Jacos	0.9524	1
Tfjac	0.9524	1
Tfjacos	0.9232	4

The results of evaluation using P@10 are given in Table 6, which show that Tfjac, Jac, and Jacos have the same score and outperform other methods in terms of generating relevant suggestions. On the other hand, the scores of Tfidf and Tfcos methods are significantly lower than those of the best methods.

3.2.5 Integrated Evaluation

The table below shows the rankings of the six query suggestion methods in terms of the four evaluation methods respectively. For the two methods whose rankings are significantly lower than the others, the ranks are multiplied by two.

The sum of the rankings in Table 7 can be transferred into MRR scores as shown in Table 8. It is clear that Tfjac is the best method overall for generating query suggestions followed by Jacos and Jac. Tfjacos, Tfcos and Tfidf are significantly worse than the other three methods.

Table 7: Summary of evaluation results.

QS methods	MRR ranking	MAP ranking	P@10 ranking	DCG ranking	Sum
Tfidf	6*(12)	4	6*(12)	6	34
Tfcos	4	5	5*(10)	5	24
Jac	5	6	1	4	16
Jacos	1	2	1	2	6
Tfjac	1	1	1	1	4
Tfjacos	3	3	4	3	13

Table 8: Integrated evaluation in MRR scores.

QS methods	MRR scores	Rank
Tfidf	0.1458	6*
Tfcos	0.1875	5*
Jac	0.4042	3
Jacos	0.7500	2
Tfjac	1.0000	1
Tfjacos	0.3125	4*

3.2.6 User Evaluation

Five PhD students studying in different fields participated in the user evaluation. The results of the user rankings in MRR scores are given in Table 9, which show that the majority of participants indicated that the query suggestions made by Jacos were the best followed by Tfjacos and Tfjac, but they are not significantly different. It should be noted that only one top suggestion from each query suggestion was considered in the user evaluation here, which might lead to biased results and should be improved in future work.

Table 9: User evaluation in MRR scores.

QS methods	MRR scores	Rank
Tfidf	0.6495	5
Tfcos	0.6549	4
Jac	0.6157	6
Jacos	0.7027	1
Tfjac	0.6732	3
Tfjacos	0.6909	2

4 CONCLUSIONS

This paper has investigated several ranked retrieval methods, adapted and combined them as well for query suggestion. Six query suggestion methods including the combined methods developed in this paper have been evaluated using four performance criteria and user evaluation as well. The experimental results show that Tfjac is the best for generating query suggestions among the six methods evaluated in terms of relevance and ranking. It is demonstrated that Tfjac is capable of combining the good query suggestions from both TF-IDF and Jaccard coefficient methods. However, this combined method may deserve further investigation and there may be room for further improvement by using better combination strategies.

It is also found that query suggestions re-ranking using Cosine similarity helps to generate better query suggestions in general. For example, the majority of the experimental results show that Jacos is the second best method which selects the query suggestion candidates from Jaccard coefficient and re-ranks the selected query suggestions using Cosine similarity. Its top query suggestion is better than that of Tfjac, as shown in the user evaluation results. It should be noted that in the user evaluation conducted here only the top suggestion from each query suggestion was evaluated. This is a limitation of the user evaluation conducted in this way and should be further investigated.

Performance evaluation usually depends on the queries used in the experiment and the judgment on the relevance of query suggestions with the original queries. This paper has designed eighty queries related to eight topics based on Google search results and users' suggestions and adopted multiple evaluation criteria from different perspectives to ensure fair comparison and evaluation. However, further work should be conducted to overcome the limitation in this aspect of the performance evaluation and in the user evaluation conducted in this paper, for example, using standard or previously used benchmarks. Future work in line of this

research would also include improving query suggestion by using knowledge base and user feedback, such as click-through data, through computational intelligence approaches.

REFERENCES

Baeza-Yates, R. and Ribeiro-Neto, B., 2011. *Modern information retrieval: the concepts and technology behind search.* England: Pearson Education Limited.

Baeza-Yates, R., Hurtado, C., and Mendoza, M., 2004. Query recommendation using query logs in search engines. *In Proc. of the 2004 International Conference on Current Trends in Database Technology*, page 588-596

Biega, J., Kuzey, E., and Suchanek, F., 2013. Inside YAGO2s: A transparent information extraction architecture. *In Proc. of WWW 2013,* Rio de Janeiro, Brazil.

Blizard, W. D., 1989. Multiset theory. *Notre Dame Journal of Formal Logic*, vol. 30, no. 1, page 36-66.

Boldi, P., Bonchi, F., Castillo, C., Donato, D., and Vigna, S., 2009. Query suggestion using query flow graphs. *In Proc. of the 2009 Workshop on Web Search Click Data,* Milan, Italy, page 56-63.

Boldi, P., Bonchi, F., Castillo, C., Donato, D., Gionis, A, and Vigna, S., 2008. The query flow graph: model and applications. *In Proc. of CIKM'08,* California, USA.

Bordag, S., 2008. A comparison of co-occurrence and similarity measures as simulations of context. *In Proc. of the 9th International Conference on Computational Linguistics and Intelligent Text Processing,* Springer-Verlag Berlin, Heidelberg, page 52-63.

Cao, H., Jiang, D., Pei, J., He, Q., Liao, Z., Chen, E., and Li, H., 2008. Context-aware query suggestion by mining click-through and session data. *In Proc. of KDD'2008*, Nevada, USA.

Costa, M., Miranda, J., Cruz, D., and Gomes, D., 2012. Query suggestion for web archive search. *In Proc. of the 10th International Conference on Preservation of Digital Objects (iPres 2013)*, Lisbon, Portugal.

Delgado, M., Martin-Bautista, M.J., Sanchez, D., Serrano, J.M., and Vila, M.A., 2009. Association rules and fuzzy association rules to find new query terms. *In Proc. of the Third Conference of the EUSFLAT*, Lisbon, Portugal, page 49–53.

Dybkjaer, L., Hemsen, H., and Minker, W. (Eds.), 2007. *Evaluation of text and speech systems.* Springer, Dordrecht, Netherlands.

Fonseca, B.M., Golgher, P.B., de Moura, E.S., and Ziviani N., 2003. Using association rules to discover search engines related queries. *In Proc. of The First Latin American Web Congress,* USA, page 66-71.

Gong, Z., Cheang, C., and Hou, L., 2005. Web query expansion by WordNet. *In LNCS 3588*, page 166-175.

He, B. and Ounis, I., 2009. Studying query expansion effectiveness. *In Proc. of the 31th European Conference on IR Research on Advances in Information Retrieval*, Toulouse, France , page 611 – 619.

Hoffart, J., Suchanek, F., Berberich, K., Lewis-Kelham, E., Melo, G., and Weikum, G., 2011. YAGO2: exploring and querying world knowledge in time, space, context, and many languages. *In Proc. of WWW 2011*, Hyderabad, India.

Hu, H., Zhang, M., He, Z., Wang, P., and Wang, W., 2013. Diversifying query suggestions by using topics from Wikipedia. *In Proc. of the 2013 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology*, Atlanta, GA.

Huang, C., Chien, L., and Oyang, Y., 2003. Relevant term suggestion in interactive web search based on contextual information in query session logs. *Journal of the American Society for Information \Science and Technology*, vol. 54, no. 7, page 638-649.

Jurafsky, D. and Martin, J. H., 2008. *Speech and language processing: an introduction to natural language processing.* Computational Linguistics and Speech Recognition. Second Edition, Prentice Hall.

Kato, M., Sakai, T., and Tanaka, K. 2011. Query session data vs. clickthrough data as query suggestion resources. *In Proc. of ECIR 2011*, Dublin, Ireland.

Kato, M., Sakai, T., and Tanaka, K., 2012. Structured query suggestion for specialization and parallel movement: effect on search behaviors. *In Proc. of WWW 2012*, Lyon, France, page 389-398.

Kato, M., Sakai, T., and Tanaka, K., 2013. When do people use query suggestion? A query suggestion log analysis. *Information Retrieval,* vol. 16, no. 6, page 725-746.

Kelly, D., Cushing, A., Dostert, M., Niu, X., and Gyllstrom, K., 2010. Effects of popularity and quality on the usage of query suggestions during information search. *In Proc. of CHI'2010*, Atlanta, USA.

Kruschwitz, U., Lungley, D., Albakour, M. and Song, D., 2013. Deriving query suggestions for site search. *In Journal of the American Society for Information Science and Technology*, vol. 64, no. 10, page 1975–1994.

Kulkarni, S. and Caragea, D., 2009. Computation of the semantic relatedness between words using concept clouds. *In Proc. of KDIR 2009*, page 183-188.

Liao, Z., Song, Y., Huang, Y., He, L., and He, Q. 2014. Task trail: an effective segmentation of user search behavior. *IEEE Transactions on Knowledge and Data Engineering* (in press)

Manning, C. D., Raghavan, P., and Schutze, H., 2008. *Introduction to information retrieval.* England: Cambridge University Press.

Mei, Q., Zhou, D., and Church, K., 2008. Query suggestion using hitting time. *In Proc. of CIKM'08*, California, USA.

Nallapati, R. and Shah, C., 2006. Evaluating the quality of query refinement suggestions in information retrieval. *In Proc. of CIKM 2006*, Arlington, Virginia, USA.

Okabe, M. and Yamada, S., 2007. Semisupervised query expansion with minimal feedback. *IEEE Transactions*

on Knowledge and Data Engineering, vol. 19, no. 11, page 1585-1589.

Otegi, A., Arregi, X., and Agirre, E., 2011. Query expansion for IR using knowledge-based relatedness. *In Proc. of the 5th International Joint Conference on NLP*, Chang Mai, Thailand, page 1467-1471.

Song, Y., Zhou, D., and He, L., 2012. Query suggestion by constructing term-transition graphs. *In Proc. of WSDM'12,* Seattle, Washington, USA.

Suchanek, F., Kasneci, G., and Weikum, G., 2007. YAGO: a core of semantic knowledge unifying WordNet and Wikipedia. *In Proc. of WWW 2007*, Banff, Alberta, Canada.

Suchanek, F., Hoffart, J., Kuzey, E., Lewis-Kelham, E., 2013. YAGO2s: modular high-quality information extraction with an application to flight planning. *In Proc. of the German Computer Science Symposium (BTW 2013),* Magdeburg, Germany.

Wan, J., Wang, W., Yi, J., Chu, C., and Song, K., 2012. Query expansion approach based on ontology and local context analysis. *Research Journal of Applied Sciences, Engineering and Technology,* vol. 4, no. 16, page 2839-2843.

Yang, J., Cai, R., Jing, F., Wang, S., Zhang, L., and Ma, W., 2008. Search-based query suggestion. *In Proc.of CIKM'08*, California, USA.

Yih, W. and Qazvinian, V., 2012. Measuring word relatedness using heterogeneous vector space models. In *Proc. of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT-2012),* Montreal, Canada.

Zanon, R., Albertini, S., Carullo, M., and Gallo, I., 2012. A new query suggestion algorithm for taxonomy-based search engines. *In Proc. of KDIR 2012*, page 151-156.