TOPIC BASED QUERY SUGGESTION USING HIDDEN TOPIC MODEL FOR EFFECTIVE WEB SEARCH

M.BARATHI ,  S.VALLI

1 Anna University, Department of Computer Science and Engineering, SMK Fomra Institute of Technology, Chennai-603103
2 Anna University, Department of Computer Science and Engineering, Chennai-600025

E-mail: 1 bharathi.damu@gmail.com, 2 valli@annaniv.edu
1 Corresponding author: bharathi.damu@gmail.com

ABSTRACT

Keyword-based web search is widely used for locating information on the web. But, web users lack sufficient domain knowledge and find it difficult to organize and formulate input queries which affect search performance. Existing method suggests terms using the statistics in the documents, query logs and external dictionaries. This novel query suggestion method suggests terms related to topics present in the input query and re-rank the retrieved documents. A generative model, Latent Dirichlet Allocation (LDA) is used to learn the topics from the underlying documents. The high probability words in a topic are selected using the Kullback liebler(KL) divergence measure and presented to the user for suggestion, to enrich the user query and to narrow the search. The re-ranking technique of this approach uses the initial retrieval position of the document to re-rank the documents. The suggested queries by the hidden topic approach and by keyword search are analysed.

Keywords: Latent Dirichlet Allocation, Kullback-Liebler Divergence, Query Suggestion, Web Search,

1. INTRODUCTION

The increasing information on the web creates many challenges for the web search. Most commercial search engine returns the same results for the different queries regardless of the users’ interest. The search engines suffer from word mismatch problem, since the query and the documents terms are compared on a logical level instead of the semantic level.

Many works such as query suggestion, collaborative search, contextual search, personalized search, query expansion, have been studied and these works use technique such as clustering, classification, semantics and ontology to improve the quality of the web search. However, users face problems when using a search engine to formulate an accurate input query. If a user is not familiar with some domain, they might fail to choose appropriate keywords in formulating the query. Furthermore, queries submitted by the users are usually short and ambiguous [1]. These short queries do not express the user needs. As a result, lots of irrelevant pages are retrieved. A synonym is a word having the same meaning as another word. For instance the word animal is a synonym of a living organism. A polysemy is a word with multiple meanings. For example, the word cell can be used to refer to a small room in one context and the basic structural and functional unit of an organism in another context which create term mismatch problem and force the user to refine their input queries. Sometime ambiguous queries lead to non-relevant results [2, 3]. Many current studies focus on improving the performance of suggestion strategy for finding queries related to the topics.

Query suggestion is also known as interactive query expansion. It solves the word mismatch problem. It helps users in formulating high quality queries by presenting a list of suggested queries to choose during the retrieval session. The selected term is then added to the initial query to enrich the user query [47, 49]. Most existing works generate the candidate terms using relevance feedback and manually constructed thesaurus [4, 5]. Recent work focus on discovering the relationship of term in general dictionaries such as word net [6], knowledge base such as Wikipedia [7], and query logs [8].

632
However, existing studies neglect the topic information which is important for specifying the user’s information needs [9]. A document contains more than one topic and a user is generally interested in a single or a few topics. Thus, queries that reflect the user’s intention on a topic perform well in retrieving high quality documents with relevant topics.

Many works have been carried out based on the latent topic information to analyze the queries in the field of information retrieval. Huang et al. [22] used clicked documents as training data and topic information to compute the similarity between the input query and the candidate query. He et al. [23] utilized the interaction information to detect topics of queries. Carman et al. [39] applied LDA based personalized ranking formulas to a query log dataset. The disambiguated queries are computed using the terms in the titles and snippets of documents present in the cluster [38]. The query expansion technique uses document clustering technique and ontology concepts to disambiguate user queries [42]. Cao et al [41] combined error correction, topic-based query suggestion and query expansion to find the relevant patents and also improved the user search.

This topic based approach is compared with the existing keyword and cluster based approach with respect to the topic model to analyze the retrieval performance.

The main objective of this topic based approach is to improve the search results. In this work a topic based query suggestion (LDA) disambiguate the query and re-rank the retrieved documents. The main contributions of this work are as follows;

1. The topic based query suggestion refine and formulates the queries.
2. To select high probability words in a topic to form suggestion.
4. A novel method to re-rank the documents.
5. Evaluation of the proposed approach using classic 3 benchmark datasets.

Based on these observations, the novel query suggestion technique suggest query terms related to the topics and re-rank the documents.

The structural re-ranking algorithm treats the query and the content individually when computing the re-ranking scores. HITS [14] and Page Rank [15] are the widely used approach for ranking documents. However, approaches depend only on the structure of the global graph or sub-graph, which could ignore the important information content of a document. The re-ranking algorithm [13] uses only the structure of the global graph which leads to the problem of topic drift.

A novel document re-ranking method based on Latent Dirichlet Allocation (LDA) [16] is used in this topic based query suggestion. Instead of relying on graph based techniques, this topic based approach try to identify the latent structure of “Topic” in the initial retrieval set. Then the similarity between the query and the initial retrieval results based on latent semantic information is computed. The probability of a document is grouped over from all the generated topics. Re-ranking is done by combining the initial retrieval scores and the latent information of the grouped documents.

The rest of the paper is organized as follows; in section 2, the related works are presented. Section 3 describes the query suggestion process. Section 4 explains how the candidate terms are selected. Sections 5 discuss the ranking method. Sections 6 illustrate the experimental result. Section 8 conclude the work and suggest future enhancement.

2. RELATED WORKS

A lot of research exists for disambiguating the user query based on interactive query suggestion technique. Most modern search engine uses auto-query completion to suggest possible complete queries [40]. But, it has a critical drawback. If the input query is totally new to the current web logs, its click information is not accessible from the web data. The retrieval results based on latent semantic information is computed. The resulting suggestions are disorganized. Also clustering technique is used to obtain terms. Cluster labels are the candidate term for query suggestion [10]. Z Liu et al [48] proposed an algorithm to improve the retrieved result based on clustered result. Li et al [45] used topically related queries to improve the quality of the suggestions.

Complete search [17] takes the number of documents for suggestions and the context as the ranking mechanism. Bast et al [7] extended this work by incorporating the term clusters into
the documents. Interactive query expansion is based on document, terms and concepts. Document based method uses statistics information namely, co-occurrence [18, 19] to suggest the relevant terms in the documents. Term based methods try to discover the relation of term from manually constructed thesaurus [4], relevance feedback [4, 5], external dictionaries (e.g.) word net [6] and query logs [8]. Concept based methods use knowledge bases namely, Wikipedia and web directory [7, 20] and query logs [21] to map query terms to a number of concepts. Huang et al [22] used clicked document sets and LDA to analyze the topic information and to measure the similarity of the two queries. The original queries are substituted with the queries detected by the topics [23]. These try to discover the topics based on the query for capturing the user search intention.

Recent studies [24-28] focus on web logs where the click information provides available implicit feedback. If the input query is totally new to current web logs, its click information is not accessible from the web query logs for query similarity computation. Some cluster related queries based on the common clicked URLs [29].

In traditional IR system, in response to the query, the system determine the best result between the query and the documents and return the list of retrieval results in decreasing order of their relevance. Recently, some research has been carried out to re-rank the results. Some of the approaches represent the document entities as a connected graph using the content information. Zhang et al [11] re-rank the search results by optimizing diversity and information richness with the use of affinity ranking graph. Kurland and Lee [12] introduced a structural re-ranking approach by exploiting asymmetric relationship among documents induced by language models. Deng et al. [13] applied a family of semi-supervised machine learning method to document graph constructed by incorporating different evidences.


3. THE QUERY SUGGESTION PROCESS

3.1 Process Overview

The block diagram of the topic based query suggestion process is shown in Figure 1. The user submits the initial query to the search engine and retrieves the top N ranked documents. These set of documents are preprocessed by removing stop words and Porter,M.F (1980)[35] suffix stripping algorithm stem each word by transforming the word to their root form. For example the words “compute”, “computing”, and “computed” are stemmed to “compute” and “dogs” as dog”. The preprocessed documents are given as input to the LDA for the generation of topics. The generated topics are compared with the user query to select the high probability words that are meaningful to the queries using kullback-liebler(KL) divergence measure. The selected terms are sorted in descending order according to their relevance. The ranked terms are provided as suggestion to the user to enrich the user query and to focus their search. The retrieved relevant documents are re-ranked based on their content similarity and this process is iterated until user expected documents are retrieved.

4. CANDIDATE TERM SELECTION FROM THE TOPICS

The topics generated by LDA, are used in selecting the candidate terms from the topics. The candidate term is the term with high probability and meaningful to the user query. The candidate term are selected from the topics. Let T be the set of generated topics. The candidate terms are selected from each topic using Equation (1)-(4). For example, consider the topic 94 named as “cancer”. 
The list of 20 related terms under this topic are patients, treatment, disease, drug, cancer, etc. A candidate term $C$ is associated with a multinomial distribution of words given by $P(C|Q)$, where $Q$ is a user query context for retrieval from the set of documents. Using KL divergence [44] measure, the $C$ high probable words are determined and sorted. The KL divergence of $Q$ from $P$ is defined as

$$D_{KL}(P||Q) = \sum_x p(x) \log \frac{p(x)}{q(x)}$$  \hspace{1cm} (1)$$

$D_{KL}(C||T_k)$ is the set of $K$ topics

$$D_{KL}(C||T_k) = \sum_w p(w|T_k) \log \frac{p(w|T_k)}{p(w|C,Q)}$$  \hspace{1cm} (2)$$

$\sum p(w|T_k) \log \frac{p(w|C,Q)}{p(C|Q) p(w|T_k)}$  \hspace{1cm} (3)$$

$\sum p(w|T_k) \text{PMI}(w,C|Q)$  \hspace{1cm} (4)$$

$T_k$ in Equation (2), (3) and (4) is the candidate term to be identified from each topic $Q$ in Equation (2), (3) and (4) is the user query context $w$ in Equation (2), (3) and (4) is the word in the topic

The candidate term $C$ and the terms in the topic model over the query represent the Point wise Mutual Information (PMI) in Equation (4).

The PMI of two words measure the semantic relationship between them. Equation (4) assigns greater weights to a candidate term if it has a stronger semantic relationship to the topic words. Hence, selected candidate terms are better representative of the entire topic $T_k$. The selected terms are sorted in decreasing order according to their relevance. The ranked candidate terms that are above the threshold value of 0.5 are considered as suggestion terms. These suggestion terms are
presented to the user to disambiguate the queries and to focus their search on user interest.

5. DOCUMENT RE-RANKING

The rank assigned by the search engine seems to be imperfect. The topic-based approach re-ranks the documents by boosting the initial weight of the documents. The low-ranked documents may be more relevant to the user. Therefore, the rank of such documents is boosted to a higher value. This is achieved by adding the initial position of the document with the highest score of the content similarity of the group of documents. Latent Dirichlet Allocation [36] is used in this topic-based approach for document re-ranking. For a given query q, set of initial results of top N documents are retrieved from the search engine. The similarity between the query and the document are computed based on the latent semantic information. The group of documents dealing with the same topic shares a strong similarity with the query. Such kind of documents are grouped together using cosine similarity [46] measure using Equation (5)

\[ s_d = \sum_{d=1}^{N} \cos(q,d) = \frac{d \cdot q}{\|d\| \cdot \|q\|} \]  

Equation (5) is the similarity score of each document. N in Equation (5) is the number of documents in the cluster and cos(q,d) is the cosine similarity of query q and document d. The rank of the document Rd is computed from the position of the document in the list which is returned by the search engine to achieve independence of the actual ranking score computed by the search engine using equation (6).

\[ R_d = \frac{N \cdot \text{Pos}(d) + 1}{N} \]  

Equation (6) is the position of document d in the query result list returned by the search engine and N is the size of the list. Thus, the first document gets a rank R_d = 1, while the last document is assigned 1/N as the rank, R_d. The rank of each cluster R_c is computed as the average of the rank of all its documents as given in Equation (7).

\[ R_c = \frac{1}{N} \sum_{d=1}^{N} R_d \]  

Equation (7) is the rank of the document d, N is the number of documents in the cluster and R_c is the score of each cluster. Each document is re-ranked using Equation (8).

\[ RRS_d = (1-\lambda)R_d + \lambda R_c \]  

The re-ranking RRS_d is done based on the initial rank of the document represented by R_d and the latent document re-ranking RRS_d using Equation (9).

\[ RRS_d = \frac{\sum_{d=1}^{N} R_d + R_c}{N} \]  

The parameter \( \lambda \) determines when re-ranking process should be done. \( \lambda \) takes either 0 or 1. If \( \lambda = 0 \) the initial rank of the documents is returned; which means no re-ranking is performed. Table 1 illustrate the document re-ranking process. To understand how re-ranking is done, top 5 documents are considered. The rank returned by the search engine for the top 5 documents d1 to d5 are 1, 2, 3, 4 and 5. The rank assigned using this method to the documents d1 to d5 is 0.80, 0.40, 1.0, 0.6 and 0.2. This topic-based approach compute the rank using Equation (6). The documents are clustered using the cosine similarity measure. The documents d2, d4, and d5 are placed in one cluster and the average score of the cluster is 0.40 and documents d1 and d3 are placed in another cluster and their average score is computed using Equation (7) and it is 0.90. The average score of each cluster is added with the initial rank of the documents in each cluster. The documents having low rank in the first retrieval may get high rank in the second retrieval and so on as shown in Table 1. Finally the re-ranked documents are presented to the user as shown in Table 2.

---

**Table 1** A Sample Of Document Re-Ranking

```plaintext
<table>
<thead>
<tr>
<th>Document</th>
<th>Initial Rank</th>
<th>Re-Ranked Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>d1</td>
<td>1</td>
<td>0.80</td>
</tr>
<tr>
<td>d2</td>
<td>2</td>
<td>0.40</td>
</tr>
<tr>
<td>d3</td>
<td>3</td>
<td>1.0</td>
</tr>
<tr>
<td>d4</td>
<td>4</td>
<td>0.6</td>
</tr>
<tr>
<td>d5</td>
<td>5</td>
<td>0.2</td>
</tr>
</tbody>
</table>
```
Re-ranking could aid query expansion, question-answering system and other applications that use IR engines.

6. LDA MODEL

LDA model, an unsupervised machine learning technique is used in identifying the latent topic information from the large document collection.

Table 2 The Re-Ranked Results For The Query “Cancer Patients”

<table>
<thead>
<tr>
<th>Documents</th>
<th>Initial Rank of Doc (search engine)</th>
<th>Rank computed by this Approach</th>
<th>Re-ranked Doc</th>
</tr>
</thead>
<tbody>
<tr>
<td>med.000148</td>
<td>2</td>
<td>0.80</td>
<td>0.80 + 0.90 = 1.70</td>
</tr>
<tr>
<td>med .000066</td>
<td>4</td>
<td>0.40</td>
<td>0.40 + 0.40 = 0.80</td>
</tr>
<tr>
<td>med.000998</td>
<td>1</td>
<td>1.00</td>
<td>1.00 + 0.90 = 1.90</td>
</tr>
<tr>
<td>med.000141</td>
<td>3</td>
<td>0.60</td>
<td>0.60 + 0.40 = 1.00</td>
</tr>
<tr>
<td>med.000229</td>
<td>5</td>
<td>0.20</td>
<td>0.20 + 0.40 = 0.60</td>
</tr>
</tbody>
</table>

Table 2 The Re-Ranked Results For The Query “Cancer Patients”

[16, 37]. LDA is based on bag-of-words assumption, where each document is represented as a vector of word count. Based on this assumption, each document is represented as probability distribution over a set of topics, while each topic is represented as probability distribution over a set of words. A generative process generates the documents as follows.

- For each document, a topic is chosen from its distribution over topics.
- A word is sampled from the distribution over the words associated with the chosen topic.

The process is repeated for all the words in the documents. A collection of D documents is associated with a multinomial distribution over T topics, denoted as Θ. Each topic is defined to be a discrete distribution over words from finite lexicon Φ. For each word in a document d, a topic Z is sampled from the multinomial distribution Θ associated with the document, and a word w from the multinomial distribution Φ associated with topic Z is sampled consequently. This generation process is repeated N times, where N is the total number of words in the document d. Θ and Φ have Dirichlet prior parameters α and β respectively. The graphical representation of LDA model is shown in Figure 2. An arrow represents conditional dependency between variables and the boxes represent repeated sampling with the number of repetitions given by the variables in the bottom of the corresponding box. The two parameters of the model are document-topic distributions Θ, and T the topic-word distribution Φ. Based on these two parameters, word belonging to topic and topic belonging to documents are found. In this study Gibbs sampling has been applied for model parameter estimation and parameter values are chosen as 0.5 for α and 0.1 for β. JAVA API is used to generate latent topic information from the given text corpus. The result is represented as three matrices DT, WT, Z.

DT, is a matrix, where D is the number of documents in the text corpus and T is the number of topics. Matrix DT_{ij} contain the number of times a word in document d has been assigned to topic T_j. Each row represent the number of documents and each column represent the number of topics.
The matrix $WT$ contains $W$ unique words used in the text corpus and $T$ is the number of topics. The matrix $WT_{ij}$ represent the count of how many times word $W_i$ is assigned to topic $T_j$. Each row represent the number of topics and each column represent the number of words. $Z$ is a $1*N$ vector, where $N$ is the total number of words in the text corpus, $Z_i$ is the topic assignment for word $w_i$. Thus, the likelihood of generating a corpus is given by Equation (10)

\[
P(D|\alpha, \beta) = \prod_{n=1}^{N} \prod_{i=1}^{V} \prod_{j=1}^{T} \frac{\Gamma(\sum_{k=1}^{K} Z_{nk} + \alpha_j) \Gamma(\sum_{k=1}^{K} Z_{nk}) \Gamma(n \beta_i) \Gamma(\sum_{k=1}^{K} Z_{nk} \beta_i)}{\Gamma(\sum_{k=1}^{K} Z_{nk} + \alpha_j \beta_i) \Gamma(\sum_{k=1}^{K} Z_{nk}) \Gamma(n \beta_i) \Gamma(\sum_{k=1}^{K} Z_{nk} \beta_i)}
\]

\[ \text{(10)} \]

7. EXPERIMENTAL RESULTS

Classic 3 benchmark dataset has been used for analysis. This dataset consists of four different collections of documents namely, CACM, CISI, CRAN and MED. There are 7095 documents. The statistics of the dataset is given in Table 3. LDA is conditioned on three parameters i.e., Dirichlet hyper-parameter $\alpha$, $\beta$ and topic $T$. The best suggestion results were obtained at 1000th topic run for $T=100$, $\alpha=0.5$ and $\beta=0.1$. For topic model at 1000th topic run, best suggestion results were obtained. The performance of the information retrieval system is evaluated using precision, recall and F-measure as given by equation (11), (12) and (13). Precision measures the exactness of the search, (i.e., the percentage accuracy of the retrieved documents. Recall measures the completeness of the search,(i.e., the percentage of the relevant documents retrieved. When designing a web search engine to compete with Google or any other search engine, the precision at low recall is far more important than the precision at high recall since most people will only look at the first page or two of search results, not the 1000th page. So Precision at low recall is well suited for any search engine.

\[
\text{Precision} = \frac{\text{Retrieved relevant documents}}{\text{Retrieved documents}}
\]

\[ \text{(11)} \]

\[
\text{Recall} = \frac{\text{Retrieved relevant documents}}{\text{All relevant documents}}
\]

\[ \text{(12)} \]

\[
F = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}
\]

\[ \text{(13)} \]

The precision and recall graph based on Topic, Cluster and Keyword based approach is given in Figures 3 and 4 and shows the results for different values of $N$, ranging from 500 to $3000$, where $N$ is the number of documents.

Liu Z (2011) [48] developed iterative single-keyword refinement (ISKR) and partial elimination based convergence(PEBC)algorithm. The algorithm generated expanded queries from the clustered results which showed significant improvement as given in Figure 5. The topic based approach suggest terms related to only particular topic. For instance in Figure 5, q1, q2, q3 are the terms that are related to three topic namely, cell is a small compartment, cell is an electrical cell and cell is a mobile phone. This improves the precision of the results when expanding the query with these terms. Glori B et al (2012)[38] claims that their approach generate a greater number of suggested queries with respect to Google and achieved better disambiguated suggested queries as given in Figure 6.
The Topic based approach is compared with the existing approaches [48] and [38] and given in

<table>
<thead>
<tr>
<th>Original query</th>
<th>Google Extended queries</th>
<th>Matrioshka suggested Queries</th>
<th>Topic based query suggestion</th>
</tr>
</thead>
<tbody>
<tr>
<td>cancer</td>
<td>types of cancer;cancer;astrology;cancer;horoscope;lung;cancer;journal;causes of cancer;cancer symptoms</td>
<td>Cancer Symptoms;Treatment:Cancer Prevention;Cancer Astrology;Sign:Cancer News:Cancer Society:Cancer National Institute:Cancer Health:American Society</td>
<td>Cancer:oncology,chemotherapy,radiation,medical,Screening,test,treatment,disease Cancer:astrology:zodiac,sign,astrology,sun,celestial longitude,plane</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Technique</th>
<th>Suggested queries</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISKR</td>
<td>q1: “Cell,express,data” q2: “Cell,biological” q3: “Cell,battery”</td>
</tr>
<tr>
<td>PEBC</td>
<td>q1: “Cell,express” q2: “Cell,language” q3: “Cell,battery”</td>
</tr>
<tr>
<td>CS</td>
<td>q1: “Cell,biphosphate,placent,mosaic” q2: “Cell,sumono,yumeka,template” q3: “Cell,battery,kinase,amala”</td>
</tr>
<tr>
<td>Data Clouds</td>
<td>q1: “Cell, multicellular” q2: “Cell,bit” q3: “Cell, stomach”</td>
</tr>
<tr>
<td>Topic based Approach</td>
<td>q1: “Cell, compartment, honeycomb,jail,room” q2: “device,battery,chemical,Eletrical cell” q3: “cellular Téléphone, cellphone, phone, radiotelephone,transmitter,receiver”</td>
</tr>
</tbody>
</table>

![Figure 4](image4.png) Comparison Of Topics, Cluster And Keyword Based Retrieval Recall Graph
Figures 5 And 6. Topic Based Approach Generate Better Disambiguated Suggested Queries Related To Each Topic And Is Given In Figure 7. The Topic based approach improves precision by 20% than the cluster based approach as in Table 4. The comparison of the result for the query “cancer” from Figure 6 show that the topic based method generate queries related to only cancer topic. But, cluster based method generates both more general and specific queries also. The cluster based approach produce the cluster label, whereas, the topic based approach give terms related to that topic for query suggestion. For example, for the original query “cancer” the cluster based approach of Glori B et al (2012) suggests the extended query as cancer symptoms treatment, cancer prevention and general query as cancer astrology signs and cancer news. The Topic based approach helps in narrowing the web search.

Table 4 Comparison Of Keyword, Cluster And Topic Based Retrieval

<table>
<thead>
<tr>
<th></th>
<th>Keyword</th>
<th>Cluster</th>
<th>Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average Precision</td>
<td>0.43</td>
<td>0.52</td>
<td>0.70</td>
</tr>
<tr>
<td>Average F-Measure</td>
<td>0.40</td>
<td>0.48</td>
<td>0.62</td>
</tr>
</tbody>
</table>

8. CONCLUSIONS AND FUTURE ENHANCEMENT

This work is a novel approach to build disambiguated queries using topic model for web search. This approach provides suggested list of meaningful queries for expansion and to make the user search more focused. The contribution of this work is in identifying the high probable words from each topic and re-ranking the documents. The topic distribution model using LDA gives accurate and fast suggestion for short queries irrespective of whether the input query appear in the query log or not. In future this work can be easily extended to the Question and Answer system to match the question issued by the user.

REFERENCES:


[12] Zhang, B., Li, H., Liu, Y., Ji, L., Xi, W., Fan, W., Chen Z and Ma, W-Y., “Improving web search results using affinity graph”. In proceedings of the 28th annual international ACM SIGIR conference on research and development in information retrieval, Salvador, Brazil, 2005, ACM.


[34] Diaz, F, “Regularizing ad hoc retrieval scores”, Proceedings of the 14th ACM international conference on Information and
knowledge management, Bremen, Germany, ACM, 2005, pp. 672-679.


