Query Suggestions in the Absence of Query Logs

By Sumit Bhatia, Debapriyo Majumdar,Prasenjit Mitra

SIGIR'11, July 24–28, 2011, Beijing, China.

Presented by : Ujjwal Acharya

Overview

- How to provide a meaningful query suggestions in absence of query log ?
- So how do you define a meaningful queries?
- A meaningful query must infer the user's query intent and information needs and must help user find the relevant documents containing relevant information.
- For fulfilling this need, a document centric probabilistic mechanism to generate query suggestions that does not depend on query logs has been proposed.

Related Works

- Most of the previous works provide query expansion and query refinement rather than query suggestions.
- **CompleteSearch Method**:
  - Provides real time auto-completion of the last query term typed by the user.
  - Requires user to type at least two characters of the last query term.
- **SimSearch Method :**
  - Phrase index is searched to find all the phrases that contain the user submitted partial query as a sub phrase.
  - The selected phrases are presented to the user if the words in the query are in the same order as in the phrases.

Proposed Approach

- Document centric probabilistic mechanism
- Phrase Extraction to create a database of phrases that can be used for completing partial user queries from document corpus.
- Using N-grams of all order 1, 2 and 3.
- Used idea similar to skip grams rather than N-grams.
- Now, N-gram is the number of non stop-words.

Method for Query Suggestions

To solve this problem, we ask this question: *Given a partial query* Q_1^k *and a phrase* $p_i \in \mathcal{P}$, *what is the probability* $P(p_i|Q_1^k)$, *i.e., the probability that the user will eventually type* p_i *after typing* Q_1^k?

Using Bayes' theorem, the probability $P(p_i|Q_1^k)$ can be written as:

$$P(p_i|Q_1^k) = \frac{P(p_i) \times P(Q_1^k|p_i)}{P(Q_1^k)} \qquad (2)$$

Method for Query Suggestions(cont'd

We start by making the observation that at any given instant of time, Q_1^k can be decomposed as follows:

$$Q_1^k = Q_c + Q_t \tag{1}$$

Our proposed model for query suggestion is given below

$$P(p_i|Q_1^k) \overset{rank}{=} \underbrace{P(p_i|Q_t)}_{\text{phrase selection probability}} \times \underbrace{P(Q_c|p_i)}_{\text{phrase-query correlation}}$$

Estimating Phrase Selection Probability

- The probability of selecting a phrase given a partial word is expressed as follows.

$$P(p_{ij}|Q_t) = \underbrace{P(c_i|Q_t)}_{\text{term completion probability}} \times \underbrace{P(p_{ij}|c_i)}_{\text{term to phrase probability}}$$

Estimating Phrase-Query Correlation

- This takes in account the relationship between a phrase and user submitted query.

$$P(Q_c|p_i) = \frac{P(Q_c, p_i)}{P(p_i)}$$

- The above probability determines given that Pi is the second half of the complete query, if Qc represents the first half of the complete query.

Estimating Phrase-Query Correlation (cont'd)

- Both of the probabilities in the previous equation can be estimated using the corpus as follows:

$$P(Q_c|p_i) = \frac{|D_{Q_c} \cap D_{p_i}|}{|D_{p_i}|}$$

- Here, Dpi and DQc represent the sets of documents that contain phrase pi and Qc respectively.

Datasets used

- Two datasets were used namely
- **TREC:**
  - Consists of more than 200,000 news articles published in Financial Times between years 1991–1994.
- **Ubuntu:**
  - Consists of more than 100,000 discussion threads crawled from ubuntuforums.org, a set of 25 queries and relevance judgments.

Baselines Methods

- The proposed methods was compared with the following baseline methods

- **Similarity based phrase search(SimSearch):**

- **CompleteSearch (CompSearch):**

Test Queries

- Generated 40 partial test queries for each dataset.
- **Type-A Queries:** These queries were generated by retaining only the first keyword.
- **Type-B Queries:** These queries were generated by retaining the first keyword of the query followed by the first k characters of the remaining query string ($2 \leq k \leq$ length of the remaining query string)

Test Queries(cont'd)

Dataset	Original Query	Type-A Query	Type-B Query
Ubuntu	automount hard drive partitions	automount	automount hard drive part
	virtualbox keyboard problem	virtualbox	virtualbox keyb
	wine microsoft office	wine	wine microso
TREC	falkland petroleum exploration	falkland	falkland petro
	encryption equipment export	encryption	encryption equip
	radioactive waste	radioactive	radioactive was

Table 2: Examples of queries used in experiments.

Evaluation

- For each test query, they collected the top 10 suggestions generated by the three methods (SimSearch, CompSearch and proposed method (Probabilistic)).
- Evaluation was performed with the help from 12 volunteers who were their colleagues and were not associated with the project.

Rating	Meaning
Y	Yes, a meaningful suggestion
N	No, not a meaningful suggestion, or badly formed as a query
D	An (almost) duplicate suggestion, conveys no new information
??	Not sure

Table 3: Different rating options available to users and their meanings.

Results And Discussions

Query = falkland		
SimSearch	**CompSearch**	**Prob**
falklands	falklands	falklands
falkland	falkland	falklands war
falkland islands	falklanders	falkland islands
falklands war		falklands conflict
falklands conflict		1982 falklands
1982 falklands		1982 falklands conflict
falkland islands government		falkland islands government
1982 falklands conflict		falklands war in 1982
falkland arms		1982 falklands war
falklanders		invasion of the falklands
Query = encryption equip		
<No Suggestions Produced>	encryption equipment	encryption equipment
		encryption digital equipment
		encryption office equipment
		encryption electronic equipment
		encryption telephone equipment
		encryption equipment and services
		encryption video equipment
		encryption medical equipment
		encryption transmission equipment
		encryption original equipment

Success Rate of Different Methods

A query suggestion method is successful for a given partial query if it is able to generate at least one meaningful suggestion for the partial query.

	Ubuntu		
	SimSearch	**CompSearch**	**Probabilistic**
Type-A	1.00	1.00	1.00
Type-B	0.75	1.00^{s}	1.00^{s}
Overall	0.875	1.00^{s}	1.00^{s}
	TREC		
	SimSearch	**CompSearch**	**Probabilistic**
Type-A	1.00	1.00	1.00
Type-B	0.15	0.95^{S}	1.00^{S}
Overall	0.575	0.975^{S}	1.00^{S}

Table 4: Success Rate of different query suggestion methods for the two datasets. Superscripts s and S indicate statistically significant improvements over SimSearch with $p < 0.05$ and $p < 0.01$, respectively (one-tailed t-test).

Quality of Suggestions

Ubuntu			
	SimSearch	**CompSearch**	**Probabilistic**
Type-A	0.4597	0.1638	$\mathbf{0.5022^{C}}$
Type-B	0.2193	0.2309	$\mathbf{0.4746^{SC}}$
Overall	0.3395	0.1974	$\mathbf{0.4884^{SC}}$
TREC			
	SimSearch	**CompSearch**	**Probabilistic**
Type-A	0.5429	0.2709	$\mathbf{0.5697^{C}}$
Type-B	0.0614	0.2010	$\mathbf{0.3975^{SC}}$
Overall	0.3022	0.2359	$\mathbf{0.4836^{SC}}$

Table 6: Mean Average Precision (MAP) values achieved by different query suggestion methods for the two datasets. Superscripts S and C indicate statistically significant improvements over SimSearch and CompSearch methods, respectively (one tailed t-test, $p < 0.01$).

Retrieval Effectiveness of Suggested Queries

- Query clarity score is used to measure the retrieval performance of suggested queries.
- Clarity score of a query increases if we add terms that reduce query ambiguity and it decreases on adding terms that make the query more ambiguous.
- Mathematically, clarity score for a query q with respect to a collection of documents C is given by

$$\text{Clarity}(q, \mathcal{C}) = \sum_{v \in \mathcal{V}} P(v|q) \log_2 \frac{P(v|q)}{P(w|\mathcal{C})},$$

- Where V is the vocabulary of the collection.

Conclusions and Future Works

- Meaningful query suggestions can be made in the absence of query logs with an unsupervised probabilistic approach using the occurrence of terms and phrases in a corpus of documents.
- **Future works**
  - One possible future goal would be to ensure that the badly formed combination of phrases are eliminated from the suggestions.
  - Use of synonyms and synonymous phrases to enable the system to suggest alternatives also needs to be explored.
  - Systematic approach towards diversifying the suggested queries
  - Apply to a relatively larger scale.