Representing Queries as Distributions

Xiaobing Xue W. Bruce Croft
Center for Intelligent Information Retrieval
Computer Science Department
University of Massachusetts, Amherst, MA, 01003, USA
{xuexb,croft}@cs.umass.edu

ABSTRACT

Representing a query appropriately helps model the underlying information need and thus improves the retrieval performance. Previous query representations either generate related words and phrases to augment the original query but ignore how these words and phrases fit together in new queries, or apply a specific reformulation operation to the original query but ignore alternative operations. In this paper, a novel representation is proposed as a distribution of queries, where each query is a variation of the original query. This representation, on one hand, considers a query as a basic unit and thus captures important dependencies between words and phrases in the query. On the other hand, it naturally combines different reformulation operations as possible ways to generate variations of the original query. This query distribution representation is carefully compared with previous query representations in this paper to show its advantages. Some recent work using this representation has shown promising results and is briefly described here.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval

General Terms

Algorithms, Experimentation, Performance

Keywords

Query Representation, Query Substitution, Query Segmentation, Information Retrieval

1. INTRODUCTION

In a typical search scenario, users pose keyword queries to express their information needs. Due to vocabulary mismatch and ambiguity, it is sometimes difficult to retrieve relevant documents using the original query. In response, techniques for generating new queries to improve retrieval performance have been developed. The new queries can be considered as a way to model the information need underlying the original keyword queries.

Many previous models have focused on generating related words and phrases to expand the original query. For example, the relevance model approach [4] adds new words to the original query, the sequential dependence model [5] adds phrase structure, and the latent concept expansion model [6] adds new term proximity features and words. Generally, the query representation generated by these models is a large, possibly weighted, "bag of terms" that contains new words and phrases. This representation, however, does not consider how these new terms can be used together in actual queries that are variations of the original query and thus misses important dependencies.

Other research on web query reformulation has tended to generate a single new query (e.g. [2][3]) by applying a specific reformulation operation. Different operations have been studied. *Query segmentation* [2] tries to detect underlying concepts in keyword queries and annotate those concepts as phrases. For example, given the query "oil industry history", query segmentation techniques may detect "oil industry" as a concept and annotate it as a phrase in the new query "(oil industry) history". *Query substitution* [3] tries to change some words of the original query to bridge vocabulary mismatch. For example, the query "oil industry history" could be changed to "petroleum industry history", since some relevant documents may contain "petroleum industry" instead of "oil industry". However, this single reformulated query representation does not consider combining different operations from a unified perspective, thus important information about alternative query reformulations is not captured.

In this paper, we propose a novel query representation that transforms the original query into a distribution of reformulated queries. A reformulated query is generated by applying different operations including adding or replacing query words, detecting phrase structures, and so on. Since the reformulated query that involves a particular choice of words and phrases is explicitly modeled, this representation captures dependencies between those query components. On the other hand, this framework naturally combines query segmentation, query substitution and other possible reformulation operations, where all these operations are considered as methods for generating reformulated queries. In other words, a reformulated query is the output of applying single or multiple reformulation operations. The probabilities of alternative reformulated queries can then be estimated within the same framework.

The Translation Model [1] is a special model that does not generate any explicit query representation. Instead, the word-to-word translation probabilities reflecting the relationships between the original query words and new words are directly embedded into the retrieval model. It is interesting to understand the connections between the Translation Model and the query distribution representation.

The rest of paper is organized as follows: Section 2 reviews existing query representations; Section 3 describes the proposed query distribution representation and compares it with other representations to show its advantages; Section 4 makes a comparison with Translation Model; Section 5 briefly describes our recent work that uses this novel representation and the last section concludes the paper.

2. EXISTING QUERY REPRESENTATIONS

In this section, we first introduce the necessary notations and then review two query representations widely used in previous work.

$C = \{D_i\}_{i=1}^{|C|}$ denotes a collection of documents, where D_i is the document. $V_W = \{w_i\}_{i=1}^{|V_W|}$ denotes a vocabulary of words in the collection C, where w_i is a word. $|V_W|$ denotes the size of V_W. $Q = \{q_i\}_{i=1}^{m}$ denotes the original keyword query posed by the user, where q_i is the query word. Usually, q_i belongs to V_W.

The Language Modeling approach [7][12] is a retrieval model widely used in information retrieval area, where the ranking score of a document is calculated as the probability of generating the query from this document. All models discussed in this paper use the language modeling approach. Given the original query Q, the documents in the collection are ranked by $P(Q|D)$, which are calculated as follows:

$$P(Q|D) = \prod_{i=1}^{m} P(q_i|D) = \prod_{i=1}^{m} [\lambda P_{ml}(q_i|D) + (1-\lambda) P(q_i|B)] \tag{1}$$

where $P(q_i|D)$ is estimated by a mixture of the maximum likelihood estimation $P_{ml}(q_i|D)$ and the background model $P(q_i|B)$ [12]. λ is the mixture parameter.

2.1 Distribution Of Terms

The first query representation discussed is a distribution of terms, which is denoted as **DOT**. Besides the original query words, the terms in this distribution include query phrases [5], new words [4][6] and new phrases [6]. Formally, this representation is defined as follows.

Given a vocabulary of terms $V_T = \{t_i\}_{i=1}^{|V_T|}$, where t_i is a term, the query representation is a distribution over V_T, i.e. $\mathbf{P}_T = \{(P(t_i|Q) \quad t_i)\}_{i=1}^{|V_T|}$, where $P(t_i|Q)$ is the probability assigned to t_i.

When $\mathbf{P}_T$ is used for retrieval, the retrieval score of a document D is calculated in Eq. 2.

$$P(\mathbf{P}_T|D) = \prod_{i=1}^{|V_T|} P(t_i|D)^{P(t_i|Q)} \tag{2}$$

Compared with Eq. 1, Eq. 2 transforms the original keyword query Q into a distribution of terms $\mathbf{P}_T$ and assigns the term weight $P(t_i|Q)$ to $P(t_i|D)$.

Specifically, different models define their own V_T. The Relevance Model (RM) [4] defines V_T as a vocabulary of words from the collection (V_W) that include the original query words. The Sequential Dependence Model (SDM) [5] defines V_T as a vocabulary of query words and query phrases from the original query.

2.2 Single Reformulated Query

The second query representation discussed is a single new query generated by applying a specific reformulation operation, which is denoted as **SRQ**. Formally, $Q_r^\star(Oper)$ denotes the new query generated after applying the reformulation operation *Oper* to the original query Q. $Q_r^\star(Oper)$ is simplified as $Q_r^\star$ if the operation *Oper* is not explicitly mentioned. When $Q_r^\star$ is used for retrieval, the score of the document D is calculated by $P(Q_r^\star|D)$.

Different reformulation operations have been studied. Query segmentation [2][8] segments the original query Q to detect its phrase structure and obtains the segmented query $Q_r^\star(Seg)$. Query substitution [3][9] replaces some original query words with new words to bridge vocabulary mismatch and produces the new query $Q_r^\star(Sub)$.

3. DISTRIBUTION OF QUERIES

In this section, a novel query representation is proposed, where the original keyword query is transformed into a distribution of reformulated queries. A reformulated query is the output of applying single or multiple reformulation operations. This representation is denoted as **DOQ**.

Formally, we first generate a vocabulary of reformulated queries $V_{Q_r} = \{Q_{r_i}\}_{i=1}^{|V_{Q_r}|}$, where Q_{r_i} is a reformulated query. Then, the original query Q is transformed into a distribution over V_{Q_r}, i.e. $\mathbf{P}_{Q_r} = \{(P(Q_{r_i}|Q) \quad Q_{r_i})_{i=1}^{|V_{Q_r}|}\}$. Here, $P(Q_{r_i}|Q)$ is the probability corresponding to Q_{r_i}. The representation itself does not specify how to generate reformulated queries and how to estimate the probability for each reformulated query. Different strategies can be adopted based on different implementations.

Given this query distribution based representation, i.e. $\mathbf{P}_{Q_r} = \{(P(Q_{r_i}|Q) \quad Q_{r_i}\}_{i=1}^{|V_{Q_r}|}$, the retrieval score of a document is calculated in Eq. 3.

$$P(\mathbf{P}_{Q_r}|D) = \prod_{i=1}^{|V_{Q_r}|} P(Q_{r_i}|D)^{P(Q_{r_i}|Q)} \tag{3}$$

where $P(Q_{r_i}|D)$ is the probability of generating Q_{r_i} from the document D. The estimation of $P(Q_{r_i}|D)$ depends on the implementation.

3.1 Comparison of Representation

In this subsection, we use the TREC query "oil industry history" as an example to compare different query representations. The representations generated by different models are displayed in Table. 1.

For the *Distribution of Term* (TOD) representation, RM outputs a distribution of words that includes the original query words such as "oil", "industry" and "history" and new words like "gas" and "petroleum". SDM outputs a distribution of original query words and phrases. Besides the original query words, it also includes phrases such as "oil industry" and "industry history".

For the *Single Reformulated Query* (SRQ) representation, query segmentation techniques [2][8] generate a segmented

Table 1: Representations generated by different models for the original query "oil industry history"

Type	Model	Representation
Term Distribution (DOT)	RM [4]	0.44 "industry", 0.28 "oil", 0.08 "petroleum", 0.08 "gas", 0.08 "county", 0.04 "history"...
	SDM [5]	0.28 "oil", 0.28 "industry", 0.28 "history", 0.08 "oil industry", 0.08 "industry history"...
Single Reformulated Query (SRQ)	Segmentation [2][8]	"(oil industry)(history)"
	Substitution [3][9]	"oil and gas industry history"
Query Distribution (DOQ)	0.28 "(oil industry)(history)", 0.24 "(petroleum industry)(history)", 0.20 "(oil and gas industry)(history)", 0.18 "(oil)(industrialized)(history)"...	

query "(oil industry)(history)". Query substitution techniques [3][9] generate a substituted query "oil and gas industry history" where "oil industry" is replaced by "oil and gas industry".

For the *Distribution Of Query* (DOQ) representation, we first generate a set of reformulated queries such as "(oil industry)(history), (petroleum industry)(history), (oil and gas industry)(history), (oil)(industrialized)(history)". Here, a reformulated query is generated by first applying query substitution and then applying query segmentation. For example, the original query is first substituted as "petroleum industry history" and then it is segmented as "(petroleum industry)(history)". In this way, different reformulation operations are naturally combined within this representation. Then, we estimate the probability for each reformulated query.

We first compare DOT with DOQ. DOT augments the original query with a bag of new terms but does not consider how to fit these terms together to form actual queries. In contrast, DOQ augments the original query with a set of new queries, which captures the important dependencies between terms. This difference is reflected on the new terms added by these two representations, either directly or through adding queries that contain the new terms. DOT adds a new term t according to its own relationship with the original query Q (i.e. $P(t|Q)$), while DOQ adds a new term according to the relationships between the query containing this term Q_r and the original query Q (i.e. $P(Q_r|Q)$), where considering Q_r as a whole captures dependencies between terms in Q_r. As shown in Table 1, RM (a representative of DOT) assigns high probability for "county" while DOQ does not, since "county" frequently cooccurs with the original query but it is not usually found in reformulated queries. On the other hand, DOQ provides high probability for "industrialized" while RM does not, since "industrialized" can be used in queries such as "(oil)(industrialized)(history)" but it rarely cooccurs with the original query.

Second, we compare SRQ with DOQ. SRQ and DOQ both consider a query as a basic unit. However, SRQ only uses a single reformulated query that is returned by applying a specific operation, while DOQ generates a variety of reformulated queries where each is the output of applying single or multiple operations. Therefore, DOQ is more general than RQ and considers valuable information of alterative reformulated queries. As shown in Table 1, segmentation techniques return a single segmented query "(oil industry)(history)" and substitution models return a single substituted query "oil and gas industry history". In contrast, QD generates a couple of reformulated queries, where each is the output of applying both the segmentation operation and the substitution operation such as "(petroleum industry)(history)".

3.2 Comparison of Retrieval Scores

In this subsection, we further compare the retrieval scores of using different representations to better understand their differences.

We first compare the retrieval scores of DOT and DOQ. In order to compare Eq. 2 and Eq. 3, some additional assumptions are made for DOQ. Note that these assumptions are not necessary for DOQ. First, we assume the reformulated query Q_r only contains terms t from the vocabulary V_T. This assumption is used to match the vocabulary used by DOT. Second, we assume $P(Q_{r_i}|D) = \prod_{t \in Q_{r_i}} P(t|D)$, which is required by DOT.

Given these two assumptions, Eq. 3 can be written as follows:

$$
\begin{aligned}
P(\mathbf{P}_{Q_r}|D) &= \prod_{i=1}^{|V_{Q_r}|} P(Q_{r_i}|D)^{P(Q_{r_i}|Q)} \\
&= \prod_{i=1}^{|V_{Q_r}|} (\prod_{t \in Q_{r_i}} P(t|D))^{P(Q_{r_i}|Q)} \qquad (4) \\
&= \prod_{i=1}^{|V_T|} P(t_i|D)^{\sum_{Q_r \in \{Q_r | t_i \in Q_r\}} P(Q_r|Q)} \qquad (5)
\end{aligned}
$$

Eq. 4 is obtained by using the second assumption. Since each Q_{r_i} in V_{Q_r} only contains terms from the vocabulary V_T by the first assumption, we can reorganize Eq. 4 to obtain Eq. 5 by merging the same term t_i together. In Eq. 5, $\{Q_r | t_i \in Q_r\}$ denotes the set of reformulated queries containing t_i.

Comparing Eq. 2 with Eq. 5, it is not difficult to find that the retrieval scores used by DOT and DOQ are both the weighted geometric mean of $P(t_i|D)$ for all t_i in the vocabulary V_T. The difference is how the weights for $P(t_i|D)$ are calculated. The weight assigned by DOT is $P(t_i|Q)$ which calculation is independent of other terms in V_T. In contrast, the weight assigned by DOQ is $\sum_{Q_r \in \{Q_r : t_i \in Q_r\}} P(Q_r|Q)$ that determines the weight of $P(t_i|D)$ by considering all queries containing t_i. The estimation of $P(Q_r|Q)$ considers Q_r as a whole and thus captures the relationships between t_i and other terms in Q_r. We can further prove the following claim, which formally explains the differences between DOT and DOQ. The details of the proof are omitted due to space limitations.

CLAIM 1. *If the estimation of $P(Q_r|Q)$ does not consider Q_r as a whole, i.e. $P(Q_r|Q) = \prod_{t \in Q_r} P(t|Q)$, DOQ is equal to DOT, i.e. $\sum_{Q_r \in \{Q_r : t_i \in Q_r\}} P(Q_r|Q) = P(t_i|Q)$.*

Second, we compare the retrieval scores of SRQ and DOQ. SRQ uses the probability $P(Q_r^\star|D)$ as the retrieval score, where $Q_r^\star$ is the single reformulated query generated by applying a specific operation. Comparing it with Eq. 3, it is clear that SRQ is a special case of DOQ, where SRQ assigns all probabilities to $Q_r^\star$ and assigns zero probability to alternative reformulated queries.

Table 2: Summary of the comparisons

vs.	DOQ
DOT	ignores the dependencies in Q_r
SRQ	ignores alternative Q_r
TM	ignores the dependencies in both Q_r and Q

4. TRANSLATION MODEL

The Translation Model (TM) [1] is a special model that does not generate any explicit query representation for the original keyword query. Instead, it directly incorporates word-to-word translation probabilities into the retrieval model. These word-to-word translation probabilities reflect the relationships between the original query words and new words, thus the Translation Model implicitly augments the original query with new words. It is interesting to understand the connections between the Translation Model and the proposed query distribution representation.

Formally, the word-to-word translation probabilities are denoted as $P(q_i|w_j)$, where $q_i \in Q$ and $w_j \in V_W$. The retrieval score of a document D is ranked by the translation probability from D to the query Q, i.e. $P(Q|D)$, which is calculated as follows:

$$P(Q|D) \quad = \quad \sum_{a_1=1}^{|D|} ... \sum_{a_m=1}^{|D|} [(\frac{1}{|D|})^m \prod_{i=1}^{m} P(q_i|D_{a_i})] \qquad (6)$$

where $(a_1, ..., a_m)$ is a group of alignment variables. a_i indicates the query word q_i is translated from the word D_{a_i} of the document D. In other words, a_i denotes the position of the word in D that q_i is translated from. $|D|$ is the length of the document D.

After some derivations, Eq. 6 can be transformed into the following equations.

$$\begin{aligned} P(Q|D) &= \sum_{Q_r=q'_1...q'_m} [\prod_{i=1}^{m} P(q'_i|D)][\prod_{i=1}^{m} P(q_i|q'_i)] & (7) \\ &= \sum_{Q_r=q'_1...q'_m} P(Q_r|D)P(Q|Q_r) & (8) \end{aligned}$$

where $Q_r = q'_1...q'_m$ is a reformulated query that consists of m query words $q'_1...q'_m$. The details of derivations from Eq. 6 to Eq. 7 are omitted due to space limitations.

Comparing Eq. 3 with Eq. 8, there are three differences. First, Eq. 3 estimates the probability of generating a new query representation by $P(\mathbf{P}_{Q_r}|D)$, while Eq. 8 estimates the probability of generating the original query by $P(Q|D)$ and considers Q_r as a hidden variable. Second, Eq. 3 calculates a geometric mean of $P(Q_r|D)$, while Eq. 8 calculates a arithmetic mean of $P(Q_r|D)$. Third, the estimation of $P(Q_r|Q)$ in Eq. 3 considers Q_r as a whole, while the estimation of $P(Q|Q_r)$ in Eq. 8 assumes $P(Q|Q_r) = \prod_{i=1}^{m} P(q_i|q'_i)$, which only considers the word-to-word relations and ignores the dependencies within both Q_r and Q.

Table 2 summarizes the comparison of the proposed DOQ representation with DOT, SRQ and TM.

5. RECENT WORK

In this subsection, we briefly describe our recent work that use the query distribution representation.

The first project [10] is on reformulating short queries. The original query is represented as a distribution of reformulated queries, where each reformulated query is generated by applying query substitution and query segmentation operations. Specifically, the passages in the target corpus that contain all or most query words are analyzed to generate reformulated queries and estimate their probabilities.

The second project [11] selects subsets from long queries. The original long query is represented as a distribution of subset queries. We consider the subset selection as a sequential labeling problem and propose a novel Conditional Random Field model to help learn the subset distribution. The proposed model captures the local and global dependencies within the query and directly optimizes the expected retrieval performance on the training set.

6. CONCLUSION

A query distribution representation is proposed in this paper. In order to better understand this novel representation, we compare it with some existing query representations from several aspects. The comparisons show that the query distribution representation captures the dependencies within the reformulated queries that are usually missed by other representations and also considers alternative reformulated queries. Some recent work using this representation is also briefly described.

7. ACKNOWLEDGEMENTS

This work was supported in part by the Center for Intelligent Information Retrieval and in part by NSF grant #IIS-0711348. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect those of the sponsor.

8. REFERENCES

[1] A. Berger and J. Lafferty. Information retrieval as statistical translation. In *SIGIR09*, pages 222–229, Berkeley, CA, 2009.

[2] S. Bergsma and Q. I. Wang. Learning noun phrase query segmentation. In *EMNLP-CoNLL07*, pages 819–826, Prague, 2007.

[3] R. Jones, B. Rey, O. Madani, and W. Greiner. Generating query substitutions. In *WWW06*, pages 387–396, Ediburgh, Scotland, 2006.

[4] V. Lavrenko and W. B. Croft. Relevance based language models. In *SIGIR01*, pages 120–127, New Orleans, LA, 2001.

[5] D. Metzler and W. B. Croft. A markov random field model for term dependencies. In *SIGIR05*, pages 472–479, Salvador,Brazil, 2005.

[6] D. Metzler and W. B. Croft. Latent concept expansion using markov random fields. In *SIGIR07*, pages 311–318, Amsterdam, the Netherlands, 2007.

[7] J. M. Ponte and W. B. Croft. A language modeling approach to information retrieval. In *SIGIR98*, pages 275–281, Melbourne, Australia, 1998.

[8] B. Tan and F. Peng. Unsupervised query segmentation using generative language models and wikipedia. In *WWW08*, pages 347–356, Beijing,China, 2008.

[9] X. Wang and C. Zhai. Mining term association patterns from search logs for effective query reformulation. In *CIKM08*, pages 479–488, Napa Valley, CA, 2008.

[10] X. Xue, W. B. Croft, and D. A. Smith. Query reformulation using passage analysis. Technical report, CIIR, UMass Amherst, 2010.

[11] X. Xue, S. Huston, and W. B. Croft. Selecting subsets of verbose queries using conditional random fields. Technical report, CIIR, UMass Amherst, 2010.

[12] C. Zhai and J. Lafferty. A study of smoothing methods for language models applied to ad hoc information retrieval. In *SIGIR01*, pages 334–342, New Orleans, LA, 2001.