J Intell Inf Syst (2008) 31:265–286
DOI 10.1007/s10844-007-0045-0

A novel retrieval approach reflecting variability of syntactic phrase representation

Young-In Song · Kyoung-Soo Han · Sang-Bum Kim · So-Young Park · Hae-Chang Rim

Received: 16 April 2007 / Revised: 3 August 2007 /
Accepted: 8 August 2007 / Published online: 22 August 2007

Abstract In this paper, we introduce *variability* of syntactic phrases and propose a new retrieval approach reflecting the variability of syntactic phrase representation. With *variability* measure of a phrase, we can estimate how likely a phrase in a given query would appear in relevant documents and control the impact of syntactic phrases in a retrieval model. Various experimental results over different types of queries and document collections show that our retrieval model based on variability of syntactic phrases is very effective in terms of retrieval performance, especially for long natural language queries.

Keywords Information retrieval · Retrieval model · Structural language model · Syntactic phrase · Head-modifier pair · Weighting phrases · Variability of phrase

Y.-I. Song · H.-C. Rim (✉)
Department of Computer Science and Engineering, Korea University,
NLP Lab., 237 Asan Science Bldg., 1, 5-ka, Anam-dong, SeongPuk-gu,
Seoul 136-701, South Korea
e-mail: rim@nlp.korea.ac.kr

Y.-I. Song
e-mail: song@nlp.korea.ac.kr

K.-S. Han · S.-B. Kim
SK Telecom, Seoul, South Korea

K.-S. Han
e-mail: Kyoungsoo.han@gmail.com

S.-B. Kim
e-mail: sangbum.kim@gmail.com

S.-Y. Park
Sang-Myoung University
Seoul, South Korea
e-mail: ssoya@smu.ac.kr

1 Introduction

The bag-of-words (BOW) assumption has been widely used in modern IR models, because it makes the models simple and tractable. However, the assumption is clearly wrong in typical natural language text and sometimes decreases the discriminative power of the retrieval models. For instance, a BOW based retrieval model can not distinguish a difference between the following two queries: *bank terminology* and *terminology bank* (Zhai 1997).

For this reason, there have been several approaches using statistical phrases for information retrieval, such as proximity-based phrase indexing (Fagan 1987), n-gram retrieval model (Miller et al. 1999; Song and Croft 1999), and dependence language modeling approaches for information retrieval (Gao et al. 2004; Metzler and Croft 2005; Srikanth and Srihari 2003). Although these approaches based on statistical phrases are able to capture dependency information between words, they usually generate too many meaningless phrases or cannot identify some important long distance dependencies.

On the other hand, several studies have investigated the usefulness of syntactic phrases for information retrieval (Arampatzis et al. 2000; Pohlmann and Kraaij 1997; Strzalkowski et al. 1994; Fagan 1987). They have focused on developing more sophisticated representation by extracting various kinds of syntactic phrases using linguistic information. Although it is obvious that such syntactic phrases are more meaningful and less noisy than statistical phrases, most large-scale experiments based on syntactic phrases have shown limited improvements in performance. In their experiments, syntactic phrases have been found to be useful for improving the performance of IR based on the BOW assumption. However, the benefit has been only moderate even with long queries, which are known to be effective when dependency is considered in IR (Brants 2004).

There are two obvious reasons for the disappointing results in the retrieval experiments based on syntactic phrases. The first reason is that a proper retrieval model combining single words and syntactic phrases does not exist (Gao et al. 2004). Since phrases and their constituent words are obviously dependent on each other, traditional IR models assuming term independence are inappropriate for using phrases. (Srikanth and Srihari 2003) has also pointed out that previous works using syntactic phrases have the problem of either ignoring the relation between a phrase and its constituent words or considering it in an ad-hoc fashion. In this case, a formal language model, such as a dependence language model or a structural language model, could be useful to combine syntactic phrases with individual terms (Gao et al. 2004; Srikanth and Srihari 2003).

Second, all previous works have not tried to explain different characteristics of phrases for IR when combining individual words and phrases. Sometimes, a phrase should be treated as a much more important retrieval unit than its constituent words. On the contrary, a single word may be more important than the phrase including it. For example, let us consider the query *World Bank Criticism* (TREC6 topic 331) that includes following two phrases: *world bank* and *bank criticism*. When we retrieve relevant documents for this query, the phrase *world bank* in the query should be treated as a more important retrieval unit than its constituent words *world* or *bank*. The reason is that the meaning of the phrase cannot be drawn separately from each constituent word and that each word is expected to occur as a part of the phrase in

relevant documents. However, there is no reason that a document containing *bank criticism* should be treated as a more relevant document than the one containing *bank* and *criticism* separately, because there could be a lot of different expressions of the phrase *bank criticism* in relevant documents.

If a retrieval model does not consider the difference between a phrase and its constituent words, the phrase in a given query and documents may be overemphasized or underrated. For instance, without careful consideration, the use of phrases such as *bank criticism* could even deteriorate the performance because of its very low frequency in the collection; traditional models would systemically boost the score of documents containing the phrase.

In this paper, we investigate usability of syntactic phrases and present a method of integrating them into a ranking formula. Our proposed retrieval model first estimates query likelihood from a normalized syntactic parse tree, and then gives an importance of a phrase over its constituent words by adopting a *variability* value, which is the probability that the syntactic phrase does not occur in a relevant document as the same phrasal form in the given query. With the *variability* value, it is possible to take into account the different characteristics between *world bank* and *bank criticism* by assigning different importance to each syntactic phrase in a document.

2 Syntactic phrase and variability

2.1 Head-modifier pair

In contrast to a statistical phrase, such as a word bigram, a syntactic phrase is a phrase extracted from a sentence based on linguistic knowledge. Among various representations of a syntactic phrase, we have used a head-modifier pair as a syntactic phrase,[1] which is one of the most popular way to capture a syntactic dependency relation from text (Arampatzis et al. 2000; Kraaij and Pohlmann 1998; Strzalkowski et al. 1997; Zhai 1997).

A head-modifier pair is a word pair consisting of a headword and its modifier on a syntactic structure of a natural language sentence.[2] The headword in of the pair is the central element in the phrase, while the modifier is an optional element or an obligatory complement of the headword.[3] Table 1 shows different representations of head-modifier relations.

[1] In IR context, the terminology *syntactic phrase* generally indicates a compound index term extracted by using syntactic relation rather than the strict definition of *phrase* in linguistics, a syntactic structure which has syntactic properties derived from its head. Based on this definition, a head-modifier pair can be regarded as a syntactic phrase.

[2] In this paper, we will denote a head-modifier pair which consists of a modifier w_i and its headword w_{h_i} as $w_i \rightarrow w_{h_i}$.

[3] In terms of linguistics, the terminologies *dependent* and *governor* are more correct expressions than the terminologies *modifier* and *head*. However, because the terminologies *modifier* and *head* have been used more commonly in IR-related works (Arampatzis et al. 2000; Kraaij and Pohlmann 1998; Strzalkowski et al. 1997), we decided to use them.

Table 1 Examples of different representations of head-modifier relations for *information retrieval*

Representations
information retrieval, *retrieval of information*, *retrieving information*, *retrieves information*, ...

Similar to several previous works (Arampatzis et al. 2000; Strzalkowski et al. 1997; Zhai 1997), we have extracted head-modifier pairs from parse trees and normalized them into canonical forms for IR purpose by the following steps:

1. A syntactic tree is converted into a dependency tree. In this step, some ambiguities of long compound noun phrases are resolved by simple heuristics similar to the frequency based method proposed by (Strzalkowski et al. 1994).
2. A head-modifier pair containing a stopword constituent is changed into a new pair by linking its non-stopword constituent to the nearest non-stopword.
3. The two non-stopword constituents in a head-modifier pair are replaced with their stem word by using the Porter stemmer. After this step, different forms of head-modifier pairs which carry same or compatible semantic contents, such as *information retrieval, retrieving information* in Table 1, are normalized into the equivalent form, *inform→retriev*.
4. Finally, normalized head-modifier pairs and single words are extracted from the dependency tree as index units.

Figure 1 shows an example of extracting head-modifier pairs from documents and queries in our retrieval system. Both a head-modifier pair and its individual words are indexed and used to rank documents for the given query based on the new retrieval model that considers variability of each head-modifier pair.

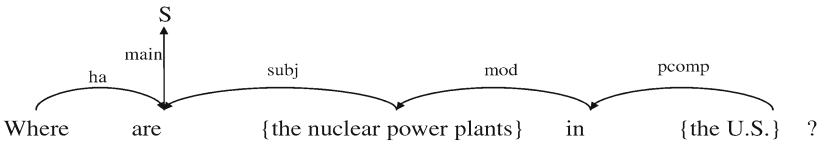
2.2 Variability of syntactic phrase

In an IR system that uses syntactic phrases such as head-modifier pairs, an occurrence of a phrase in a query would be a good evidence for scoring up the relevance of a document containing the phrase. Let us consider the following two queries: *world bank* and *traffic problem*. It is clear that most relevant documents for the query *world bank* should contain the head-modifier pair *world→bank* in the same form. For the query *traffic problem*, some relevant documents may contain *traffic* and *problem* separately although there also can be a considerable amount of documents containing the same head-modifier pair *traffic→problem*. In these cases, it is not difficult to agree that the document containing the head-modifier pair can be regarded as a more relevant document than the document containing the constituent words separately although there is a difference of reliability between two head-modifier pairs.

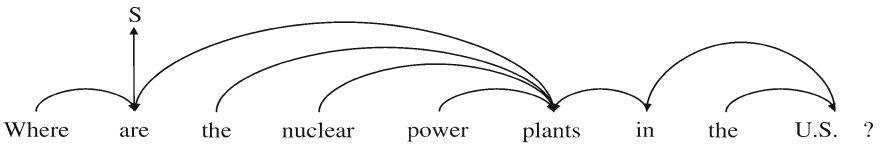
However, some syntactic phrases in a query hardly occur as a same phrasal form but usually do occur as a different form in a document. For example, the query *ferry*

Input text: "Where are the nuclear power plants in the U.S?"

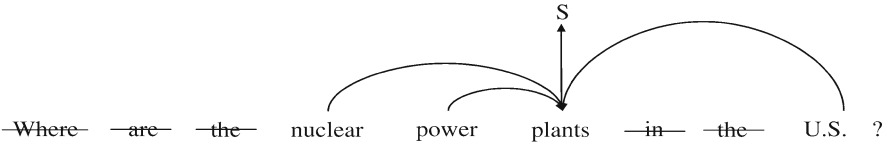
Step (0). The input text is parsed with a Functional Dependency Grammar (FDG) parser



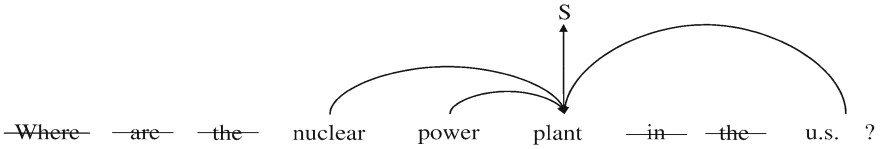
Step (1). The syntactic tree is converted into a simple dependency tree and remained ambiguities in compound noun phrases are resolved



Step (2). Stopwords are removed and the dependency tree is reconstructed



Step (3). Non-stopword constituents are normalized with Porter stemmer



Step (4). Every head-modifier pair and single words in the dependency tree are extracted as index units as follows:

- **Head-modifier pairs**: nuclear→plant, power→plant, u.s.→plant
- **Individual words**: nuclear, power, plant, u.s.

Fig. 1 The example of extracting head-modifier pairs from text. In our retrieval system, the functional dependency grammar (FDG) based parser is used for syntactic parsing and Porter stemmer is used for normalizing morphological variants

sinking can be presented in different forms in a document without any modification of its meaning as follows:

– … few passengers of the *ferry* were survived from the *sinking* …
– Many passengers were asleep when the *ferry* ran aground at 12:43 a.m. The vessel *sank* within an hour …

Although the two words *ferry* and *sink* in both the examples are closely related to each other, they cannot be extracted together as a syntactic phrase such as a head-modifier pair; in these examples, the two words are not syntactically related, but only semantically related. We will refer to such an alternative representation preserving the original meaning of a syntactic phrase in a query as a *variant* of the phrase.[4] Such variants are generated by various linguistic phenomena, such as ellipsis, co-referencing, replacement of near-synonym, etc. In this example, the occurrence of the head-modifier pair *ferri→sink* cannot be a stronger evidence of the relevancy than the occurrences of the constituent words because its meaning can be expressed without the head-modifier relation.

If we could recognize all variants of a phrase in a text and use them in retrieving relevant documents, the different importance of each syntactic phrase may not be a significant matter in an IR system using phrases. However, it is not a feasible solution; it requires too many sophisticated NLP techniques such as reference resolution or word sense disambiguation, and some of them are practically not available. Thus, it can be more effective to differentiate an importance of each phrase according to relationship between a phrase and its constituent words, rather than to try to identify its variants.

In order to reflect such different characteristics, we define variability v_i of a syntactic phrase (or a head-modifier pair) as the probability that a modifier word w_i, which has the dependency relation with its headword w_{h_i} in the query Q, is occurred as a single word without its headword w_{h_i} in the relevant documents R for Q:

$$v_i = p(h = 0 | w_i \in R, w_i \rightarrow w_{h_i} \in Q), \tag{1}$$

where $w_i \rightarrow w_{h_i}$ is the head-modifier pair consisting of a modifier w_i and its headword w_{h_i}, and h is a binary random variable denoting the existence of the word w_{h_i}. Here, $h = 0$ indicates the case that the given modifier w_i occurs in R without its headword w_{h_i}, and $h = 1$ indicates that w_i coocurrs in R with its headword w_{h_i}. This probability measures the possibility that variants for the head-modifier pair in a query are used in relevant documents.

Among the above example queries, the variability of the head-modifier pair *world→bank* must be almost zero because the meaning of *world bank* (the name of a specific bank) must be represented in documents by the same head-modifier pair.

[4]The terminology *variant* defined in this paper does not indicate *morphological variant* (e.g. *retrieving information* and *retrieve information*) or *syntactic variant* (e.g. *information retrieval* and *retrieval of information*) because they are normalized into an equivalent head-modifier pair (e.g. *inform→retriev*) through the normalization process in the Section 2.1. In this paper, the *variants* of a phrase only indicate the alternative expressions which have the same meaning of the phrase.

However, the variability of the pair *ferri→sink* must be very high, because there are many different ways to describe the event of *ferry sinking* as shown in the above example sentences.

Intuitively, the occurrence of a head-modifier pair with very low variability should be regarded as more reliable evidence than the occurrences of the individual constituent words. In contrast, the occurrence of a head-modifier pair with very high variability is not a good evidence for being relevant compared to the case where the constituent words occur individually without any syntactic relation.

We believe that this variability is useful for giving different importance to the occurrence of individual words and head-modifier pairs in calculating the relevance score of the document given a query. In the next section, we describe how to effectively estimate the variability.

3 Predicting variability

By using the TREC topics 151–200 and their corresponding relevant documents in AP and WSJ document set on Tipster Disk 1, we have extracted 1,187 head-modifier pairs from the queries and measured the variability value of each pair. Table 2 shows some head-modifier pairs with their variability values. In the Table, we can notice that some head-modifier pairs which are hardly replaced by other expressions, such as *mutual→fund* and *fast→food*, generally have very low variability. On the contrary, other pairs not having such a strong relationship, for instance, *car→develop* and *investig→hire*(*inverstigator→hiring*'), are highly variable. These examples sufficiently describe the concept of variability defined in the previous section.

Although we could measure the variability values for those head-modifier pairs, it is almost impossible to measure variability for every possible head-modifier pair in an arbitrary query because we never know relevant documents for the query. For this reason, we try to estimate the variability of an arbitrary head-modifier pair by assuming the variability as a feature function and performing regression of the feature function using the possible training data.

Table 2 Examples of variability values of head-modifier pairs in TREC topics 151–200

Head modifier pairs (original forms in queries)	Variability value
Mutual→ fund (mutual→ fund)	0.00
Fast→ food (fast→ food)	0.12
Acid→ rain (acid→ rain)	0.19
Nurs→ home (nursing→ home)	0.31
Public→ investig (public→ investigator)	0.39
Generic→ drug (generic→ drug)	0.52
Public→ school (public→ school)	0.65
Overcrowd→ prison (overcrowded→ prison)	0.92
Food→ restaur (food→ restaurant)	0.94
Investig→ hire (investigator→ hiring)	0.94
Car→ develop (car→ development)	0.97
Orang→ cause (orange→ cause)	0.97
textile→ product (textile→ product)	1.00

Table 3 Features for predicting variability

Feature	Definition & value
Preferred modification distance (PMD)	**Definition:** The most frequent modification distance between a modifier and a headword in a document collection. **Value:** 1,2,3, or long
Preferred phrasal type (PPT)	**Definition:** The most frequent phrasal type (POS tag of a head-word) of a given head-modifier pair in a document collection. **Value:** NP, VP, Others
Uncertainty of modification distance (UMD)	**Definition:** Entropy(H) of modification distance(d) of a given head-modifier pair in a document collection. **Value:** $H(p(d = x \mid w_i \rightarrow w_{h_i}))$ *where* $d \in < 3$ *or long*
Ratio of multiple occurrences (RMO)	**Definition:** Ratio of head-modifier pairs repeatedly used in a document. **Value:** $\frac{\sum_{\forall D; C(w_i \rightarrow w_{h_i}, D) > 2} C(w_i \rightarrow w_{h_i}, D)}{C(w_i \rightarrow w_{h_i}, C)}$ *where,* $C(x, y)$ *means frequency of* x *in* y
Ratio of a single word (RSW)	**Definition:** Ratio of modifier words which do not have the same head-modifier pair in a document for a given head-modifier pair. **Value:** $\frac{\sum_{\forall D; C(w_i \rightarrow w_{h_i}, D) \neq 0} C(w_i, D) + \alpha}{C(w_i \rightarrow w_{h_i}, C) + \beta}$ *where,* α *and* β *are control parameters which can be determined empirically. We use the value of the equation after quantizing to three level:* ≤ 3, > 3 *and* ≤ 9, > 9

All probabilities in this table are estimated by discount smoothing

We have carefully investigated 1,187 head-modifier pairs and their variability values, and we have found that the variability of a head-modifier pair highly depends on the five characteristics of a pair shown in Table 3. For example, extremely low variability head-modifier pairs such as *mutual→fund* usually appear as a noun phrase, and the distance between their constituent words is usually fixed within a short distance. In addition, they usually occur repeatedly in a document and its constituent words rarely occur alone. In contrast, highly variable head-modifier pairs such as *investig→hire* and *car→develop* do not have such tendencies. They are frequently transformed to various phrasal types, for instance, noun phrase to verb phrase, and the modification distance between constituent words is not fixed in

Table 4 Correlation between the probabilities predicted by our model and the probabilities estimated from the relevant documents

Test collection	TR4nl	TR7t
Correlation coefficient	0.7896	0.7319

documents. In cases of such head-modifier pairs, repetition of the pair rarely occurs in a document.

Thus, we first define variability of a head-modifier pair $w_i \rightarrow w_{h_i}$ as a logistic function:

$$\begin{aligned} v_i &= p(h = 0 | w_i \in R, w_i \rightarrow w_{h_i} \in Q) \\ &\approx p(h = 0 | x_i) \\ &= \frac{1}{Z(\lambda_1, \ldots, \lambda_5)} exp[\lambda_1 f_1(x_i, h = 0) + \ldots + \lambda_5 f_5(x_i, h = 0)], \end{aligned} \tag{2}$$

where x_i is the feature vector of the head-modifier pair $w_i \rightarrow w_{h_i}$, f_j indicates the jth feature, and $Z(\lambda_1, \ldots, \lambda_5)$ is a normalization factor.

Table 5 The ten highest and lowest variable head-modifier pairs in TREC4nl

Head modifier pairs	Predicted v	v values in R
Fuel→cell (fuel→cell)	0.27	0.26
Medic→wast (medical→waste)	0.44	0.26
Social→secur (social→security)	0.45	0.01
Nation→park (national→park)	0.50	0.44
Dna→test (dna→testing)	0.50	0.67
Affirm→action (affirmative→action)	0.58	0.02
Blood→pressur (blood→pressure)	0.60	0.38
Infant→mortal (infant→mortality)	0.61	0.52
High→pressur (high→pressure)	0.61	0.42
Rain→forest (rain→forest)	0.64	0.22
...		
Great→emerg (great→emergence)	0.997	1.0
Agenc→function (agency→function)	0.997	1.0
Industri→affect (industry→affect)	0.997	0.96
Materi→garbag (material→garbage)	0.998	1.0
Ensur→propos (ensure→propose)	0.998	1.0
Export→compar (export→compare)	0.998	1.0
Halt→organ (halt→organization)	0.998	1.0
Peopl→pressur (people→pressure)	0.998	1.0
Year→stori (year→story)	0.998	1.0
Agreement→advanc(agreement→advance)	0.998	0.98

Finally, we compute the lambda values by the maximum entropy method using 1,187 training head-modifier pairs.[5]

To examine the effectiveness of our variability prediction method, we have calculated the correlation coefficient between the outputs of our predictor and the variability estimated from relevant documents using TREC topics 201–250 and their corresponding relevant documents on Tipster Disk 2, 3 (TR4nl), and TREC topics 351–400 and their corresponding relevant documents on Tipster Disk 4, 5 (TR7t).

The results are shown in Table 4. Despite of the insufficient size of training data, the variability predicted by our method has a strong correlation with the actual variability estimated in the relevant documents: The coefficient value in TR4nl collection is about 0.79 and 0.73 in TR7t collection. This means that our prediction method can reliably estimate the variability.

Table 5 shows the highest and lowest variable head-modifier pairs in TR4nl predicted by our prediction method. In this table, *Predicted* v means the value estimated by our predictor and v *values in* R is the probability calculated in the relevant documents.

4 A retrieval model reflecting variability of syntactic phrase

4.1 A basic model

In our language modeling approach, a query Q is represented by two elements, S and T, similar to the other structural language modeling approaches used for speech recognition (Chelba et al. 1997; Chelba and Jelinek 1999). S means the word sequence, $S = (w_1, \ldots, w_n)$, and T is the dependency tree consisting of normalized head-modifier pairs, $T = (w_1 \rightarrow w_{h_1}, \ldots, w_n \rightarrow w_{h_n})$. Based on this representation of a query, our retrieval model formulates the query generation in two phases. A tree T is first generated from a document D according to the distribution $p(T|D)$, and a word sequence S is then generated by $p(S|T, D)$.

$$\begin{aligned} p(Q|D) &= p(S, T|D) \\ &= p(S|T, D)p(T|D) \end{aligned} \tag{3}$$

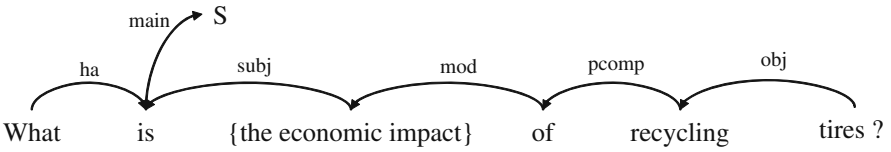
In principle, we should recover $p(Q|D)$ as the marginal summation $\sum_{\forall T_i} P(S, T_i|D)$ but, in practice, this summation is assumed to be dominated by the most probable tree T of Q. The most probable dependency tree T is acquired from one-best result of a syntactic parser and the normalization process mentioned in Section 2.1. One example is shown in Fig. 2.

[5] In order to use the maximum entropy model for the regression problem, we generate training instances according to the variability of each training head-modifier pair.

1. Query

"Where is the economic impact of recycling tires?"

2. Syntactic parsing result



3. Modified tree after normalization

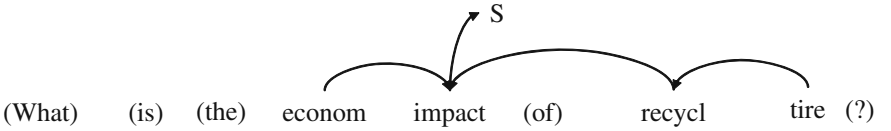


Fig. 2 An example of generating a dependency tree T from a query. The third one is used for the final query representation with its word sequence. S in the figure is the symbol denoting a sentential head.

To make (3) into a tractable one, we have made two independence assumptions. The first one is that the word w_i is only dependent on the head-modifier pair $w_i \rightarrow w_{h_i}$ which includes w_i as a modifier and conditionally independent to the other words.

$$\begin{aligned} p(S|T, D) &= p(w_1, \ldots, w_n|T, D) \\ &= \prod_{i=1}^{n} p(w_i|w_1 \rightarrow w_{h_1}, \ldots, w_n \rightarrow w_{h_n}, D) \\ &\approx \prod_{i=1}^{n} p(w_i|w_i \rightarrow w_{h_i}, D) \end{aligned} \tag{4}$$

The second assumption is that all head-modifier pairs are statistically independent each other. By using this assumption, we get a decomposed probability for $p(T|D)$:

$$\begin{aligned} p(T|D) &= p(w_1 \rightarrow w_{h_1}, \ldots, w_n \rightarrow w_{h_n}|D) \\ &\approx \prod_{i=1}^{n} p(w_i \rightarrow w_{h_i}|D) \end{aligned} \tag{5}$$

From (4) and (5), the probability $p(Q|D)$ can be rewritten as:

$$
\begin{aligned}
p(Q|D) &\approx \prod_{i=1}^{n} p(w_i|w_i \to w_{h_i}, D) \prod_{i=1}^{n} p(w_i \to w_{h_i}|D) \\
&= \prod_{i=1}^{n} p(w_i|w_i \to w_{h_i}, D)\, p(w_i \to w_{h_i}|D) \\
&= \prod_{i=1}^{n} p(w_i, w_i \to w_{h_i}|D) \\
&= \prod_{i=1}^{n} p(w_i|D) \prod_{i=1}^{n} p(w_i \to w_{h_i}|w_i, D) \qquad (6)
\end{aligned}
$$

Each probabilistic term is then smoothed by a collection model using interpolation parameters λ_w and λ_d:

$$
\begin{aligned}
p(Q|D) \approx & \prod_{i=1}^{n} \{\lambda_w p_{ml}(w_i|D) + (1-\lambda_w) p_{ml}(w_i|C)\} \\
& \times \prod_{i=1}^{n} \left\{ \begin{array}{l} \lambda_d p_{ml}(w_i \to w_{h_i}|w_i, D) \\ +(1-\lambda_d) p_{ml}(w_i \to w_{h_i}|w_i, C) \end{array} \right\}, \qquad (7)
\end{aligned}
$$

where λ_w is defined as $\frac{|D|}{|D|+\mu}$ using a document length $|D|$ and the parameter μ of the Dirichlet distribution, and λ_d is a fixed constant between 0 and 1. All probabilities are estimated by MLE in the equation. We call it as Basic Structural Language Model (BSLM).

4.2 Incorporating variability

The BSLM itself can play a role as a ranking formula. It is not only a tractable retrieval model but a model that can integrate a head-modifier pair into the structural language modeling framework in a well-established manner.

However, there still remains an unsolved problem. In the BSLM, the parameter λ_d is a factor controlling importance of head-modifier pairs over their constituent words. The higher λ_d is, (7) is closer to $p(w_i \to w_{h_i}|D)$ and ignores the unigram probability $p(w_i|D)$. In contrast, when λ_d is 0, the value of (7) is dominated by $p(w_i \to w_{h_i}|w_i, C) \cdot p(w_i|D)$, and the result of document ranking is the same as the unigram language model based on Dirichlet smoothing. This means that the importance of all head-modifier pairs over their constituent words is assumed to be uniform in BSLM with the fixed λ_d. Obviously, it is more desirable that λ_d is differentiated according to the importance of each head-modifier pair.

Thus, we replace the fixed parameter λ_d by $1.0 - variability$ of a head-modifier pair as follows:

$$p(Q|D) \approx \prod_{i=1}^{n} \{\lambda_w p_{ml}(w_i|D) + (1-\lambda_w) p_{ml}(w_i|C)\} \times \prod_{i=1}^{n} \left\{ \begin{array}{l} (1-v_i) \cdot p_{ml}(w_i \rightarrow w_{h_i}|w_i, D) \\ +v_i \cdot p_{ml}(w_i \rightarrow w_{h_i}|w_i, C) \end{array} \right\}, \quad (8)$$

where v_i is the value estimated for the head-modifier pair $w_i \rightarrow w_{h_i}$ by our variability prediction method.

In this equation, the impact of each head-modifier pair is differentiated by its variability. A head-modifier pair with very low value of v_i, such as *world→bank* or *mutual→fund*, is treated as a single term, because the equation is dominated by the probability generating the head-modifier pair from a document, such as $p(world \rightarrow bank|D)$ or $p(mutual \rightarrow fund|D)$. In these cases, the unigram probability of each constituent word, *world* or *mutual*, is almost completely ignored in document ranking. On the contrary, the relevance score of the document is not affected by the occurrences of a head-modifier pair with very high v_i such as *ferri→sinking* or *bank→crtic* in a document. In this case, the results of ranking become very similar to the results of the unigram language model.

From now on, we call the retrieval model based on (8) as Variability incorporated SLM (VSLM).

5 Experiments

5.1 Experimental setup

5.1.1 Data for evaluating retrieval models

For measuring the efficiency of two proposed retrieval models, SLM and VSLM, we have carried out a large-scale evaluation using several TREC test collections. As well known, the performance of a retrieval model can be varied according to the characteristics of the test collections. Therefore we designed experiments with various configurations on different types of query sets and different size of document collections.

The experiments were conducted using two query sets: TREC topics 201–250 (description field only) and TREC topics 351–400 (title field only). Each query set has a different nature of queries. TREC topics 201–250 are closer to "natural language" style queries. They mostly consist of one complete natural language sentence of which the length is about 10 to 15 words including stopwords, whereas TREC topics 351–400 are typical short queries consisting of 2 or 3 words.

For evaluating the retrieval performance according to varying size of document collections, we also used the following several collections from TREC for each query set: For TREC topics 201–250, (1) all documents in Tipster disk 2 and disk 3(TREC4nl), (2) Association Press section on disk 2 and disk 3(AP4nl), (3) Wall street journal section on disk 2 (WSJ4nl), (4) ZIFF section on disk 2 and 3 (ZIFF4nl) are used, and for TREC topics 351–400, (1) all documents in Tipster disk 4 and Disk

5 minus CR (TREC7t), (2) Financial Times section on disk 4 (FT7t), (3) Los Angeles Times on disk 5 (LA7t), (4) FBIS on disk 5 (FBIS7t) are used for the evaluations.

All of the above query sets and document collections are completely different from the queries and documents used for training the variability predictor in order to prevent any influence on experimental results.

5.1.2 Linguistic analysis and indexation

For indexing documents, we have extracted head-modifier pairs and stem words from documents by the method described in the Section 2.1 with the following linguistic components: Connexor FDG parser,[6] the syntactic parser based on the functional dependency grammar (Tapanainen and Jarvinen 1997), was used for parsing queries and documents, and Porter stemmer (Porter 1997)was used for stemming. In this step, stopwords were removed using a list of 365 stopwords, and the head-modifier pairs occurred less than 3 times in the document collections are also removed for maintaining indexes as a manageable size.

Head-modifier pairs and stem words extracted from documents were separately indexed in our system. We used the traditional inverted indexing architecture consisting of posting files and dictionaries for both the stem words and the head-modifier pairs, but there were several differences: The identifier for a head-modifier is generated by combining the identifiers of a headword and a modifier in the stem word index, and the dictionary for head-modifier pairs includes additional information for the variability prediction, such as a preferred modification distance of a pair.

In the retrieval phase, our retrieval system extracted head-modifier pairs and stem words from a query in the same way that a document was analyzed, and then the system matched the query and documents based on the pairs and the words: The head-modifier pairs were used to match dependencies from the query with relations in documents, and the stem words were used to match the single words of the query with words in documents. In this step, the variability values of the head-modifier pairs were also calculated by using the prediction model described in the Section 3 and the information of head-modifier pairs in the dictionary. Finally, our system ranked all retrieved documents for the query with a retrieval model.

5.1.3 Other configurations

For comparison, we have also implemented the unigram language model with Dirichlet smoothing (UM; Zhai and Lafferty 2001) which is known to perform well in the ad-hoc retrieval task, and have used it as our baseline representing a BOW-based retrieval model.

Parameters for our models and UM were either decided empirically or selected based on the experiments reported in (Zhai and Lafferty 2001). The interpolation parameter λ_d in our SLM model is fixed as 0.05, where the optimal or near-optimal

[6] Connexor FDG parser is a commercial product which is not publicly available, but one can examine the parser at http://www.connexor.com/demo/. Figure 1 in the Section 2.1 shows one parsing example of Connexor FDG parser.

retrieval performances are shown across all test collections.[7] Dirichlet smoothing parameter μ in the unigram probability is set to 2,000.

The performances of retrieval models are measured by several evaluation metrics. To evaluate overall performances, we have used the non-interpolated average precision (AvgPr).

5.2 Experimental results

5.2.1 UM, BSLM vs. VSLM

Tables 6, 7 and 8 show the experimental results of the baseline retrieval model UM and two structural language models BSLM and VSLM.[8] All performances are evaluated with statistical significance test. An * and a ** indicate the statistical significance where p value < 0.05 and p value < 0.02, respectively. In the

Table 6 Comparison of retrieval performances of UM, BLSM and VLSM in natural language style queries

Coll(#Queries)	UM(baseline)		BSLM		VSLM	
	AvgPR	%chg	AvgPR	%chg	AvgPR	%chg
TR4nl(50)	0.1926	–	0.1992	+3.43	0.2242*	+16.41
AP4nl(50)	0.2606	–	0.2707	+3.88	0.2961*	+13.62
WSJ4nl(45)	0.2193	–	0.2430	+10.81	0.2607**	+18.88
ZIFF4nl(32)	0.1710	–	0.1862	+8.89	0.2285	+33.63
Average	0.2109	–	0.2248	+6.75	0.2524	+20.63

Table 7 Comparison of retrieval performances of UM, BLSM and VLSM in short title queries

Coll(#Queries)	UM(baseline)		BSLM		VSLM	
	AvgPR	%chg	AvgPR	%chg	AvgPR	%chg
TR7t(50)	0.1883	–	0.1987	+5.52	0.2026	+7.59
FT7t(49)	0.2388	–	0.2539	+6.32	0.2592	+8.54
LA7t(50)	0.2138	–	0.2367	+10.71	0.2398	+12.16
FBIS4t(38)	0.1909	–	0.1984	+3.93	0.2019	+5.76
Average	0.2080	–	0.2219	+6.62	0.2259	+8.52

[7]Because we focused on investigating the effect of variability in our experiments, we tuned the parameter λ_d of BSLM on the test collections.

[8]For the evaluation, we carefully examined the performance of the baseline model by comparing several researches using the same Dirichlet unigram language model as a baseline model (Srikanth and Srihari 2003; Zhai and Lafferty 2001). In the all query sets and document collections used in our evaluation, the performances of our baseline are very similar to the performances of the baseline in the previous works and the difference is not significant.

Table 8 Comparison of retrieval performances of UM, BLSM and VLSM in *partial* sets of short title queries

Coll(#Queries)	UM(baseline)		BSLM		VSLM	
	AvgPR	%chg	AvgPR	%chg	AvgPR	%chg
TR7t-*p*(36)	0.1833	–	0.1979	+7.97	0.2032	+10.86
FT7t-*p*(29)	0.2731	–	0.2982	+9.19	0.3069	+12.38
LA7t-*p*(30)	0.1993	–	0.2375	+19.17	0.2428*	+21.83
FBIS4t-*p*(20)	0.2073	–	0.2216	+6.90	0.2282	+10.08
Average	0.2158	–	0.2388	+10.81	0.2453	+13.79

Table 8, *partial* and *-p* represent the subset of short queries that include more than one head-modifier pair, and the pairs appeared in the query must occur at least once in the document collection. The basic structural language model BSLM has significantly outperformed UM across all query sets and document collections, and VSLM, which uses the variability value, has improved the performance of BSLM considerably. BSLM has achieved 6.75 and 6.62% improvements over the performance of UM in the natural language query set and the short query set respectively, while VSLM has achieved 20.63 and 8.52% improvements in each query set.

Although there are significant improvements over the performance of UM by BSLM and VLSM, there is an important difference between the behaviors of two models. While VLSM performs much better for natural language queries in terms of the degree of improvement, BSLM does not show any difference between the natural language queries and the short queries. It is clear that there is obviously more chance to improve the UM with long natural language queries compared to short queries with only two or three words, and the retrieval model using syntactic phrases, such as head-modifier pairs, has more merit where natural language queries are used. For this reason, we think that the variability used in VLSM is quite effective to make the retrieval model using head-modifier pairs perform properly.

Table 9 presents the comparison between the number of queries where one model outperforms the other models in the average precision. As shown in the table, there are many queries where SLM shows worse performance than UM. On the contrary, the number of queries where VSLM shows worse performance than UM is relatively small. This table also shows the usefulness of reflecting variability of head-modifier pairs into a retrieval model.

Table 9 Comparison between UM(U), BSLM(B) and VSLM(V)

	U > B	U < B	U > V	U < V	B > V	B < V
TR4nl	26	23	21	28	12	37
TR7t-*p*	17	19	16	20	9	27

The numbers in tables mean the number of the queries matched the case

5.2.2 Behavior of BSLM vs VSLM

Figure 3 shows the examples of the performance variation by BSLM according to λ_d. As λ_d gets closer to 0, the ranking formula of BSLM suggested in (7) becomes similar to UM.

As shown in the table, the optimal λ_d for BSLM differs from query to query. For the query *orphan drug* and *hydrogen energy*, the optimal λ_d is 0.5 and 0.01 respectively. This means that the occurrence of the head-modifier pair *hydrogen→energy* in a document should be regarded as a relatively less important evidence while the occurrence of 'orphan→drug' should be treated as an important evidence. Since the optimal λ_d differs from query to query, BSLM with fixed parameter λ_d performs poorly compared to VSLM. With our VSLM model, $1.0 - v_{orphan \rightarrow drug}$ and $1.0 - v_{hydrogen \rightarrow energy}$ are 0.491 and 0.036, respectively,[9] and these values make VSLM to perform better than BSLM with the fixed λ_d of 0.05.

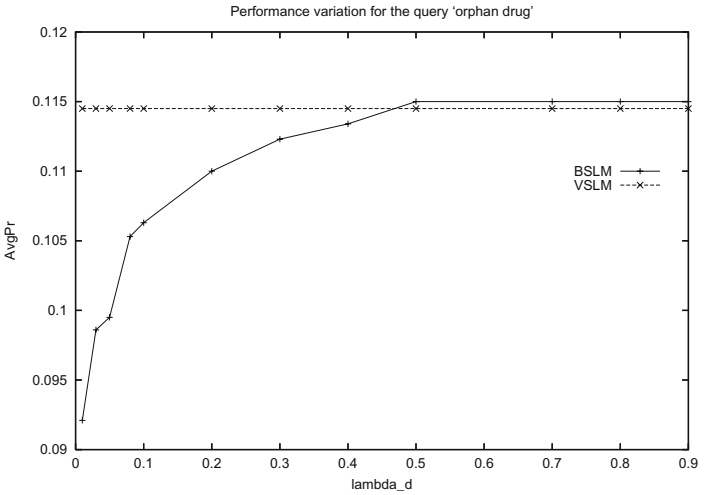
When the several head-modifier pairs are extracted from a query, the behavior of BSLM becomes more complicated. For the query *drug legalization benefits*, two head-modifier pairs *drug→ legalization* and *legalization→benefit* are extracted from the query. In this case, we can get only small benefit even if we find optimal λ_d for the query because the λ_d is just a compromised value between the two different good λ_d values. On the other hand, VSLM significantly outperforms BSLM by giving different variability values to each pair. This example shows how the differentiation of each head-modifier pair in a query is important for the retrieval effectiveness.

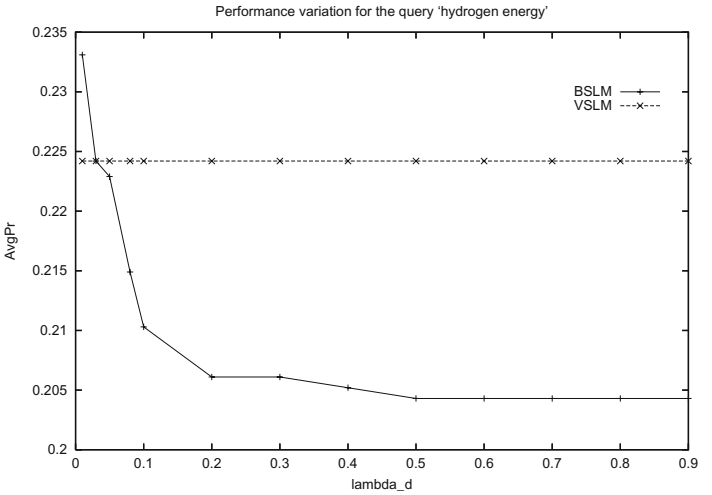
5.2.3 BSLM and VSLM at top-ranked level

We have investigated how BSLM and VSLM behave at top-ranked level. In the experiments of Mitra et al's work (Mitra et al. 1997), a phrase usually does not improve the precision at high ranks. They have argued that the use of phrases tends to over-emphasize only one aspect of the intention of multi-word queries consisting of multi-aspects, and it causes the retrieval model using phrases to perform poorly. As a consequence, they have concluded that phrases, such as head-modifier pairs in our approach, cannot consistently enhance the precision at high ranks.

Similar results are also observed in our experimental results with BSLM as shown in Table 10. In the table, *partial* indicates the same subset of short queries in Table 8, and P@10 and p@20 indicate the precision at 10 documents and the precision at 20 documents, respectively. As shown Table 10, the improvements are only moderate and inconsistent across the test collections when we use BSLM as a retrieval model. However, when considering variability in the retrieval model, the experimental results are significantly different from the results of BSLM and the previous research (Mitra et al. 1997). As shown in Table 11, the performances in p@10d and p@20d with VSLM are greatly improved in most cases, which show that using variability is also very effective at top-ranked level. This result explains that the failure at top ranked documents in the previous works does not imply the inadequateness of phrases or dependencies between words, and it may be caused by the inadequateness of the retrieval model used to combine phrases and words.

[9] In VSLM model, λ_d is determined by $1.0 - v_i$ for the head-modifier pair $w_i \rightarrow w_{h_i}$.


Performance variation for the query 'orphan drug'
AvgPr
lambda_d
BSLM
VSLM
0.12
0.115
0.11
0.105
0.1
0.095
0.09
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9


Performance variation for the query 'hydrogen energy'
AvgPr
lambda_d
BSLM
VSLM
0.235
0.23
0.225
0.22
0.215
0.21
0.205
0.2
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9

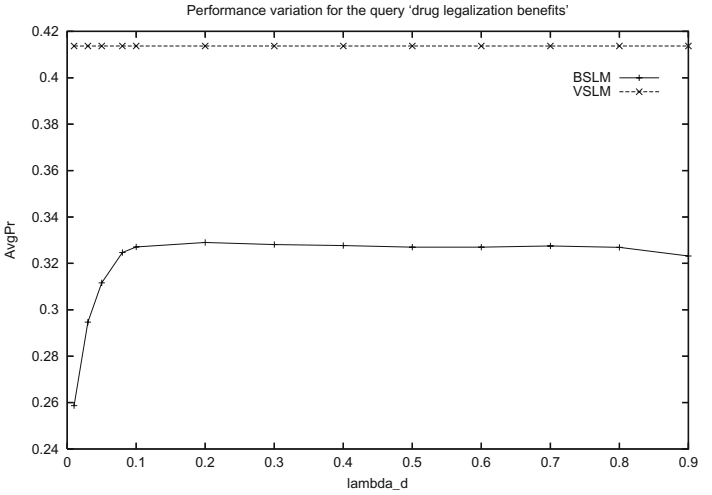

Performance variation for the query 'drug legalization benefits'
AvgPr
lambda_d
BSLM
VSLM
0.42
0.4
0.38
0.36
0.34
0.32
0.3
0.28
0.26
0.24
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9

Fig. 3 Performance variation for the query *orphan drug*, *hydrogen energy* and *drug legalization benefits* according to λ_d in BSLM

5.2.4 Error analysis

Although our proposed VSLM with variability has outperformed UM and BSLM considerably, there are several cases where VSLM performs poorly compared to UM and BSLM.

First, the head-modifier pair from a given query sometimes occurs in a non-relevant document by chance. In this case, the performance of VSLM is slightly worse than UM, because VSLM boosts the relevance score of the non-relevant document containing the pair. This problem is common in all retrieval models using syntactic phrases, but VSLM would be more serious if the variability of the head-modifier pair is quite low. For example, the head-modifier pair *altern→medicin* in the query *Alternative Medicine* (TREC7 topic 381) is a collocation, so its variability value estimated by our prediction model is very low. However, the relevant documents for the query do not contain such the head-modifier pair: Instead of *alternative medicine*, different expressions such as *herbal medicine* or *traditional chinese medicine* are used in the relevant documents. In this case, the retrieval performance of VSLM is lower than the performance of UM or BSLM because the documents containing the pair are strongly preferred in VSLM. The same problem is also found in the query *Native American Casino* (TREC7 topic 372). The head-modifier pair *nativ→american*, extracted from the collocation *native american*, also has a low variability value but it is not helpful to retrieve relevant documents: Most relevant documents for the query do not contain *native american*, but only contain its synonym *indian*. In such cases, our VSLM model usually fails to yield a better result than UM or BSLM.

Second, VSLM may deteriorate the performance when it gives different variability values to two head-modifier pairs which should be equally treated. For example, let us consider the query *Food/Drug Law* (TREC7 topic 370). In this query, two head-modifier pairs *food→ law* and *drug→law* are extracted and assigned by its variability.

Table 10 Comparison of precision between UM and BSLM at high ranks

Colls	UM(baseline)		BSLM	
	p@10d	p@20d	p@10d(%Δ)	p@20d(%Δ)
TR4nl	0.4060	0.3460	0.4200(+3.4)	0.3560(+2.8)
AP4nl	0.3633	0.3041	0.3796(+4.5)	0.3143(+3.4)
WSJ4nl	0.2844	0.2333	0.3378(+18.8)	0.2656(+13.8)
ZIFF4nl	0.1312	0.0969	0.1281(−2.4)	0.0937(−3.3)
Average	0.2962	0.2451	0.3164(+6.09)	0.2574(+4.20)
TR7t(*partial*)	0.4167	0.3542	0.4417(+6.00)	0.3931(+10.98)
FT7t(*partial*)	0.3345	0.2448	0.3276(−2.06)	0.2362(−3.51)
LA7t(*partial*)	0.2967	0.2300	0.3233(+8.97)	0.2600(+13.04)
FBIS7t(*partial*)	0.3150	0.2450	0.3550(+12.70)	0.2700(+10.20)
Average	0.3407	0.2685	0.3619(+6.40)	0.2898(+7.68)

Table 11 Comparison of precision between BM and VSLM at high ranks

Colls	UM(baseline)		VSLM	
	p@10d	p@20d	p@10d(%Δ)	p@20d(%Δ)
TR4nl	0.4060	0.3460	**0.4460(+9.9)**	**0.3800(+9.8)**
AP4nl	0.3633	0.3041	**0.4102(+12.9)**	**0.3388(+11.4)**
WSJ4nl	0.2844	0.2333	0.3244(+14.1)	**0.2756(+18.1)**
ZIFF4nl	0.1312	0.0969	**0.1375(+4.8)**	**0.1000(+3.2)**
Average	0.2962	0.2451	**0.3295(+10.41)**	**0.2736(+10.64)**
TR7t(*partial*)	0.4167	0.3542	**0.4778(+10.86)**	**0.4208(+18.80)**
FT7t(*partial*)	0.3345	0.2448	**0.3621(+8.25)**	**0.2466(+0.74)**
LA7t(*partial*)	0.2967	0.2300	**0.3533(+19.08)**	**0.2750(+19.57)**
FBIS7t(*partial*)	0.3150	0.2450	**0.4050(+28.57)**	**0.2900(+18.37)**
Average	0.3407	0.2685	**0.3996(+17.64)**	**0.3081(+14.37)**

Bold values denote the cases that VSLM performs better than both of UM and BSLM

Although there is no reason to differentiate those two pairs, our proposed model prefers documents containing one of the pairs with lower variability.

Also, we found some cases that the performance of our retrieval system has been decreased by improper normalizations of Porter stemmer. For example, the word *organic* in the query *organic soil enhancement* (TREC7 topic 388) should not be matched with the word *organization* or *organ* in documents, but Porter stemmer normalizes them into the equivalent form *organ*. Although it is a very common problem in IR systems using a stemmer, the problem can be more serious in our approach because such an improper normalization can also yield a wrong head-modifier pair normalization: By Porter stemmer, the text segments *organic enhancement*, *organization enhancement*, *organ enhancement* are normalized into the same head-modifier pair *organ→enhanc* despite of the differences in their meaning. It can cause a mismatching problem between a query containing *organic enhancement* and a non-relevant document containing *organization enhancement*.

However, in our experiments, we have found only a few queries where such a mismatching problem of head-modifier pairs has occurred. Even in the queries having the words normalized improperly, the improper stemming has not significantly influenced effectiveness of head-modifier pairs in retrieval. It seems to be because a syntactic phrase such as a head-modifier pair has enough ability to resolve the ambiguity of a stemmed word. For example, the word *insurance* in the query *health insurance* can be matched with the word *insurable* because they share the same stem *insur*. Obviously, it is not desirable. However, such ambiguity of the stem word *insur* may not have influence on effectiveness of the head-modifier pair *health→insur* because a syntactic dependency between *health* and *insurable*, such as *health insurable*, is rarely occurred in documents.

6 Conclusions

In this paper, we have investigated how to combine syntactic head-modifier pairs and individual words in a retrieval model by considering different relative importance of head-modifier pairs over their constituent words. We have introduced *variability*,

the probability that a head-modifier pair in a query is used in relevant documents not in a same phrasal form to express the meaning of the head-modifier pair. We have devised a maximum entropy model based method for predicting variability of head-modifier pairs. We have tested our predicting method and obtained the result that the correlation between estimated variability values and calculated values in the given relevant document set was about 0.75.

We have introduced a new model using the variability, named Variability incorporated Structural Language Model(VSLM), and evaluated the model using various query sets and test collections. Through various experiments, we could obtain about 21% improvement for the natural language style queries and about 9% improvement for the short queries compared to the traditional BOW based retrieval model. Specifically, our proposed model using the variability has shown a considerable improvement in precision at high rank. For both the short query set and natural language style query set, our variability based model have achieved more than 10% improvement in p@10 and p@20 compared to the BOW based model. These experimental results imply that our approach can be more effective in the application area where high precision is required or long natural language queries are used as an input, for example, QA-related IR.

For future work, we plan to investigate the effectiveness of variability with various retrieval models and various types of phrases, for instance, a statistical phrase such as a bigram.

References

Arampatzis, A., van der Weide, T., Koster, C., & van Bommel, P. (2000). Linguistically-motivated information retrieval. In *Encyclopedia of Library and Information Science*. New York: Marcel Dekker.

Brants, T.: (2004). Natural language processing in information retrieval. In *Proceedings of CLIN 2004* (pp. 1–13). Antwerp, Belgium.

Chelba, C., Engle, D., Jelinek, F., Jimenez, V. M., Khudanpur, S., Mangu, L., et al. (1997). Structure and performance of a dependency language model. In *Proceedings of Eurospeech '97* (pp. 2775–2778). Rhodes, Greece.

Chelba, C., & Jelinek, F. (1999). Recognition performance of a structured language model. In *Proceedings of Eurospeech '99* (pp. 1567–1570).

Fagan, J. (1987). Automatic phrase indexing for document retrieval. In *Proceedings of SIGIR '87* (pp. 91–101).

Gao, J., Nie, J.-Y., Wu, G., & Cao, G. (2004). Dependence language model for information retrieval. In *Proceedings of SIGIR '04* (pp. 170–177).

Kraaij, W., & Pohlmann, R. (1998). Comparing the effect of syntactic vs. statistical phrase indexing strategies for dutch. In *ECDL '98: Proceedings of the Second European Conference on Research and Advanced Technology for Digital Libraries* (pp. 605–617). London, UK.

Metzler, D., & Croft, W. B. (2005). A Markov random field model for term dependencies. In *Proceedings of SIGIR '05* (pp. 472–479).

Miller, D. R. H., Leek, T., & Schwartz, R. M. (1999). A hidden Markov model information retrieval system. In *Proceedings of SIGIR '99* (pp. 222–229).

Mitra, M., Buckley, C., Singhal, A., & Cardie, C. (1997). An analysis of statistical and syntactic phrases. In *Proceedings of RIAO* (pp. 200–214).

Pohlmann, R., & Kraaij, W. (1997). The effect of syntactic phrase indexing on retrieval performance for Dutch texts. In *Proceedings of RIAO'97* (pp. 176–187).

Porter, M. F. (1997). *An algorithm for suffix stripping* (pp. 313–316). San Francisco, CA, USA: Morgan Kaufmann.

Song, F., & Croft, W. B. (1999). A general language model for information retrieval. In *Proceedings of CIKM '99* (pp. 316–321).

Srikanth, M., & Srihari, R. (2003). Exploiting syntactic structure of queries in a language modeling approach to IR. In *Proceedings of CIKM '03* (pp. 476–483).

Strzalkowski, T., Carballo, J.P., & Marinescu, M. (1994). Natural language information retrieval: TREC-3 report. In *The Third Text REtrieval Conference* (*TREC 3*).

Strzalkowski, T., Guthrie, L., Karlgren, J., Leistensnider, J., Lin, F., Perez-Carballo, J., et al. (1997). Natural Language information retrieval: TREC-5 report. In *The Fifth Text REtrieval Conference* (*TREC 5*) (pp. 291–313).

Tapanainen, P., & Jarvinen, T. (1997). A non-projective dependency parser. In *Fifth Conference on Applied Natural Language Processing* (pp. 64–71).

Zhai, C. (1997). Fast statistical parsing of noun phrases for document indexing. In *Proceedings of the fifth conference on Applied natural language processing* (pp. 312–319).

Zhai, C., & Lafferty, J. (2001). A study of smoothing methods for language models applied to Ad Hoc information retrieval. In *Proceedings of SIGIR-01* (pp. 334–342).