

# Topic Signature Language Models for Ad Hoc Retrieval

Xiaohua Zhou, *Student Member, IEEE*, Xiaohua Hu, *Member, IEEE*, and Xiaodan Zhang, *Student Member, IEEE*

**Abstract**—Semantic smoothing, which incorporates synonym and sense information into the language models, is effective and potentially significant to improve retrieval performance. Previously implemented semantic smoothing models such as the translation model have shown good experimental results. However, these models are unable to incorporate contextual information. To overcome this limitation, we propose a novel context-sensitive semantic smoothing method that decomposes a document into a set of weighted context-sensitive topic signatures and then maps those topic signatures into query terms. The language model with such a context-sensitive semantic smoothing is referred to as the topic signature language model. In detail, we implement two types of topic signatures, depending on whether ontology exists in the application domain. One is the ontology-based concept and the other is the multiword phrase. The mapping probabilities from each topic signature to individual terms are estimated through the EM algorithm. Document models based on topic signature mapping are then derived. The new smoothing method is evaluated on the TREC 2004/2005 Genomics Track with ontology-based concepts, as well as the TREC Ad Hoc Track (Disks 1, 2, and 3) with multiword phrases. Both experiments show significant improvements over the two-stage language model, as well as the language model with context-insensitive semantic smoothing.

**Index Terms**—Information retrieval, language model, semantic smoothing, topic signature, concept, multiword phrase.

## 1 INTRODUCTION

THE language modeling (LM) approach to information retrieval (IR), initially proposed by Ponte and Croft [21], has been popular with the IR community in recent years due to its solid theoretical foundation and promising empirical retrieval performance. In essence, this approach centers on the document model estimation and the query generative likelihood calculation according to the estimated model. However, it is challenging to estimate an accurate document model due to the sparsity of training data. On one hand, because the query terms may not appear in the document, we need to assign a reasonable nonzero probability to the unseen terms. On the other hand, we need to adjust the probability of the seen terms to remove the effect of the background collection model or even irrelevant noise. Thus, the core of the LM approach to IR is to “smooth” document models. Zhai and Lafferty [26], [28] propose several effective background smoothing techniques that interpolate the document model with the background collection model.

A potentially more significant and effective method is semantic smoothing that incorporates synonym and sense information into the language model [15]. Berger and Lafferty [2] incorporate a kind of semantic smoothing into the language model by statistically mapping document terms onto query terms using a translation model trained from synthetic document-query pairs. However, the trans-

lation model is context insensitive (that is, it is unable to incorporate sense and contextual information into the language model) and, therefore, the resulting translation may be mixed and fairly general. For example, the term “mouse” without context may be translated to both “computer” and “cat” with high probabilities. Jin et al. [14] and Cao et al. [4] present two other ways to train the translation probabilities between individual terms, but their approaches still suffer from the same context insensitivity problem as [2]. Thus, it is urgent to develop a framework to semantically smooth document models within the LM retrieval framework.

In this paper, we propose a novel context-sensitive semantic smoothing (CSSS) method based on topic decomposition. A document is decomposed into a set of weighted topic signatures and, then, those topic signatures are mapped into individual terms for the purpose of document expansions. We define a topic signature as either an ontology-based concept or an automated multiword phrase. Because a concept or a multiword phrase itself contains contextual information and its meaning is usually unambiguous, the mapping from topic signatures to individual terms should have higher accuracy and result in better retrieval performance as compared to the semantic translations between single words. For example, “mouse” in conjunction with “computer” could be a topic signature, and the signature might be translated to “keyboard” with a high probability but to “cat” with a low probability due to additional contextual constraints.

We develop an ontology-based algorithm to extract concept-based topic signatures and adopt an existing algorithm referred to as Xtract [23] to identify phrase-based topic signatures. Furthermore, we develop an expectation-maximization (EM)-based algorithm to estimate probabilities of mapping each topic signature into individual terms

- The authors are with the College of Information Science and Technology, Drexel University, 3141 Chestnut Street, Philadelphia, PA 19104. E-mail: xiaohua.zhou@drexel.edu, {thu, xzhang}@cis.drexel.edu.

Manuscript received 18 Oct. 2006; revised 16 Mar. 2007; accepted 9 Apr. 2007; published online 1 May 2007.

For information on obtaining reprints of this article, please send e-mail to: tkde@computer.org, and reference IEEECS Log Number TKDE-0481-1006. Digital Object Identifier no. 10.1109/TKDE.2007.1058.

in the vocabulary. The new smoothing method is tested on collections from two different domains in order to show its robustness. The extraction of concepts needs domain ontology. Thus, we evaluate the effectiveness of concepts on the TREC 2004/2005 Genomics Track. The extraction of multiword phrases does not need any external human knowledge and can be applied to any public domains. Therefore, we test the effectiveness of multiword phrases on TREC Disks 1, 2, and 3, which contain news articles from several sources, including the Associated Press (AP), J.M. Smucker (SJM), and Wall Street Journal (WSJ). The experimental results show that significant improvements are obtained over the two-stage language model (TSLM) [28] and the language model with context-insensitive semantic smoothing (CISS).

The contribution of this paper is threefold. First, it proposes a new document representation using a set of weighted terms and topic signatures. The new scheme also explores the relationship between individual terms and more complicated topic signatures. Second, it develops an EM-based algorithm to estimate the semantic relationships between topic signatures and individual terms and further uses those semantic relationships to smooth the document model, which is referred to as CSSS in this paper. The smoothed document models can be used not only for text retrieval but also for many other text mining applications such as text categorization. Third, it empirically proves the effectiveness of the CSSS for LM IR.

The remainder of this paper is organized as follows: In Section 2, we review previous work related to topic signatures. In Section 3, we first formally define topic signatures and present the approaches to the topic signature extraction and, then, we describe in detail the method of the CSSS. Section 4 shows the experimental results on TREC 2004/2005 Genomics Track collections, where topic signatures are implemented as ontology-based concepts. Section 5 shows the experimental results on TREC Disks 1, 2, and 3, where multiword phrases are used as topic signatures. Section 6 concludes our paper.

## 2 RELATED WORK

The idea of topic decomposition and translation for LM IR is not new. It was used for query expansion as well as document expansion in literature. Song and Bruza adopted information flow (IF) for query expansion in [24]. The context of a concept is represented by a hyperspace analog to language (HAL) vector and the degree of one concept inferring another can then be computed through vector operators. Song and Bruza [24] also invented a heuristic approach to combine multiple concepts, which enabled information inference from a group of concepts (premises) to one individual concept (conclusion). Thus, their query expansion technique was somehow context sensitive. However, it was difficult to be extended to document model expansions. Besides, the degree to which one individual concept could be inferred from another combined concept was not theoretically motivated: Its robustness needs to be further validated.

Similarly, Bai et al. [1] used significant term pairs to expand query models. The combination of two terms is helpful to disambiguate their context and thus can capture

more sense of the query. The expanded query model based on significant term pairs looked like the following:

$$p(w|Q) = (1 - \lambda) \sum_{q_i, q_j \in Q} p_R(w|q_i q_j) p(q_i q_j | Q) + \lambda p_{ML}(w|Q). \quad (1)$$

Here, the second term is a unigram query model for smoothing purposes and the first term (query expansion) is based on topic decomposition and translation. The topic decomposition term  $p(q_i q_j | Q)$  is simply assumed to be uniformly distributed. The topic translation term  $p_R(w|q_i q_j)$  is estimated based on term co-occurrence statistics. The coefficient  $\lambda$  controls the influence of the expansion component. Like the IF approach, this approach is also inappropriate for document model expansions because the distribution of term pairs in a document is obviously not uniform. Besides, the co-occurrence-based estimation algorithm tends to assign higher probability values to general terms than specific terms.

Berger and Lafferty proposed the statistical translation model for the first time in [2]. With this model, a term in a document is statistically mapped to query terms, described as follows:

$$p(q|d) = \sum_w t(q|w)l(w|d), \quad (2)$$

where  $t(q|w)$  is the translation probability from document term  $w$  to query term  $q$ , and  $l(w|d)$  is the unigram document model. The translation model achieved significant improvement over the simple language model on two TREC collections [2]. However, the model only captures the semantic relationship between individual words and is unable to incorporate the contextual information into the translation procedure. In addition, the training of translation probability requires a large number of real query-document pairs, which are very difficult to obtain. For this reason, Berger and Lafferty used synthetic data in the experiment. Besides, a document often contains a considerable number of unique terms and, thus, the model expansion through document and query term mapping is computationally intensive.

The cluster language model [16] may be the first trial of topic decomposition and translation for document model expansions. Liu and Croft [16] incorporated cluster information into document model estimation:

$$p(w|d) = \frac{N_d}{N_d + \mu} p_{ML}(w|d) + \left(1 - \frac{N_d}{N_d + \mu}\right) p(w|cluster), \quad (3)$$

where  $N_d$  is the length of the document and  $\mu$  is a parameter for smoothing. The document clusters are very similar to our topic signatures in the sense that both use a set of documents with similar context rather than a single document to estimate a more accurate topic model. However, in their cluster model, a document is associated with a single cluster, which may become problematic for especially long documents, whereas a document can have multiple topic signatures in our model. Furthermore, the clustering for a large collection is extremely inefficient. Last, lots of decisions need to be made empirically for clustering, based on the

domain knowledge and the collection (for example, the number of clusters, clustering algorithm, static clustering, or query-specific clustering), whereas the topic signature model does not have these problems.

Latent topic models such as pLSI [13] assume that a document is generated by a set of topic models with certain distribution. Each topic model is further about the distribution of words in a given vocabulary. With topic model assumption, a document is modeled as follows:

$$p(w|d) = \sum_{i=1}^k p(t_i|d)p(w|t_i). \quad (4)$$

Here,  $k$  is the total number of topics in the corpus. The parameter  $p(w|t_i)$  is the probability of topic  $t_i$  generating word  $w$ . The parameter  $p(t_i|d)$  is the probability of document  $d$  being generated by topic  $t_i$ . Within the framework of latent topic models, a document can be associated with multiple topics and, thus, it overcomes the limitation of the cluster language models. Hoffman evaluated the pLSI model for retrieval tasks within the framework of vector space model [13]. The pLSI model significantly outperformed the LSI model and the standard raw term matching method. However, the size of four testing collections is far from the representative of realistic IR environments, and the baseline model is also far from the state of the art, making the effectiveness of the pLSI model on retrieval unclear.

The idea of topic signature is actually very similar to the latent topic. The major difference lies in their implementations, that is, the estimation of parameters. The number of free parameters  $p(t_i|d)$  and  $p(w|t_i)$  in the latent topic models is mainly in proportion to the number of documents for a large collection, which will cause a serious overfitting problem when the EM algorithm [8] is used for model estimations. The estimation process also lacks scalability because all parameters should be estimated simultaneously. The worst is that, when a new document is coming, there is no way to estimate the topic mixture  $p(t_i|d)$ . In our approach, we explicitly extract topic signatures from documents in the corpus. Thus, we can estimate each topic signature model  $p(w|t_i)$  separately. Furthermore, we can simply use a maximum likelihood estimator to approach  $p(t_i|d)$  no matter whether the document is new or not. In short, the estimation of parameters for topic signature language model is very efficient and scalable, as well as applicable, to new testing documents.

Wei and Croft [25] proposed an LDA-based document model for ad hoc retrieval. Unlike the pLSI model, where topic mixture is conditioned on each document, the LDA model samples topic mixture from a conjugate Dirichlet prior that remains the same for all documents [3]. This change can solve the overfitting problem and the problem of generating new document in pLSI. To make up the possible information loss, the LDA model is further interpolated with a simple language model. The final document model is

$$p(w|d) = \lambda \left( \frac{N_d}{N_d + u} p_{ML}(w|d) + \left( 1 - \frac{N_d}{N_d + u} \right) p(w|coll) \right) + (1 - \lambda) \sum_{i=1}^k p(t_i|d)p(w|t_i). \quad (5)$$

The LDA model improved the retrieval performance over both the simple language model and the cluster language model on five TREC collections [25]. The LDA model is estimated through Gibbs sampling, which is computationally intensive. Thus, compared to the topic signature language model, the LDA model suffers from the computing intensity and lack of scalability.

### 3 TOPIC SIGNATURE LANGUAGE MODELS

In this section, we describe topic signature language models in detail. First, we define two types of topic signatures and introduce the extraction algorithms. Second, a statistical model (that is, a distribution of words) is estimated for each topic that the corresponding topic signature represents. Third, topic signature models are used for document expansion (smoothing). Last, we discuss the scalability and complexity of the estimation of the topic signature language model.

#### 3.1 Context-Sensitive Topic Signatures

The implementation of topic signatures plays a crucial role in our CSSS approach. First, the topic signature must be context sensitive and, thus, it should contain at least two terms unless word sense is adopted. Second, constituents of a topic signature should have syntactic relation. Otherwise, we cannot count their frequency in a document and it becomes difficult to estimate their distributions. Third, it should be easy and efficient to extract topic signatures from texts. Following these criteria, we recommend two types of topic signatures: One is the ontology-based concept and the other is the multiword phrase. In this section, we formally define these two types of topic signature and briefly introduce the corresponding extraction algorithms.

##### 3.1.1 Ontology-Based Concept as Topic Signature

In our previous work [32], we implemented topic signatures as concept pairs, as inspired by Harabagiu and Lacatusu's topic representations [10]. Formally, a topic signature is defined with two order-free components, as in  $t(w_i, w_j)$ , where  $w_i$  and  $w_j$  are two concepts related to each other syntactically and semantically. Because two concepts in a pair help determine the context for each other, the meaning of a concept pair is often unambiguous and its semantic translation to individual concepts is very specific and accurate. However, the combination of two concepts causes a large vocabulary space, which makes it inefficient to index large collections. The distribution of concept pairs is also quite sparse and, thus, it is difficult to obtain sufficient data for many concept pairs to estimate their translation probabilities to individual concepts. Aware of the unambiguousness of a concept in an ontology, we simply use ontology-based concepts as topic signatures in this paper.

A *concept* is a unique meaning in a domain. It represents a set of synonymous terms in the domain. For example, C0020538 is a concept about the disease of hypertension in the UMLS Metathesaurus [35] and it also represents a set of synonymous terms, including *high blood pressure*, *hypertension*, and *hypertensive disease*. Therefore, concept-based indexing and searching helps relieve the synonymy and polysemy problems in IR, especially genomic IR, where a term (for example, a gene or a protein) might have many

**Example Sentence:**

A recent epidemiological study (C0002783, research activity) revealed that obesity (C0028754, disease) is an independent risk factor for periodontal disease (C0031090, disease).

**Word Index:** recent, epidemiological, study, research, activity, reveal, obesity, independent, risk, factor, periodontal, disease

**Concept Index:** C0002783, C0028754, C0031090

Fig. 1. The demonstration of concept extraction and indexing. Stop words are removed and words are stemmed.

synonyms while also representing a different concepts in different context [30].

In general, the extraction of concepts from texts is still a challenging problem. Fortunately, in the domain of biology and medicine, a large ontology called UMLS [35] was developed, which made the task of concept extractions possible. The extraction of biological concepts is a hot topic in bioinformatics and a survey of those methods can be found in [19]. However, most approaches only segment a sequence of words into phrases but do not further map the identified phrases into concepts. For this reason, we adopt MaxMatcher [31], which is a dictionary-based biological concept extraction tool, for the UMLS concept extractions.

In order to increase the extraction recall while retaining the precision, MaxMatcher uses approximate matches between the word sequences in text and the concepts defined in a dictionary or ontology such as the UMLS Metathesaurus. It outputs concept names and unique IDs representing a set of synonymous concepts. The unique concept IDs are used as an index in our experiments. In the example shown in Fig. 1, the underlined phrases are extracted concept names followed by the corresponding concept ID and semantic type. The details of the algorithm for MaxMatcher can be found in our previous work [31]. MaxMatcher has been evaluated on the GENIA corpus [36]. The precision and recall reached 71.60 percent and 75.18 percent, respectively, by using approximate match criterion.

### 3.1.2 Multiword Phrase as Topic Signature

The use of phrases has a long history in IR. A typical method for utilizing phrases will identify phrases within queries (for example, “star war” and “space program”), scan documents to identify query phrases, and score the document if it contains query phrases [20]. The recognition of query phrases within documents can be done in one of the following manners [20]:

- Boolean. This is also called conjunctive phrases [5]. All subterms of a query phrase co-occur in a document.
- Adjacent. This has the exact same form as the query phrase.
- Proximity. All subterms of a query phrase occur in close proximity in a document.

In this paper, we utilize multiword phrases in a different manner. We treat phrases frequently occurring in a given collection as topic signatures and try to find a set of individual words to represent the topic signature (the multiword phrase). Then, we can expand a document language model by statistically mapping topic signatures into query terms (individual words). For this purpose, we identify multiword phrases within only documents. The

**Example Sentence:**

How the many changes in the former Soviet Union (now the Commonwealth of Independent States) will affect the future of their space program remains to be seen.

**Word index:** change, form, soviet, union, commonwealth, independent, state, affect, future, space, program, remain, see

**Multiword Phrase Index:** Soviet Union, independent state, space program

Fig. 2. The demonstration of multiword phrase extraction and indexing. Stop words are removed and words are stemmed.

definition of phrase in this paper is roughly equivalent to the definition of query phrases in traditional phrase models. It is a sort of rigid noun phrase or collocation. It contains two or more individual words which are adjacent to each other in sequence. It often begins with an adjective or a noun and ends with a noun. The semantics of a phrase usually has the following types:

- Organization: International Business Machine Corp.,
- Person: George Bush, Ronald Reagan,
- Location: United States, Los Angeles, and
- Subject: Space Program, Star War.

We use a slightly modified version of Xtract [23] to extract phrases in documents. Xtract is designed to extract three types of collocations: predicative relations, rigid noun phrases, and phrasal templates. It begins with extracting significant bigrams using statistical techniques, then expands 2-Grams to N-Grams, and, finally, adds syntax constraint to the collocations. In Fagan’s notion of phrases [5], [9], the phrases extracted by Xtract are constrained by both statistical and syntactic criteria. In the original version, two words are defined as a bigram if and only if they co-occur within a sentence and their lexical distance is less than five words. Because we are only interested in rigid noun phrases, the first word is limited to an adjective or a noun, the second word must be a noun, and their distance threshold is set to four words in our implementation (see Fig. 2).

Xtract uses four parameters—strength ( $k_0$ ), spread ( $U_0$ ), peak z-score ( $k_1$ ), and percentage frequency ( $T$ )—to control the quantity and quality of the extracted phrases. In general, the bigger those parameters, the higher the quality but the smaller the quantity of phrases that Xtract produces. Smadja recommended a setting  $(k_0, k_1, U_0, T) = (1, 1, 10, 0.75)$  to achieve good results. In the experiment, we set those four parameters to  $(1, 1, 4, 0.75)$ . Xtract is an effective approach to the phrase extraction. The precision is about 80 percent, which is good enough for our IR use. It is also very efficient. For example, it takes only two hours to extract phrases from the AP 89 collection (84,678 documents) by using our Java version implementation, whereas Annie (a named entity recognition component of GATE [6]) takes about 12 hours to recognize entities from the same collection.

In the experiment, we also tried another two types of multiword phrases in order to increase phrase coverage. One is named entities (person, location, and organization) identified by GATE [6]. The other is WordNet noun phrases [18]. However, the extra phrases did not bring further improvement of IR performance. A possible explanation is that both GATE entities and WordNet noun phrases are purely “syntactic” phrases and those extra phrases (not

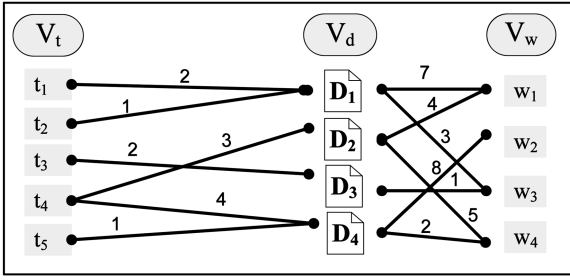


Fig. 3. Illustration of document indexing.  $V_t$ ,  $V_d$ , and  $V_w$  are the topic signature set, document set, and word set, respectively. The number on each line denotes the frequency of the corresponding topic signature or word in the document.

extracted by Xtract) are often infrequent in our testing collections. In our phrase language model, infrequent phrases (topic signature) result in little effect on document expansions.

### 3.2 Topic Signature Model Estimates

Suppose we have indexed all documents with individual terms and topic signatures (see Fig. 3). For each topic signature  $t_k$ , we have a set of documents ( $D_k$ ) containing that topic signature. Intuitively, we can use the document set  $D_k$  to approximate the semantic profile for  $t_k$ , that is, to determine the probability of mapping the signature to terms in the vocabulary. If all terms appearing in the document set center on the topic signature  $t_k$ , then we can simply use a maximum likelihood estimator and the problem is as simple as frequency counting. However, some terms address the issue of other topic signatures, whereas some are background terms of the whole collection. We use the generative model proposed in [27] to remove noise. Assume that the set of documents containing  $t_k$  is generated by a mixture model (that is, interpolating the topic model with the background collection model  $p(w|C)$ ):

$$p(w|\theta_{t_k}, C) = (1 - \alpha)p(w|\theta_{t_k}) + \alpha p(w|C). \quad (6)$$

Here, the coefficient  $\alpha$  is accounting for the background noise and  $\theta_{t_k}$  refers to the parameter set of the topic model associated with the topic signature  $t_k$ . In all of the experiments in this paper, the background coefficient  $\alpha$  is set to 0.5. Under this mixture language model, the log likelihood of generating the document set  $D_k$  is

$$\log p(D_k|\theta_{t_k}, C) = \sum_w c(w, D_k) \log p(w|\theta_{t_k}, C). \quad (7)$$

Here,  $c(w, D_k)$  is the document frequency of term  $w$  in  $D_k$ , that is, the co-occurrence count of  $w$  and  $t_k$  in the whole collection. The topic model for  $t_k$  can be estimated using the EM algorithm [8]. The EM update formulas are

$$\hat{p}^{(n)}(w) = \frac{(1 - \alpha)p^{(n)}(w|\theta_{t_k})}{(1 - \alpha)p^{(n)}(w|\theta_{t_k}) + \alpha p(w|C)}, \quad (8)$$

$$p^{(n+1)}(w|\theta_{t_k}) = \frac{c(w, D_k)\hat{p}^{(n)}(w)}{\sum_i c(w_i, D_k)\hat{p}^{(n)}(w_i)}. \quad (9)$$

Our topic signature model is significantly different from previous ones described in [2], [4], [14], [15] in two aspects.

TABLE 1  
Examples of Topic Signature Models

Space Program		Star War		Third World Debt	
Term	Prob.	Term	Prob.	Term	Prob.
space	0.101	star	0.088	debt	0.072
program	0.071	war	0.066	Brady	0.039
NASA	0.048	missile	0.06	loan	0.038
shuttle	0.043	strategy	0.051	world	0.038
astronaut	0.041	defense	0.051	treasury	0.037
launch	0.040	nuclear	0.043	bank	0.035
mission	0.038	space	0.034	Nicholas	0.034
flight	0.037	initialize	0.033	debtor	0.030
earth	0.037	Pentagon	0.032	trillion	0.027
moon	0.035	weapon	0.031	reduction	0.027
orbit	0.032	bomber	0.031	forgive	0.025
satellite	0.031	budget	0.028	monetary	0.025
Mar	0.030	stealthy	0.025	Mexico	0.025
explorer	0.028	program	0.025	economy	0.023
station	0.028	spend	0.024	billion	0.023
rocket	0.027	armed	0.023	reduce	0.022
technology	0.026	fiscal	0.022	burden	0.022
project	0.025	Reagan	0.021	lend	0.021
science	0.023	cut	0.021	creditor	0.021
budget	0.023	Bush	0.019	secretary	0.020

The three multiword phrases are automatically extracted from the collection of AP 89 by Xtract. We only list the top 20 topical words for each phrase. It is worth noting that the word “third” is removed from indexing as a stop word and, thus, it does not appear in the translation result of the third phrase.

First, previous models take an individual term as the topic signature and are unable to incorporate contextual information into the translation procedure. Our model uses context-sensitive topic signatures which are unambiguous in most cases. Thus, the resulting mapping will be more specific. From the three examples shown in Table 1, we can see that the phrase-word mapping is quite coherent and specific. Take the example of the phrase “space program.” If we estimate the topic models for its constituent terms “space” and “program” separately, then both models (see Fig. 4) contain mixed topics and are fairly general. Some terms such as *NASA*, *astronaut*, *moon*, *satellite*, *rocket*, and *Mars*, which are very much related to the subject of space program, appear in the phrase topic model but in neither of the subterm topic models.

Second, the method for model estimation is different. Berger and Lafferty [2] use document-query pairs to train translation probabilities. However, it is unlikely to obtain a large amount of real data. For this reason, they use synthetic data for model estimation. The title language model, proposed in [14], uses title-document pairs to train translation probabilities. The major drawback of the title model is that only a small portion of the terms in the vocabulary would appear in the title. The Markov chain model [15] deals with translations in a different fashion. However, the resulting query model is fairly general and the computation of the inverse matrix is prohibitive to large collections. Cao et al. [4] take into account word semantics when computing term associations, but they ignore the sense of words.

We truncate terms with extremely small probabilities in each topic model for two purposes. First, with a smaller translation space, the document smoothing will be much more efficient. Second, we assume that terms with

<b>Space:</b> space 0.245; shuttle 0.057; launch 0.053; flight 0.042; air 0.035; program 0.031; center 0.030; administration 0.026; develop 0.025; like 0.023; look 0.022; world 0.020; director 0.020; plan 0.018; release 0.017; problem 0.017; work 0.016; place 0.016; mile 0.015; base 0.014
<b>Program:</b> program 0.193; washington 0.026; congress 0.026; administration 0.024; need 0.024; billion 0.023; develop 0.023; bush 0.020; plan 0.020; money 0.020; problem 0.020; provide 0.020; writer 0.018; d 0.018; help 0.018; work 0.017; president 0.017; house .017; million 0.016; increase 0.016;

Fig. 4. The demonstration of word-word translation, which is estimated by the same approach described in Section 3.1. The translation results contain mixed topics and are fairly general in comparison with the result of the phrase-word translation.

extremely small probability are noise (that is, not semantically related to the given topic signature). In detail, we disregard all terms with translation probability less than 0.001 and renormalize the probabilities of the remaining terms.

### 3.3 Document Model Smoothing

Suppose we have indexed all documents in a given collection  $C$  with terms (individual words) and topic signatures, as illustrated in Fig. 3. The probability of mapping a topic signature  $t_k$  to any individual term  $w$ , denoted as  $p(w|t_k)$ , is also given. Then, we can easily obtain a document model as follows:

$$p_t(w|d) = \sum_k p(w|t_k)p_{ml}(t_k|d). \quad (10)$$

The likelihood of a given document generating the topic signature  $t_k$  can be estimated with

$$p_{ml}(t_k|d) = \frac{c(t_k, d)}{\sum_i c(t_i, d)}, \quad (11)$$

where  $c(t_i, d)$  is the frequency of the topic signature  $t_i$  in a given document  $d$ .

We refer to the above model as the translation model, after Berger and Lafferty's work [2]. As we discussed in the previous section, the translation from context-sensitive topic signatures to individual terms would be very specific. Thus, the smoothed (expanded) document models will be more accurate. However, not all topics in a document can be expressed by topic signatures (for example, multiword phrases). Take the example of AP88-90. A document in this collection contains 179 unique words but only contains 32 multiword phrases on the average (see Table 2). If only the translation model is used, then there will be serious information loss. A natural extension is to interpolate the translation model with a unigram language model. We use the two-stage method [28] to smooth the unigram language model:

$$p(Q|D) = \prod_{q \in Q} \left\{ (1 - \gamma) \frac{tf(q, D) + \mu p(q|C)}{|D| + \mu} + \gamma p(q|C) \right\}, \quad (12)$$

where  $p(q|C)$  is the collection background model and  $\gamma$  and  $\mu$  are two coefficients for tuning. We also refer to this

TABLE 2  
Average Numbers of Unique Words and Unique Topic Signatures per Document in Six Collections

Collection	avg. # of unique words per doc	avg. # of unique topic signatures per doc
Genomics 2004	71.3	39.2
Genomics 2005	75.2	37.6
AP89	180.1	31.8
AP88-89	178.6	31.7
WSJ90-92	196.6	35.6
SJMN91	164.2	25.3

smoothed unigram model as the simple language model or the baseline language model in this paper.

The final document model for retrieval use is described as follows: It is a mixture model with two components, namely, a simple language model and a translation model:

$$p_{bt}(w|d) = (1 - \lambda)p_b(w|d) + \lambda p_t(w|d). \quad (13)$$

The translation coefficient ( $\lambda$ ) controls the influence of the two components in the mixture model. With training data, the translation coefficient can be trained by optimizing the retrieval performance measure such as the average precision. In the experiments in this paper, we train the optimal translation coefficient on one collection and then apply the learned translation coefficient to other collections.

### 3.4 Scalability and Complexity

In comparison to the simple language models [18] and traditional probabilistic language models such as Okapi [22], the topic signature language model needs the following extra computational cost: 1) the extraction of topic signatures from documents in offline mode, 2) the estimation of topic models for each topic signature in offline mode, and 3) document model expansions based on topic signature translations in online mode. Fortunately, the additional computation scales very well and its complexity is acceptable in practice. Furthermore, the issue of scalability and complexity is significantly improved over the statistical translation model [2] and the LDA-based document model [25].

The extraction of topic signatures is time consuming compared with the individual term extraction. However, it does not cause a serious problem because it can be executed in the offline and incremental modes. In the experiment, the dragon toolkit [34] is used for document indexing. The dragon toolkit implements a Java version of Xtract [23] for multiword phrase extraction. Take the example of indexing the AP collection in Disks 1, 2, and 3 (about 240,000 news articles) on a Linux server. It takes about 15 minutes to index individual terms and three hours to index topic signatures (multiword phrases). From this example, we can see that the indexing time for topic signatures is acceptable as an offline task.

The estimation of topic models is highly computationally intensive. In general, the parameter space is in proportion to the number of documents in the corpus, the size of the vocabulary, and the number of topics. The computational complexity is in proportion to the number of documents, the number of topics, and the number of iterations for

convergence. Therefore, the estimation algorithms proposed in [2] and [25] do not scale very well and are time consuming for large collections. For example, the estimation of the LDA model for the AP collection using Gibbs sampling (please refer to [25] for detailed settings) costs about 72 hours, whereas our approach uses only 45 minutes to estimate topic models for all topic signatures. Our approach estimates topic models for each topic signature separately, which dramatically reduces the parameter space and makes the model converge with fewer iterations. Thus, our estimation approach increases the scalability and reduces the complexity.

The online document model expansion based on topic models is computationally intensive because it involves the summation of translation probabilities, as shown in (10). The complexity is in proportion to the number of topics for a document. The number of topics is equal to the number of unique terms in the statistical translation model [2], the number of latent topics in LDA-based models [25], and the number of unique topic signatures in the topic signature language model, respectively. As shown in Table 2, the number of topic signatures is significantly less than the document length and the number of latent topics in the LDA model (for example, the optimal number of topics is 800 in [25]) in typical testing collections. Thus, our approach has the lowest complexity during the stage of online document model expansions.

## 4 EXPERIMENTS WITH ONTOLOGY-BASED CONCEPTS

### 4.1 Evaluation Metrics and Baseline Models

Following the convention of TREC, we use the mean average precision (MAP) as the major performance metric and the overall recall at 1,000 documents as a supplemental metric. The noninterpolated average precision is defined as:

$$\frac{1}{|\text{Rel}|} \sum_{D \in \text{Rel}} \frac{|\{D' \in \text{Rel}, r(D') \leq r(D)\}|}{r(D)}, \quad (14)$$

where  $r(D)$  is the rank of document  $d$  and  $\text{Rel}$  is the set of relevant documents for a query  $Q$ . By averaging the noninterpolated average precision across all queries of a collection, we obtain the MAP for the collection.

In the experiment, we use the TSLM [28] as the first baseline. The exact formula for the two-stage model is described in (12). To show how strong the baseline is, we also compare the baseline to the famous Okapi model [22]. The exact formula for the Okapi model is shown as follows:

$$\text{Sim}(Q, D) = \sum_{q \in Q} \left\{ \frac{tf(q, D) \log \left( \frac{N - df(q) + 0.5}{df(q) + 0.5} \right)}{0.5 + 1.5 \frac{|D|}{\text{avg}_d l} + tf(q, D)} \right\}, \quad (15)$$

where

- $tf(q, D)$  is the term frequency of  $q$  in document  $D$ ,
- $df(q)$  is the document frequency for  $q$ , and
- $\text{avg}_d l$  is the average document length in the collection.

The major difference between the statistical translation model [2] and the proposed topic signature language model is that the latter incorporates the contextual information

TABLE 3  
The Descriptive Statistics of Testing Collections

Collections	Word	Concept	Rel./Doc	Q.Len/Q.#
Genomics 2004	92,362	65,257	8,268/42,251	6.4/50
Genomics 2005	80,168	57,879	4,584/35,474	6.0/49

into the document model expansions (smoothing). Thus, it is very natural to further compare the CSSS to the CISS. Because it is difficult to obtain a large number of real query-document pairs, we use word-word co-occurrence data to train a context-insensitive version of translation probabilities in the experiment. The parameter estimation algorithm is the same as the one for the context-sensitive version (that is, the translation from topic signature to individual words). The retrieval model is still the mixture of a TSLM and a translation model, as described in (13). However, the translation component is formulated slightly differently:

$$p_t(w|d) = \sum_k p(w|w_k) p_{ml}(w_k|d). \quad (16)$$

It statistically maps each individual word instead of a context-sensitive topic signature in a document onto query terms.

### 4.2 Testing Collections

Our current implementation of the concept-based topic signature extraction needs domain ontology. For this reason, we validate our CSSS method on genomic collections because UMLS can be used as the domain ontology for this area. The testing collections are TREC Genomics Tracks 2004 [11] and 2005 [12]. The original collection is a 10-year subset of Medline abstracts and contains about 4,600,000 abstracts. We only used the subcollection (that is, the human-relevance-judged document pool, which is 42,251 documents for 2004 and 35,474 documents for 2005) for our experiment. The ad hoc retrieval tasks of the two tracks include 50 topics (queries), respectively. The statistics of the testing collections are shown in Table 3.

### 4.3 Document Indexing and Query Processing

We index all documents with the UMLS-based concepts and individual words, as demonstrated in Figs. 1 and 3. For each document, we record the frequency count of each topic signature (that is, the UMLS concept), individual words, and the basic statistics. For each topic signature and individual words, we record their frequency count in each document, as well the basic statistics. For word indexing, stop words are removed and each word is stemmed. For topic signatures appearing in 10 or more documents, we estimate their topic models (that is, the translation probabilities) by using the EM algorithms.

The query formulation is fully automated. The extraction of query terms (individual words) from topic descriptions is the same as the process of document indexing. In the TREC 2004 Genomics Track, a topic was described in three sections: title, information need, and context. The information provided by the context section is a little noisy. Our pilot study showed that the baseline (both Okapi and TSLM) using the context section achieved the performance much worse than the one without context. For this reason, we only use the title section and information need section in the

TABLE 4  
Comparison of the Two-Stage Language Model (TSLM)  
to the Okapi Model

Collection	Recall			MAP		
	TSLM	Okapi	Change	TSLM	Okapi	Change
TREC04	6544	6847	+4.6%	0.352	0.369	+4.8%
TREC04†	6680	6869	+2.8%	0.384	0.370	-3.7%
TREC05	4093	4193	+2.4%	0.265	0.270	1.9%

The sign † indicates that the initial query is weighted.

experiment. In the TREC 2005 Genomics Track, query 135 was removed because it contains no relevant document.

As stated in [17], the query terms in the title section are clearly more important than those in the remaining sections. For this reason, we weight query terms according to the sections from which they are extracted. Following the method proposed in [17], we optimize the weight of different sections by maximizing the MAP of the baseline retrieval model. The optimal weights for the title section and the information need section are 1.0 and 0.2, respectively. In Tables 4, 5, and 6, the sign (†) indicates that the initial query is weighted.

#### 4.4 Effect of Document Smoothing

We set parameters  $\gamma$  and  $\mu$  in the TSLM to 0.05 and 200, respectively, because the language model achieves the best performance with this configuration. To give readers the sense of how good the baseline language model is, we also report the performance of the Okapi retrieval model in Table 4. The Okapi model is slightly better than the two-stage model, but, roughly, these two models are comparable to each other.

The translation coefficient ( $\lambda$ ) in the topic signature language model is optimized by maximizing the MAP on the TREC 2004 Genomics Track by using unweighted query. The learned optimal value is 0.3 and, then, we apply this learned value to other two collections. The result is shown in Table 5. In order to validate the significance of the improvement, we also run the paired-sample t-test. As expected, the topic signature language model outperforms the TSLM in terms of the average precision and overall recall at the significance level of 0.01 on both TREC 2004 and 2005.

To see the robustness of the topic signature language model, we change the settings of the translation coefficient. The variance of the MAP with the translation coefficient  $\lambda$  is shown in Fig. 5. When the translation coefficient ranges

TABLE 5  
The Comparison of the Two-Stage Language Model (TSLM) to  
the Topic Signature Language Model (That Is, the CSSS)

Collections		TSLM	CSSS	Change
TREC04	MAP	0.352	0.422	+19.9%**
	Recall	6544	7279	+11.2%**
TREC04†	MAP	0.384	0.446	+16.2%**
	Recall	6680	7395	+10.7%**
TREC05	MAP	0.265	0.322	+21.5%**
	Recall	4093	4291	+4.8%**

The signs \*\* and \* indicate that the improvement is statistically significant according to the paired-sample t-test at the levels of  $p < 0.01$  and  $p < 0.05$ , respectively. The sign † indicates that the initial query is weighted.

TABLE 6  
Comparison of the Context-Sensitive Semantic  
Smoothing (CSSS) to the Context-Insensitive  
Semantic Smoothing (CISS)

Collections		TSLM	CISS	vs. TSLM	CSSS	vs. CISS
Genomics 2004	MAP	0.352	0.408	+15.9%**	0.422	+3.4%*
	Recall	6544	7176	+9.7%**	7279	+1.4%*
Genomics 2004†	MAP	0.384	0.432	+12.5%**	0.446	+3.2%*
	Recall	6680	7359	+10.2%**	7395	+0.5%
Genomics 2005	MAP	0.265	0.322	+21.5%**	0.322	+0.0%
	Recall	4093	4283	+4.6%**	4291	+0.2%

The rightmost column is the change of the CSSS over the CISS. The signs \*\* and \* indicate that the improvement is statistically significant according to the paired-sample t-test at the levels of  $p < 0.01$  and  $p < 0.05$ , respectively.

from 0 to 0.9, the topic signature language model always performs better than the baseline on three collections. This shows the robustness of the new model. More interestingly, the best performance is achieved at the setting point of  $\lambda = 0.3$  for all three curves. After that point, the performance is downward. A possible explanation is that the extracted topic signatures do not capture all points of the document, but the baseline language model captures those missing points. For this reason, when the influence of the translation model is too high in the mixture model, the performance is downward and even worse than that of the baseline. Therefore, if we can find a better topic signature representation for documents and queries, or we can refine the extraction of topic signatures, then the IR performance might be further improved.

#### 4.5 Context Sensitive versus Context Insensitive

Basically, the CISS is based on the word-word mapping, as done in [2], [4], [14], and [15]. The comparison of the CISS to the CSSS is presented in Table 6. For each collection, we tune the translation coefficient ( $\lambda$ ) to maximize the MAP. The optimal  $\lambda$  is about 0.3 for all three collections. First, we can see that the CISS significantly outperforms the TSLM on all three collections. The gain of the CISS model over the baseline language model is consistent with the conclusions of previous work such as [2], [4], [14], and [15]. However, the CISS is slightly less effective than the CSSS, as expected.

Second, the improvement of the CSSS over the CISS seems to not be much on the Genomics Tracks. On Genomics Track 2005, there is almost no improvement. A possible explanation is that most document terms are biological terms such as protein, gene, and cell names. Compared to general terms such as words in news articles, the meaning of biological and medical terms (for example,

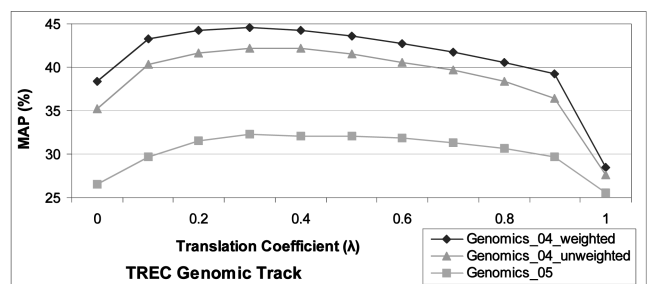


Fig. 5. The variance of MAP with the translation coefficient ( $\lambda$ ), which controls the influence of the translation model.

TABLE 7  
The Descriptive Statistics of Six Testing Collections

Collections	Word	Phrase	Rel./Doc	Q.Len/Q.#
AP89/1-50	145,349	114,096	3,301/84,678	3.4/47
AP88&89/51-100	204,970	127,736	6,101/164,597	3.4/49
AP88&89/101-150	204,970	127,736	4,822/164,597	4.0/50
WSJ90-92/101-150	135,864	75,687	2,049/74,520	3.8/48
WSJ90-92/151-200	135,864	75,687	2,041/74,520	4.6/49
SJMN91/51-100	173,727	95,986	2,322/90,257	3.4/48

p53, brca1, and orc1) is more consistent, even if without additional contextual constraints. Thus, the word-word mapping itself was very specific and accurate in the Genomics collections.

## 5 EXPERIMENTS WITH MULTIWORD PHRASES

### 5.1 Testing Collections

In this section, we evaluate the effectiveness of automated multiword phrases as topics signatures. Compared to ontology-based concepts, the extraction of multiword phrases does not need any external human knowledge and can be applied to any public domain. The model is validated on six TREC ad hoc collections from Disks 1, 2, and 3. We select these collections for three reasons. First, these collections are well studied and many published results are available for comparison. Second, the content of these collections is all about general news stories, on which the Xtract is supposed to work very well on the automated phrase extraction. Third, compared to the vocabulary in the Genomics collections, the vocabulary of news stories is more ambiguous and, thus, the CSSS is supposed to take the advantage over the CISS. The descriptive statistics of these testing collections are shown in Table 7.

### 5.2 Document Indexing and Query Processing

We obtain two separate indices, the word index and the phrase index, for each collection. For word indexing, each document is processed in a standard way. Words are stemmed (using porter-stemmer), and stop words are removed. We use a 319-word stop list compiled by van Rijsbergen. Xtract [23] is employed to extract multiword phrases from documents. For phrases appearing in 10 or more documents, we estimate their translation probabilities to single-word terms.

The query formulation is fully automated. For each collection, we remove all queries (topics) that contain no relevant documents. Early TREC topics are often described in multiple sections, including title, description, narrative, and concept. As many other studies did [1], [15], [16], [25], [27], we use only the title section. The extraction of query terms from topic descriptions is the same as the process of word indexing. That is, each topic is tokenized and stemmed and stop words are removed. The average length of queries and the total number of queries for each collection are listed in Table 7.

### 5.3 Effect of Document Smoothing

We set the parameters  $\gamma$  and  $\mu$  in the TSLM to 0.5 and 750, respectively, in the experiment because almost all collections achieve the optimal MAP at this configuration.

TABLE 8  
The Comparison of the Two-Stage Language Model (TSLM) to the Okapi Model

Collection/Topics	Recall			MAP		
	TSLM	Okapi	Change	TSLM	Okapi	Change
AP89/1-50	1621	1618	-0.2%	0.187	0.187	0.0%
AP88-89/51-100	3428	3346	-2.4%	0.252	0.239	-5.2%
AP88&89/101-150	3055	3087	+1.0%	0.219	0.220	+0.5%
WSJ90-92/101-150	1510	1488	-1.5%	0.239	0.249	+4.2%
WSJ90-92/151-200	1612	1624	+0.7%	0.314	0.304	-3.2%
SJMN91/51-100	1350	1348	-0.1%	0.190	0.184	-3.2%

Interestingly, the Okapi model and the TSLM have similar retrieval performance in the experiment, as shown in Table 8. This is also a kind of indication that both baseline models are well tuned.

The translation coefficient ( $\lambda$ ) in the topic signature language model is optimized by maximizing the MAP on the collection of AP 89 topics 1-50. The optimal value is 0.3 and we then apply this learned coefficient to other five collections. Interestingly, all collections achieve the best performance at the setting point of  $\lambda = 0.3$ . We then compare the result of the topic signature language model to the TSLM. The comparison is shown in Table 9. In order to validate the significance of the improvement, we also run the paired-sample t-test. The incorporation of the phrase-word translation improves both MAP and the overall recall over the baseline model on all six collections. Except for the recall on the collection of WSJ 90-92 topics 151-200, the improvements over the TSLM are all statistically significant at the level of  $p < 0.05$  or even  $p < 0.01$ . Considering that the baseline model is already very strong, we think that the topic signature language model is very promising to improve IR performance.

To see the robustness of the topic signature language model, we also change the settings of the translation coefficient. The variance of MAP with the translation coefficient  $\lambda$  is shown in Fig. 6. In a wide range from 0 to 0.6, the topic signature language model always performs better than the baseline on all six collections. This shows the robustness of the model. For all six curves in Fig. 6, the best performance is achieved at the setting point of  $\lambda = 0.3$ . After that point, the performance is downward. A possible explanation is that the extracted topic signatures (multiword phrases) do not capture all points of the document, but the TSLM captures those missing points. For this reason, when the influence of the translation model is too high in the mixture model, the performance is downward and even worse than that of the baseline.

### 5.4 Context Sensitive versus Context Insensitive

In news articles, many terms are ambiguous: A term may have different meanings in different contexts. Thus, the word-word translation may be fairly general and contains mixed topics. The phrase-word translation solves this problem since multiword phrases have very specific meaning and are mostly unambiguous.

The comparison of the CSSS to the CISS is shown in Table 10. For each collection, we tune the translation coefficient ( $\lambda$ ) to maximize the MAP of the CISS. The optimal  $\lambda$  is about 0.1 for all six collections, which is smaller than the optimal value for the CSSS ( $\lambda = 0.3$ ). It is also a

TABLE 9  
The Effect of Document Expansions Based on  
Phrase-Word Mapping

Collection/Topics		TSLM	CSSS	Change
AP89 1-50	MAP	0.187	0.206	+10.2%**
	Recall	1621	1748	+7.8%**
AP88-89 51-100	MAP	0.252	0.288	+14.3%**
	Recall	3428	3771	+10.0%*
AP88-89 101-150	MAP	0.219	0.246	+12.3%**
	Recall	3055	3445	+12.8%**
WSJ90-92 101-150	MAP	0.239	0.256	+7.1%**
	Recall	1510	1572	+4.1%*
WSJ90-92 151-200	MAP	0.314	0.334	+6.5%**
	Recall	1612	1620	+0.5%
SJM91 51-100	MAP	0.190	0.208	+9.5%**
	Recall	1350	1472	+9.0%**

The signs \*\* and \* indicate that the improvement is statistically significant according to the paired-sample t-test at the levels of  $p < 0.01$  and  $p < 0.05$ , respectively.

kind of indication that the word-word translation is much noisier than the phrase-word translation. From the experimental results, we can first see that the CISS greatly outperforms the TSLM and most of the improvements are statistically significant. Second, the CSSS has considerable gains over the CISS, especially on the measure of MAP.

In addition, the CSSS is computationally more efficient than the CISS. The CSSS is based on the phrase-word mapping, whereas the CISS is based on the word-word mapping. As shown in Table 2, an average document in testing collections contains about 180 unique words but only about 30 unique multiword phrases. In other words, the CSSS is six times faster than the CISS for the construction of co-occurrence data and the document model expansions (smoothing).

### 5.5 Versus Other Types of Phrases

The different types of phrases may have a different impact on retrieval performance. Fagan reported a significant improvement on some collections using “statistical” phrases but none with “syntactic” phrases in his thesis [9]. In this paper, we used kinds of phrases with both “syntactic” and “statistical” constraints extracted by Xtract and obtained very positive results. An interesting question is then raised up:

“Can other types of phrases (for example, WordNet phrases and named entities) still get positive results with the topic signature language model?”

To test this idea, we add WordNet noun phrases and named entities including person, organization, and location to the document index and see if the IR performance is further improved or even decreased. WordNet noun phrases are manually selected phrases. The named entities are automatically extracted by GATE [6] purely according to syntactic rules. Thus, neither of them is constrained by statistical criteria. Take the example of the AP 89 collection. Before adding extra phrases, the collection has 114,096 phrases. After adding WordNet noun phrases and named entities, the number of phrases is increased by about 50,000. However, the increase of phrase coverage does not make any improvement on the IR performance. The other five collections are in the similar case. Examining the extra noun phrases in a closer look, we find out that most of those phrases are infrequent in the testing collections. Actually,

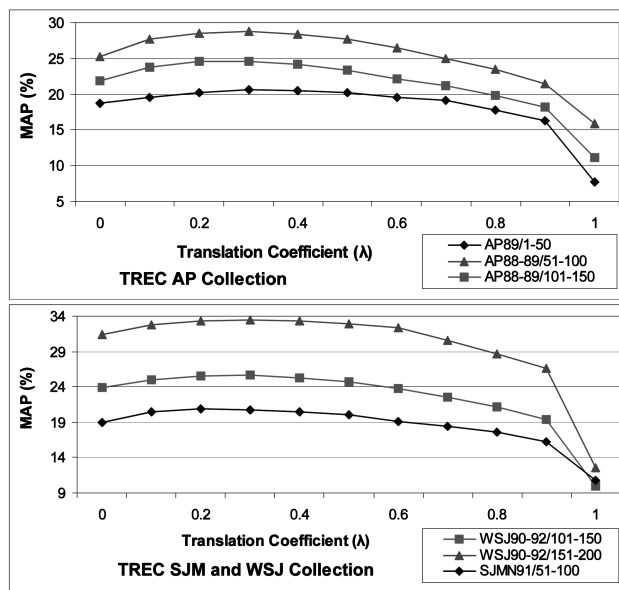


Fig. 6. The variance of MAP with  $\lambda$ , which controls the influence of the context-sensitive translation component in the mixture language model.

the majority of phrases frequently occurring in the collection are already extracted by Xtract. Those infrequent phrases will have little effect on the document model expansions and thus have no effect on retrieval performance. Therefore, in order to make the topic signature (phrase) language model effective, we should use phrases frequently occurring in the collection or constrained by “statistical” criteria.

## 6 CONCLUSIONS AND FUTURE WORK

In this paper, we proposed a topic signature language model for ad hoc text retrieval. This new model decomposed a document into a set of weighted context-sensitive topic signatures and then mapped those topic signatures into individual query terms. Because the topic signature itself contained contextual information, the document model expansion based on topic signatures would be more

TABLE 10  
Comparison of the Context-Sensitive Semantic Smoothing  
(CSSS) to the Context-Insensitive Semantic Smoothing (CISS)

Collections		TSLM	CISS	vs. TSLM	CSSS	vs. CISS
AP89 1-50	MAP	0.187	0.195	+4.3%*	0.206	+5.6%
	Recall	1621	1730	+6.7%*	1748	+1.0%
AP88-89 51-100	MAP	0.252	0.272	+7.9%*	0.288	+5.9%*
	Recall	3428	3735	+9.0%*	3771	+1.0%
AP88-89 101-150	MAP	0.219	0.235	+7.3%**	0.246	+4.7%
	Recall	3055	3237	+6.0%*	3445	+6.4%*
WSJ90-92 101-150	MAP	0.239	0.244	+2.1%	0.256	+4.9%*
	Recall	1510	1568	+3.8%**	1572	+0.3%
WSJ90-92 151-200	MAP	0.314	0.324	+3.2%	0.334	+3.1%
	Recall	1612	1646	+2.1%*	1620	-1.6%
SJM91 51-100	MAP	0.190	0.199	+4.7%*	0.208	+4.5%
	Recall	1350	1427	+5.7%**	1472	+3.2%

The rightmost column is the change of CSSS over CISS. The signs \*\* and \* indicate that the improvement is statistically significant according to the paired-sample t-test at the levels of  $p < 0.01$  and  $p < 0.05$ , respectively.

accurate as compared to the document model expansion based on context-insensitive term mapping proposed in previous work such as [2], [4], [14] and thus improved the retrieval performance.

We implemented two types of topic signatures in this paper. When a domain-specific ontology is available, ontology-based concepts can be used as topic signatures. Otherwise, automated multiword phrases are an alternative. We evaluated the effectiveness of ontology-based concepts on the TREC Genomics Tracks 2004 and 2005 and the effectiveness of multiword phrases on TREC Ad Hoc Track Disks 1, 2, 3. The topic signature language model significantly outperformed the TSLM on all collections. We further implemented a context-insensitive version of semantic smoothing. It has the same framework as the topic signature language model, but the document model expansion (smoothing) is based on the context-insensitive word-word mapping rather than the context-sensitive signature-word mapping. As expected, it is less effective than the CSSS, though it achieves significant improvement over the simple language model.

The topic signature language is the linear interpolation of the simple language model and the topic-signature-based translation model. It is required to set the translation coefficient, which controls the influence of the translation component in the mixture model. It is somewhat ad hoc in nature. Fortunately, the experiments showed the robustness of the model. When the translation coefficient took different values in a wide range (0-0.9 for ontology-based concepts and 0-0.6 for multiword phrases), the topic signature language model always performed better than the baseline. More interestingly, all collections achieved the best MAP at the same setting (that is,  $\lambda = 0.3$ ). This means that it is feasible to train the optimal translation coefficient on one collection and then apply the learned coefficient to other collections in practice.

We also found out that two factors would affect the effectiveness of the topic signature language model. One is the degree of the ambiguity of terms in the collection. If the terms (for example, in news collections) are very ambiguous, then the topic signature model (that is, the CSSS) can take much advantage over the CISS. The other factor is the occurrence frequency of the topic signatures in the collection. If the topic signatures infrequently occur in the collection, then the model has little effect on improving the IR performance.

This paper made the following contributions: First, we presented a new document representation, that is, representing a document as a set of weighted topic signatures and terms. The new representation could be applied to other retrieval, summarization, and text classification tasks. Second, we proposed an EM-based method to estimate the semantic relationships between context-sensitive topic signatures and single-word terms simply using co-occurrence data and then formalized the approach to document expansions based on topic signature mapping. Third, we empirically proved the superiority of the CSSS over the CISS, as well as the simple background smoothing.

Probabilistic topical models such as pLSI [13] and LDA [25] also take the context into account and thus can handle the word polysemy problem. In this paper, we analyzed their computational complexity in the setting of IR and concluded that these two models were computationally less

efficient than the topic signature language model in the stage of offline topic model estimation and the stage of online document model smoothing. However, the comparison of the effectiveness of three models on retrieval tasks is still unclear. It should be interesting to have a comprehensively comparative study on these three models in the future with respect to their efficiency and effectiveness for ad hoc text retrieval.

Besides how we can optimize the mixture weights of the topic signature language model remains an open issue. In this paper, we empirically tuned a fixed translation coefficient on the training data set and achieved good results. Ideally, the translation coefficient should be conditioned on each document because the relative information provided by the topic signatures varied with different documents. In addition, the topic signature language model can also be applied to applications other than IR. Traditional text mining problems such as text clustering and text classification are also based on document models. Thus, it is natural to extend the application of the topic signature language model to those areas. Our previous work [33] successfully applied this model to agglomerative document clustering. In the future, we will further evaluate its effectiveness in related areas.

## ACKNOWLEDGMENTS

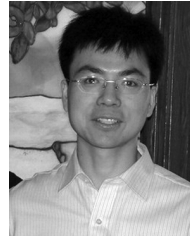
This work is supported in part by US National Science Foundation (NSF) Career Grant IIS 0448023, NSF CCF 0514679, Pennsylvania Department of Health Tobacco Settlement Formula Grant (No. 240205 and No. 240196), and Pennsylvania Department of Health Grant (No. 239667).

## REFERENCES

- [1] J. Bai, J.Y. Nie, and G. Cao, "Context-Dependent Term Relations for Information Retrieval," *Proc. Empirical Methods in Natural Language Processing (EMNLP '06)*, July 2006.
- [2] A. Berger and J. Lafferty, "Information Retrieval as Statistical Translation," *Proc. 22nd Ann. Int'l ACM Conf. Research and Development in Information Retrieval (SIGIR '99)*, pp. 222-229, 1999.
- [3] D. Blei, A. Ng, and M. Jordan, "Latent Dirichlet Allocation," *J. Machine Learning Research*, vol. 3, pp. 993-1022, 2003.
- [4] G. Cao, J.Y. Nie, and J. Bai, "Integrating Word Relationships into Language Models," *Proc. 28th Ann. Int'l ACM Conf. Research and Development in Information Retrieval (SIGIR '05)*, pp. 298-305, 2005.
- [5] W.B. Croft, H.R. Turtle, and D.D. Lewis, "The Use of Phrases and Structured Queries in Information Retrieval," *Proc. 14th Ann. Int'l ACM Conf. Research and Development in Information Retrieval (SIGIR '91)*, pp. 32-45, 1991.
- [6] H. Cunningham, "GATE: A General Architecture for Text Engineering," *Computers and the Humanities*, vol. 36, pp. 223-254, 2002.
- [7] S. Deerwester, T.S. Dumais, W.G. Furnas, K.T. Landauer, and R. Harshman, "Indexing by Latent Semantic Analysis," *J. Am. Soc. Information Science*, vol. 41, no. 6, pp. 391-407, 1990.
- [8] A.P. Dempster, N.M. Laird, and D.B. Rubin, "Maximum Likelihood from Incomplete Data via the EM Algorithm," *J. Royal Statistical Soc.*, vol. 39, pp. 1-38, 1977.
- [9] J. Fagan, "Experiments in Automatic Phrase Indexing for Document Retrieval: A Comparison of Syntactic and Non-Syntactic Methods," PhD dissertation, Technical Report 87-868, Computer Science Dept., Cornell Univ., 1987.
- [10] S. Harabagiu and F. Lacatusu, "Topic Themes for Multi-Document Summarization," *Proc. 28th Ann. Int'l ACM Conf. Research and Development in Information Retrieval (SIGIR '05)*, pp. 42-48, 2005.

- [11] W. Hersh et al., "TREC 2004 Genomics Track Overview," *Proc. 13th Text Retrieval Conf. (TREC '04)*, 2004.
- [12] W. Hersh et al., "TREC 2005 Genomics Track Overview," *Proc. 14th Text Retrieval Conf. (TREC '05)*, 2005.
- [13] T. Hoffman, "Probabilistic Latent Semantic Indexing," *Proc. 22nd Ann. Int'l ACM Conf. Research and Development in Information Retrieval (SIGIR '99)*, pp. 50-57, 1999.
- [14] R. Jin, A. Hauptmann, and C. Zhai, "Title Language Model for Information Retrieval," *Proc. 25th Ann. Int'l ACM Conf. Research and Development in Information Retrieval (SIGIR '02)*, pp. 42-48, 2002.
- [15] J. Lafferty and C. Zhai, "Document Language Models, Query Models, and Risk Minimization for Information Retrieval," *Proc. 24th Ann. Int'l ACM Conf. Research and Development in Information Retrieval (SIGIR '01)*, pp. 111-119, 2001.
- [16] X. Liu and W.B. Croft, "Cluster-Based Retrieval Using Language Models," *Proc. 24th Ann. Int'l ACM Conf. Research and Development in Information Retrieval (SIGIR '01)*, pp. 186-193, 2001.
- [17] D. Miller, T. Leek, and M.R. Schwartz, "A Hidden Markov Model Information Retrieval System," *Proc. 22nd Ann. Int'l ACM Conf. Research and Development in Information Retrieval (SIGIR '99)*, pp. 214-221, 1999.
- [18] G.A. Miller, "WordNet: A Lexical Database for English," *Comm. ACM*, vol. 38, no. 11, pp. 39-41, 1995.
- [19] R.J. Mooney and R. Bunescu, "Mining Knowledge from Text Using Information Extraction," *SIGKDD Explorations*, special issue on text mining and natural language processing, vol. 7, no. 1, pp. 3-10, 2005.
- [20] J. Pickens and W.B. Croft, "An Exploratory Analysis of Phrases in Text Retrieval," *Proc. RIAO Conf. '00*, pp. 1179-1195, 2000.
- [21] J. Ponte and W.B. Croft, "A Language Modeling Approach to Information Retrieval," *Proc. 21st Ann. Int'l ACM Conf. Research and Development in Information Retrieval (SIGIR '98)*, pp. 275-281, 1998.
- [22] S.E. Robertson et al., "Okapi at TREC-4," *Proc. Fourth Text Retrieval Conf. (TREC '95)*, 1995.
- [23] F. Smadja, "Retrieving Collocations from Text: Xtract," *Computational Linguistics*, vol. 19, no. 1, pp. 143-177, 1993.
- [24] D. Song and P.D. Bruza, "Towards Context-Sensitive Information Inference," *J. Am. Soc. Information Science and Technology*, vol. 54, pp. 321-334, 2003.
- [25] X. Wei and W.B. Croft, "LDA-Based Document Models for Ad-Hoc Retrieval," *Proc. 29th Ann. Int'l ACM Conf. Research and Development in Information Retrieval (SIGIR '06)*, pp. 178-185, 2006.
- [26] C. Zhai and J. Lafferty, "A Study of Smoothing Methods for Language Models Applied to Ad Hoc Information Retrieval," *Proc. 24th Ann. Int'l ACM Conf. Research and Development in Information Retrieval (SIGIR '01)*, pp. 334-342, 2001.
- [27] C. Zhai and J. Lafferty, "Model-Based Feedback in the Language Modeling Approach to Information Retrieval," *Proc. 10th Int'l Conf. Information and Knowledge Management (CIKM '01)*, pp. 403-410, 2001.
- [28] C. Zhai and J. Lafferty, "Two-Stage Language Models for Information Retrieval," *Proc. ACM Conf. Research and Development in Information Retrieval (SIGIR '02)*, 2002.
- [29] X. Zhou, X. Hu, X. Lin, H. Han, and X. Zhang, "Relation-Based Document Retrieval for Biomedical Literature Databases," *Proc. 11th Int'l Conf. Database Systems for Advanced Applications (DASFAA '06)*, pp. 689-701, Apr. 2006.
- [30] X. Zhou, X. Zhang, and X. Hu, "Using Concept-Based Indexing to Improve Language Modeling Approach to Genomic IR," *Proc. 28th European Conf. Information Retrieval (ECIR '06)*, pp. 444-455, Apr. 2006.
- [31] X. Zhou, X. Zhang, and X. Hu, "MaxMatcher: Biological Concept Extraction Using Approximate Dictionary Lookup," *Proc. Ninth Biennial Pacific Rim Int'l Conf. Artificial Intelligence (PRICAI '06)*, pp. 1145-1149, Aug. 2006.
- [32] X. Zhou, X. Hu, X. Zhang, X. Lin, and I.-Y. Song, "Context-Sensitive Semantic Smoothing for the Language Modeling Approach to Genomic IR," *Proc. 29th Ann. Int'l ACM Conf. Research and Development on Information Retrieval (SIGIR '06)*, pp. 70-77, Aug. 2006.
- [33] X. Zhou, X. Zhang, and X. Hu, "Semantic Smoothing of Document Models for Agglomerative Clustering," *Proc. 20th Int'l Joint Conf. Artificial Intelligence (IJCAI '07)*, pp. 2928-2933, Jan. 2007.

- [34] X. Zhou, X. Zhang, and X. Hu, "The Dragon Toolkit Developer Guide," Data Mining and Bioinformatics Laboratory, Drexel Univ., <http://www.dragontoolkit.org/tutorial.pdf>, 2007.
- [35] UMLS, <http://www.nlm.nih.gov/research/umls/>, 2007.
- [36] GENIA Corpus, <http://www-tsujii.is.s.u-tokyo.ac.jp/GENIA/>, 2007.



**Xiaohua Zhou** received the bachelor's and master's degrees from Shanghai Jiao Tong University, China, in 1999 and 2002, respectively. He is a PhD candidate in the College of Information Science and Technology at Drexel University, Philadelphia. His current research interests include information retrieval, text data mining, and information extraction. He is a student member of the IEEE.



**Xiaohua Hu** received the BSc degree in software from Wuhan University in 1985, the MEng degree in computer engineering from the Institute of Computing Technology, Chinese Academy of Science, in 1988, the MSc degree in computer science from Simon Fraser University, Canada, in 1992, and the PhD degree in computer science from the University of Regina, Canada, in 1995. He is currently an associate professor and the founding director of the Data Mining and Bioinformatics Laboratory in the College of Information Science and Technology at Drexel University. He is also serving as the IEEE Computer Society Bioinformatics and Biomedicine Steering Committee Chair and the IEEE Computational Intelligence Society Granular Computing Technical Committee Chair (2007-2008). He joined Drexel University in 2002. Earlier, he worked as a research scientist in some world-leading R&D centers such as Nortel Research Center and GTE Laboratories. In 2001, he founded the DMW Software in Silicon Valley, California. His current research interest includes biomedical literature data mining, bioinformatics, text mining, semantic Web mining and reasoning, rough set theory and application, information extraction, and information retrieval. He has published more than 140 peer-reviewed research papers in various journals, conferences, and books, and coedited nine books/proceedings. He has received a few prestigious awards, including the 2005 NSF Faculty Early Career Development (NSF Career) Award, the Best Paper Award from the 2007 International Conference on Artificial Intelligence, the Best Paper Award from the 2004 IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology, the 2006 IEEE Granular Computing Outstanding Service Award, and the 2001 IEEE Data Mining Outstanding Service Award. He is the founding editor-in-chief of the *International Journal of Data Mining and Bioinformatics*, and associate editor/editorial board member of four international journals. His research projects are funded by the US National Science Foundation (NSF), US Department of Education, and the Pennsylvania Department of Health. He is a member of the IEEE.



**Xiaodan Zhang** received the bachelor's degree in library and information science from Northeast Normal University, Changchun, China, in 1997, and the master's degree in computer science from Jinan University, Guangzhou, China, in 2003. He is a PhD student in the College of Information Science and Technology at Drexel University in Philadelphia. His research topic covers graph-based, model-based, and semantic-based text data mining. He is a student member of the IEEE.

► For more information on this or any other computing topic, please visit our Digital Library at [www.computer.org/publications/dlib](http://www.computer.org/publications/dlib).