

Using Concept-Based Indexing to Improve Language Modeling Approach to Genomic IR*

Xiaohua Zhou, Xiaodan Zhang, and Xiaohua Hu

College of Information Science & Technology, Drexel University,
3141 Chestnut Street, Philadelphia, PA 19104
xiaohua.zhou@drexel.edu, {xzhang, thu}@cis.drexel.edu

Abstract. Genomic IR, characterized by its highly specific information need, severe synonym and polysemy problem, long term name and rapid growing literature size, is challenging IR community. In this paper, we are focused on addressing the synonym and polysemy issue within the language model framework. Unlike the ways translation model and traditional query expansion techniques approach this issue, we incorporate concept-based indexing into a basic language model for genomic IR. In particular, we adopt UMLS concepts as indexing and searching terms. A UMLS concept stands for a unique meaning in the biomedicine domain; a set of synonymous terms will share same concept ID. Therefore, the new approach makes the document ranking effective while maintaining the simplicity of language models. A comparative experiment on the TREC 2004 Genomics Track data shows significant improvements are obtained by incorporating concept-based indexing into a basic language model. The MAP (mean average precision) is significantly raised from 29.17% (the baseline system) to 36.94%. The performance of the new approach is also significantly superior to the mean (21.72%) of official runs participated in TREC 2004 Genomics Track and is comparable to the performance of the best run (40.75%). Most official runs including the best run extensively use various query expansion and pseudo-relevance feedback techniques while our approach does nothing except for the incorporation of concept-based indexing, which evidences the view that semantic smoothing, i.e. the incorporation of synonym and sense information into the language models, is a more standard approach to achieving the effects traditional query expansion and pseudo-relevance feedback techniques target.

1 Introduction

Biomedical literature contains a wealth of valuable information. How to help scientists find desired information effectively and efficiently is an important research endeavor. In recent years, genomic information retrieval (GIR) is getting more and more attention from IR community. TREC Genomic Track has attracted lots of talented IR researchers to participate in.

* This research work is supported in part from the NSF Career grant (NSF IIS 0448023). NSF CCF 0514679 and the research grant from PA Dept of Health.

However, GIR is challenging IR community most likely due to the following reasons. First, unlike general searching that Google and Yahoo are working on, GIR are working for the scientists who have very specific information need. Second, GIR is dealing with a huge collection of biomedical literature that hinders many existing IR approaches that may be backed by a perfect theoretical model but not scalable to large document collections. Third, in genomic-related literature, a term is often comprised of multiple words; the word-based unigram IR models may lose the semantics of the term. Last, severe synonym and polysemy problem would cause trouble while an IR system tries to match query terms with indexing terms according to their strings instead of meanings.

In this paper, we focus on addressing the synonym and polysemy problem in GIR rather than attempting to solve all the problems. On one hand, synonyms of terms such as genes, proteins, cells and diseases are widely used in biomedical literature. On the other hand, the polysemy of many terms plus the use of partial names and abbreviations have caused the ambiguity of terms. The synonym and polysemy have affected the performance of genomic IR. A fundamental way to solve this problem is to index and search documents through a set of concepts. A concept has a unique meaning in a domain and therefore will not cause any ambiguity. All synonymous terms in the domain will share same concept identities and thus concept-based indexing will easily solve the synonym problem too.

The sense-based information retrieval is a kind of implementation of concept-based indexing and searching. However, word sense disambiguation (WSD) is a challenging task in the area of natural language processing (NLP). The performance (e.g. precision and recall) of WSD in general domain is not satisfying yet, which discourages IR researchers to incorporate word sense into their IR models. Some researchers reported positive outcome of sense-based IR models [15] but most of them failed to show any performance improvement partially due to the low accuracy of WSD in general domain [11]. Furthermore, word senses can not capture well the meaning of many terms in a technical domain such as biomedicine. For example, three individual word senses can not express the meaning of the concept "*high blood pressure*".

Many alternative approaches are then proposed to addressing the synonym and polysemy issue in IR. Latent semantic indexing (LSI) [2] tries to identify the latent semantic structure between terms; thus it can in part solve the synonym problem. However it is not suited to large document collections because the factorization of large matrix is prohibitive. Meanwhile, LSI can not handle the polysemy problem well. Vector space models and other traditional probabilistic models [10, 14] use query expansions to relax the synonym problem. Unlike LSI that is well supported by solid mathematical models, various query expansion techniques are often heuristic. But they achieve great success in IR practice. The translation model [1] extended from the basic unigram language model is a more formal approach for achieving the effects that query expansion techniques target. Berger and Lafferty reported significant improvement of IR performance with translation models [1]. However, there are several difficulties with translation model approach to semantic smoothing under language modeling framework (refer to Section 2 for details) [4].

Recent developments in large domain ontology such as UMLS¹ and statistical language modeling approach to information retrieval lead us to a re-examination of the concept-based indexing and searching. The language modeling approach to IR, initially proposed by Ponte and Croft [9], has been popular with IR community in recent years due to its solid theoretical foundation and promising empirical retrieval performance. We think a well-motivated retrieval framework such as language models might well take the full advantage of concept-based indexing. Meanwhile, the availability of large domain ontology will allow us to extract concepts from documents and queries efficiently and effectively.

To verify our idea, we build a prototyped IR system that indexes and searches documents through both controlled concepts and phrases, and then conduct a comparative experiment on the TREC 2004 Genomic Track data using a basic unigram language model for retrieval. The concept-based approach achieved a 36.94% MAP (mean average precision), significantly higher than the 29.17% MAP of the baseline approach (using phrases for index and search). The result of the concept-based approach is also significantly superior to the average performance (21.72%) of the official runs in TREC 2004 Genomic Track and is comparable to the performance of the best run (40.75%). Considering most official runs including the best run extensively used various query expansion and pseudo-relevance feedback techniques while our approach did nothing except for the incorporation of concept-based indexing, the concept-based approach demonstrated its effectiveness on solving synonym and polysemy issue in IR.

The rest of the paper is organized as follows: Section 2 describes the background of language modeling approach to IR. Section 3 presents a generic ontology-based approach to the concept extraction. Section 4 shows the experiment design and result. A short conclusion finishes the paper.

2 Language Modeling Approach to IR

In this section, we shortly review the work on language modeling approach to IR and point out the urgency of the development of semantic smoothing approaches and then propose our concept approach that directly uses concepts to index and search documents with the language modeling framework.

Language modeling approach to information retrieval (IR) was firstly proposed by Ponte and Croft [9]. Basically, the language model uses the generative probability of a query according to the language model of each document in the collection, $p(q|d)$, to rank the document for IR. Lafferty and Zhai further made the underlying semantics of the language model clear by linking the notion of relevance to the language model [5]. Under their framework, the relevance of a document to the query is defined as (2.1). Assuming the document is independent of the query conditioned on the event $R = \bar{r}$, the ranking formula is reduced to (2.2). Further ignoring the document prior such as PageRank used by Google in (2.2), the rank formula could be further reduced to be as simple as (2.3).

¹ <http://www.nlm.nih.gov/research/umls/>

$$\log \frac{p(r|Q, D)}{p(\bar{r}|Q, D)} = \log \frac{p(Q|D, r)}{p(Q|D, \bar{r})} + \log \frac{p(rD)}{p(\bar{r}|D)} \quad (2.1)$$

$$\stackrel{\text{rank}}{=} \log p(Q|D, r) + \log \frac{p(rD)}{p(\bar{r}|D)} \quad (2.2)$$

$$\stackrel{\text{rank}}{=} \log p(Q|D, r) \quad (2.3)$$

Let $Q = (A_1, A_2, \dots, A_m)$ and assume that the attributes (terms) are independent given R and the document D . The ranking formula is then transformed to (2.4) and the term frequency in each document is used to estimate the term $\log p(A_i | D, r)$. For the simplicity of the notation, we will use (2.5) as the basic ranking formula for IR in the present paper.

$$\log p(Q|D, r) = \sum_i \log p(A_i | D, r) \quad (2.4)$$

$$\log p(Q|D) = \sum_i \log p(A_i | D) \quad (2.5)$$

However, some query terms may not appear in a given document; thus language models for IR must be smoothed because zero probability can not be assigned to query terms. The *Jelinek-Mercer method* is one of the simplest ways to smooth language models [17]. It involves a linear interpolation of the maximum likelihood model with a background collection model, using a coefficient λ to control the influence of the background collection model C :

$$\log p_\lambda(a_i | d) = \log\{(1 - \lambda)p(a_i | d) + \lambda p(a_i | C)\} \quad (2.6)$$

Semantic smoothing, which incorporates synonym and sense information into the language model, is regarded as a potentially more effective smoothing approach [4]. With semantic smoothing, a document containing term *high blood pressure* may be retrieved for the query term *hypertension*; a document containing term *ferroportin-1* may not be retrieved for query term *ferroportin-1* because the former refers to a gene in human while the latter refers to a gene in mouse.

Berger and Lafferty present a translation model [1] that maps a document term t into a query term a_i . With term translations, the estimation of the generative probability for query term a_i becomes (2.7). In the simplest way, a document term can be translated into a query term with high probability if they are synonyms to each other. Thus, the translation model is kind of semantic smoothing. It achieved significant improvement in practice over the baseline system as described in [9].

$$\log p(a_i | d) = \log \sum_j p(a_i | t_j) p(t_j | d) \quad (2.7)$$

However, there are several difficulties with translation model approach to semantic smoothing in language modeling framework [4]. First, the estimation of translation probability would be a problem due to the lack of sufficient training data. Second, the calculation of the ranking score would be prohibitive for large document collections. Third, it can not incorporate sense information into the language model, i.e. it can not handle polysemy problem well.

We propose in this paper the direct use of controlled concepts for indexing and searching with the framework of language models. Except terms (often words) will be replaced by concepts as indexing and searching unit, no additional modification is required on the basic unigram language model. Thus the concept approach keeps language models as simple as described in (2.5) and (2.6). Furthermore, the calculation of the ranking score will be very efficient in comparison with the translation models. In addition, the concept approach solves both synonym and polysemy problems in IR. The major concern with this approach may be the extraction of concepts. However, with the availability of human-coded domain ontology, we are able to extract multi-word concept names with high accuracy. The further disambiguation of concept name, i.e. mapping a concept name to a unique concept ID in the domain according to the contextual information, is much easier than word sense disambiguation because term polysemy in technical domains such as biomedicine is rarer than generic domains. Therefore, it is reasonable to expect good overall performance of concept extractions.

3 Concept Extraction and Indexing Schema

In this section, we will briefly review the past work on biological term extractions and then introduce our generic ontology-based approach. In general, the concept extraction is done in two steps. In the first step, we extract multi-word concept names such as “*high blood pressure*”. We call them *phrases* in this paper. Because a concept name may correspondence to more than one concept ID in UMLS, we need the second step to disambiguate the concept name using the contextual information.

The approaches to biological term extraction roughly fall into two categories, with dictionary [16, 18] or without dictionary [7, 8, 12, 13]. The latter approaches use either hand-coded or machine learned rules to extract terms. It is able to recognize new terms, but it assign semantic class rather than concept IDs to extracted terms. For this reason, we do not use this line of approaches. The dictionary-based approaches use either noun phrase [16] produced by shallow parsers or part of speech patterns [18] to generate term candidates and then check the candidates with the dictionary. Both of them recognize terms based on exact character matching and would yield high precision. However, the extraction recall is often very low because a term name usually has many variants but a dictionary collects very few term variants.

$$I(w) = \max\{S_j(w) \mid j \leq n\} = \max\left\{\frac{1/N(w)}{\sum_i 1/N(w_{ji})} \mid j \leq n\right\} \quad (3.1)$$

To overcome the limitation of exact character matching, we develop an IE system called MaxMatcher that is namely able to recognize concept names by approximate matching. The basic idea of this approach is to capture the important tokens (not all tokens) of a concept name. For example, the token *gyrb* is obviously important to the concept *gyrb protein*; we will treat it as a concept name even if the token *protein* is not present. So the problem is reduced to how to score the importance of each token to a given concept name. Formally, given a concept that has n concept names or variants and let $S_j(w)$ denotes the importance of the token w to the j -th variant name, and let $N(w)$ denotes the number of concepts whose variant names contain token w in the dictionary, and let w_{ji} denotes the i -th token in the j -th variant name of the concept, the importance of w to the concept is defined as in (3.1).

Table 1. Demonstrate the calculation of the importance score of each token to concept C0120543 in UMLS (this concept has three variant names). The number in the parenthesis of the first column is the number of concepts whose variant names contain that token. The final importance score of each token to this concept is listed in the rightmost column.

Token	gyrb protein	gyrb gene product	DNA gyrase subunit b	Score
gyrb (1)	0.99998	0.99990		0.99998
protein (47576)	0.00002			0.00002
gene (22186)		0.00005		0.00005
product (18724)		0.00005		0.00005
b (9548)			0.00083	0.00083
DNA (1884)			0.00421	0.00421
gyrase (8)			0.98995	0.98995
subunit (1580)			0.00501	0.00501

Using the importance score formula in (3.1), we can easily build a matrix each cell of which stores the importance score of a token (row) to a concept (column) in the dictionary (i.e. UMLS in this paper). Then the concept name extraction is equivalent to tokenize sentences in a document and maximize the match between token sequences and concept names with a few syntactic constraints. The detailed extraction algorithm is presented in Figure 1. We treat a verb, preposition, punctuation and so on as the boundary of a concept name. If two or more concept candidates are found for an extracted concept name, we will further use surrounding tokens (3 to the left and 3 to the right) to narrow down the candidates in the same way as the extraction algorithm shown in Figure 1. The candidate with maximum importance score is chosen in the end unless only one candidate is remained.

Approximate Matching is a neat approach to the concept extraction. It completes phrase extraction and phrase meaning disambiguation within one step. More importantly, it achieves high precision as well as high recall. The evaluation of the extraction module on GENIA² 3.02 corpus achieved 56.32% *F-score* for exact match and 73.35% for approximate match, which are significantly better than approaches described in [16] and [18] (see table 2). We did not do formal evaluation for meaning disambiguation because no concept is annotated in GENIA corpus.

² <http://www-tsujii.is.s.u-tokyo.ac.jp/GENIA/>

```

Find next starting token  $t_s$ 
 $k = 0$ 
 $C = \{c \mid t_s \in T(c)\} / * T(c)$  is the set of tokens appearing in names of concept  $c$  */
For each  $c \in C$   $S_c = I(t_s, c) / * I(t_s, c)$  is the score of token  $t_s$  to concept  $c$  */
While next token  $t$  is not boundary token AND  $k < skip$ 
   $N = \{c \mid t \in T(c) \wedge c \in C\}$ 
  IF  $N = \emptyset$  Then  $k = k + 1$ 
  Else
     $C = N$ 
    For each  $c \in C$   $S_c = S_c + I(t, c)$ 
  End If
Wend
 $C = \{c \mid S_c > threshold \wedge c \in C\}$ 
If  $|C| > 0$  Then
  return concept name and candidate concepts  $c \in C$ 
End If

```

Fig. 1. The algorithm for extracting one concept name and its candidate concept IDs. The *threshold* is set to 0.95; the maximum number (*skip*) of skipped tokens is set to 1.

Table 2. The performance comparison of three different dictionary-based term extraction systems. Please read [16] for the detail of the evaluation method. BioAnnotator actually tested several configurations. But only the configuration with only dictionaries is compared.

IE Systems	Exact Match			Approximate Match		
	Recall	Precision	F-score	Recall	Precision	F-score
MaxMatcher	57.73	54.97	56.32	75.18	71.60	73.35
BioAnnotator [16]	20.27	44.58	27.87	39.75	87.67	54.70
PatternMatcher [18]	26.63	31.45	28.84	61.56	72.69	66.66

Given a document, MaxMatcher will extract a set of phrases and concepts. We will use both of them for indexing, called phrase-based indexing and concept-based indexing, respectively. One indexing example is presented below. The advantage of concept-based indexing over phrase-based indexing is twofold. First, it is able to solve synonym problem well because all synonymous terms share same concept IDs. Second, a concept stands for a unique meaning in a domain and thus, it will not cause any ambiguity.

Example

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

Phrase Index: *epidemiological study, obesity, periodontal disease*

Concept Index: *C0002783, C0028754, C0031090*

4 Experiments

We implement a basic unigram language model as described by formula 2.5 and 2.6. The coefficient λ in (2.6) is empirically set to 0.1 in our experiment. The dictionary

used for concept extraction is UMLS 2005AA version. With this prototyped IR system, biomedical literature can be indexed and searched either by concepts (*concept approach*) or by phrases (*baseline approach*).

The document collection we used for the experiment is from the TREC 2004 Genomic Track. The collection is a 10-year subset (1994-2003, 4.6 million documents) of the MEDLINE bibliographic database. However, human relevance judgments were merely made to a relative small pool. The pools were built from the top-precedence run from each of the 27 groups. They took the top 75 documents for each topic and eliminated the duplicates to create a single pool for each topic. The average pool size was 976, with a range of 476-1450 [3]. Our prototyped IR system only index and search all human relevance judged documents, i.e. the union of 50 single pools that contains total 42, 255 unique documents.

Following the convention of TREC, we take MAP (Mean Average Precision) as the primary measure for IR performance evaluation. MAP is a comprehensive indicator of IR performance that captures both precision and recall. P@10 (the precision of top 10 documents) and P@100 (the precision of top 100 documents) are treated as secondary measures in our evaluation.

Table 3. The comparison of our runs with official runs participated in TREC04 Genomics Track. Runs in TREC are ranked by Mean Average Precision (MAP) [3].

Run	MAP (%)	P@10	P@100
Concept Approach (Our Run)	36.94	59.80	44.76
Baseline Approach (Our Run)	29.17	49.53	40.82
pllsgen4a2 (the best)	40.75	60.04	41.96
uwntDg04tn (the second)	38.67	62.40	42.10
pllsgen4a1 (the third)	36.89	57.00	39.36
PDTNsmp4 (median)	20.74	40.56	23.18
edinauto5 (the worst)	0.12	0.36	1.3
Mean@TREC04 (47 runs)	21.72	42.69	26.37

The concept approach with a basic unigram language model achieves the 36.94% MAP, 59.80% P@10 and 44.76% while the baseline approach (phrase-based indexing and searching) achieves 29.17% MAP, 49.53% P@10 and 40.82% P@100, respectively. The paired-sample T test ($M=7.77\%$, $t=3.316$, $df=49$, $p=0.002$) shows the concept approach is significantly better than the baseline approach in terms of mean average precision. Thus, we can conclude that concept-based indexing and searching in conjunction with language model would significantly improve the performance of IR especially in a very specific domain such as biomedicine. This outcome, however, is slightly different from the result of many previous studies on sense-based IR which failed to show significant performance improvement. A possible explanation is that the concept extraction with an ontology in a very specific domain such as biomedicine would achieve much higher accuracy than word sense disambiguation in generic domains. Furthermore, the language models provide the chance to “smooth” the generative probability (or importance) of terms in a formal manner, which may allow the concept approach to fully take its potential.

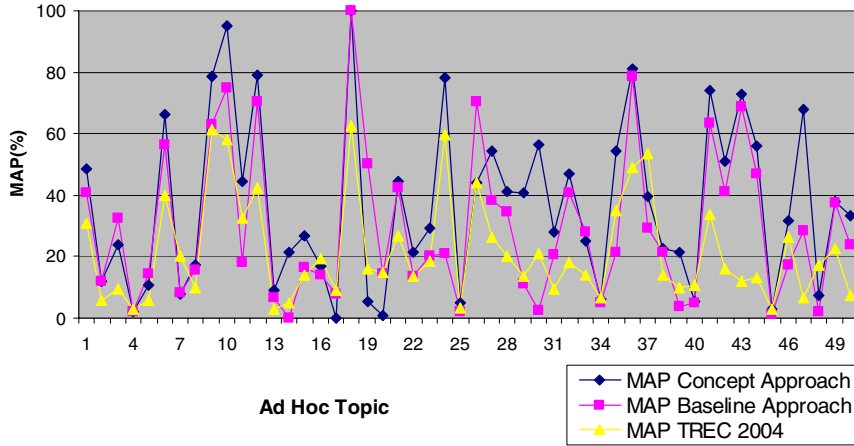


Fig. 2. The comparison of the MAP of our runs (Concept Approach and Baseline Approach) with the average MAP of official runs in TREC 2004 Genomic Track on 50 ad hoc topics

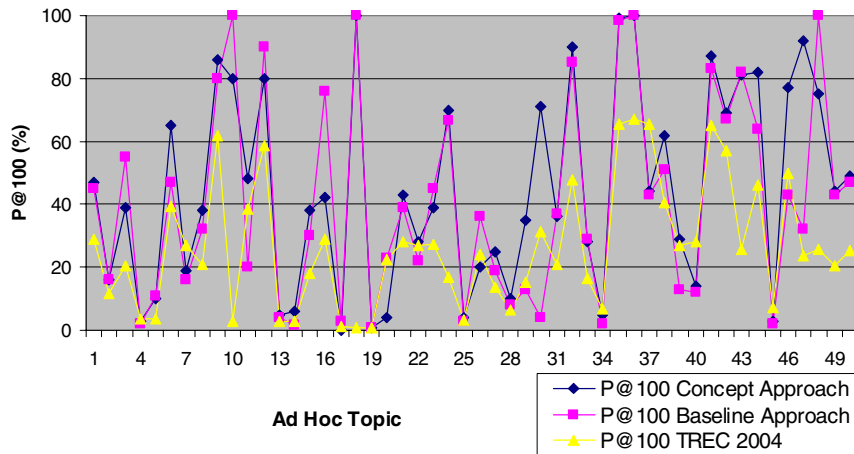


Fig. 3. The comparison of the P@100 of our runs (Concept Approach and Baseline Approach) with the average P@100 of official runs in TREC 2004 Genomic Track on 50 ad hoc topics

We further compare the sense approach with the official runs in TREC 2004 Genomic Track. Most runs in the track extensively apply various query expansion and pseudo-relevance feedback techniques to their IR models while our sense approach did nothing except for incorporating concept-based indexing into a basic unigram language model. Surprisingly, the performance of the sense approach is still much better than the average of the runs in the track and is comparable to the best run. The P@100 (44.76%) is even better than that of the best run. This outcome give us more reason to believe that semantic smoothing, i.e. the incorporation of synonym and sense information into the language models, is a more standard approach to achieving the effects the traditional query expansion and pseudo-relevance feedback techniques target.

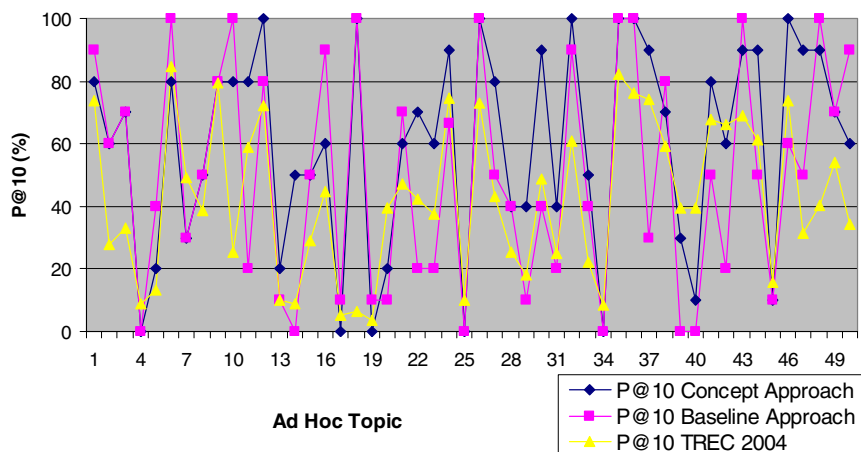


Fig. 4. The comparison of the P@10 of our runs (Concept Approach and Baseline Approach) with the average P@10 of official runs in TREC 2004 Genomic Track on 50 ad hoc topics

5 Conclusions and Future Work

For biomedical literature, synonyms of terms such as genes, proteins, cells and diseases are widely used while the polysemy of many terms and the use of partial names and abbreviations cause the ambiguity of terms. The synonym and polysemy has affected the performance of genomic IR. Unlike the emerging translation model and the traditional query expansion techniques, we address the issue of synonym and polysemy by incorporating concept-based indexing into a basic language model. In other words, we directly use concepts rather than phrases or individual words to index and search documents under the language modeling framework. It not only maintains the simplicity of language models, but also makes the ranking schema efficient and effective. The comparative experiment on the TREC 2004 Genomic Track data showed that the concept approach achieved significant performance improvement over the baseline approach. This outcome, however, is slightly different from the result of many previous studies on sense-based IR which failed to show significant performance improvement. A possible explanation is that the concept extraction with an ontology in a very specific domain such as biomedicine would achieve much higher accuracy than word sense disambiguation in generic domains. Furthermore, the language models provide the chance to “smooth” the generative probability (or importance) of terms in a formal manner, which may allow the concept approach to fully take its potential.

The performance of the concept model is also significantly superior to the average of official runs in TREC 2004 Genomic Track and is comparable to the performance of the best run. Because most official runs in the track extensively use various query expansion and pseudo-relevance feedback techniques while our approach does nothing except for the incorporation of concept-based indexing, we have more reasons to believe that semantic smoothing, i.e. the incorporation of synonym and sense information into the language models, is a more standard approach to achieving the effects the traditional query expansion and pseudo-relevance feedback techniques target.

For future work, we will continue to refine the method for concept extraction that we believe will affect the retrieval performance of the concept model. We will also test the generalization of our positive outcome by incorporating concept-based indexing into other retrieval models such as vector space model and other traditional probabilistic models. Last, we will take effort on other challenging issues of genomic IR as described in the introduction section.

References

1. Berger, A. and Lafferty, J.D., "Information Retrieval as Statistical Translation", *In proceedings of the 1999 ACM SIGIR Conference on Research and Development in Information Retrieval*, 1999, pp. 222-229.
2. Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., and Harshman, R., "Indexing by latent semantic analysis", *Journal of the American Society for Information Science*, 1990, 41(6), pp. 391-407.
3. Hersh W, et al. "TREC 2004 Genomics Track Overview", The thirteenth Text Retrieval Conference, 2004.
4. Lafferty, J. and Zhai, C., "Document language models, query models, and risk minimization for information retrieval", *2001 ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'01)*, 2001
5. Lafferty, J. and Zhai, C., "Probabilistic relevance models based on document and query generation", *In Language Modeling and Information Retrieval*, Kluwer International Series on Information Retrieval, Vol. 13, 2003.
6. Lesk, M., "Automatic Sense Disambiguation: How to Tell a Pine Cone from an Ice Cream Cone", *Proceedings of the SIGDOC'86 Conference, ACM*, 1986.
7. Mooney, R. J. and Bunescu, R. "Mining Knowledge from Text Using Information Extraction", *SIGKDD Explorations* (special issue on Text Mining and Natural Language Processing), 7, 1 (2005), pp. 3-10.
8. Palakal, M., Stephens, M.; Mukhopadhyay, S., Raje, R., Rhodes, S., "A multi-level text mining method to extract biological relationships", *Proceedings of the IEEE Computer Society Bioinformatics Conference (CBS2002)*, 14-16 Aug. 2002 Page(s):97 - 108
9. Ponte, J.M. and Croft, W.B., "A Language Modeling Approach to Information Retrieval", *Proceedings of the 21st annual international ACM SIGIR conference on Research and Development in Information Retrieval*.
10. Robertson, S. E. and K. Sparck Jones, "Relevance weighting of search terms", *Journal of the American Society for Information Science*, 1976, 27, 129--146.
11. Sanderson, M. 1994, "Word sense disambiguation and information retrieval", *Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*, p.142-151, July 03-06, 1994, Dublin, Ireland.
12. Soderland, S., Fisher, D., Aseltine, J., and Lehnert, W., "CRYSTAL: Inducing a Conceptual Dictionary", *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence*, 1995, pp. 1314-1319.
13. Soderland, S., "Learning Information Extraction rules for Semi-structured and free text", *Machine Learning*, Vol. 34, 1998, pp. 233-272.
14. Sparck Jones, K., Walker, S., and Robertson, S.E., "A probabilistic model of information retrieval: Development and comparative experiments Part I", *Information Processing and Management*, 2000, Vol. 36, pp. 779-808

15. Stokoe, C. and Tait, J. I. 2004. Towards a Sense Based Document Representation for Information Retrieval, in *Proceedings of the Twelfth Text REtrieval Conference (TREC)*, Gaithersburg M.D.
16. Subramaniam, L., Mukherjea, S., Kankar, P., Srivastava, B., Batra, V., Kamesam, P. and Kothari, R., "Information Extraction from Biomedical Literature: Methodology, Evaluation and an Application", *In the Proceedings of the ACM Conference on Information and Knowledge Management*, New Orleans, Louisiana, 2003.
17. Zhai, C. and Lafferty, J., "A study of smoothing methods for language models applied to information retrieval" , *ACM Transactions on Information Systems*, Vol. 2, No. 2, April 2004
18. Zhou, X., Han, H., Chankai, I., Prestrud, A., and Brooks, A., "Converting Semi-structured Clinical Medical Records into Information and Knowledge", *Proceeding of The International Workshop on Biomedical Data Engineering (BMDE) in conjunction with the 21st International Conference on Data Engineering (ICDE)*, Tokyo, Japan, April 5-8, 2005.