Inf Retrieval (2009) 12:51–68
DOI 10.1007/s10791-008-9075-7

An empirical study of gene synonym query expansion in biomedical information retrieval

Yue Lu · Hui Fang · Chengxiang Zhai

Received: 7 April 2008 / Accepted: 17 October 2008 / Published online: 12 November 2008

Abstract Due to the heavy use of gene synonyms in biomedical text, people have tried many query expansion techniques using synonyms in order to improve performance in biomedical information retrieval. However, mixed results have been reported. The main challenge is that it is not trivial to assign appropriate weights to the added gene synonyms in the expanded query; under-weighting of synonyms would not bring much benefit, while overweighting some unreliable synonyms can hurt performance significantly. So far, there has been no systematic evaluation of various synonym query expansion strategies for biomedical text. In this work, we propose two different strategies to extend a standard language modeling approach for gene synonym query expansion and conduct a systematic evaluation of these methods on all the available TREC biomedical text collections for ad hoc document retrieval. Our experiment results show that synonym expansion can significantly improve the retrieval accuracy. However, different query types require different synonym expansion methods, and appropriate weighting of gene names and synonym terms is critical for improving performance.

Keywords Biomedical information retrieval · Synonym query expansion · Language modeling

Y. Lu (✉) · C. Zhai
Department of Computer Science, University of Illinois at Urbana-Champaign, 201 N Goodwin Ave, Urbana, IL 61801, USA
e-mail: yuelu2@uiuc.edu

C. Zhai
e-mail: czhai@cs.uiuc.edu

H. Fang
Electrical and Computer Engineering, University of Delaware, 140 Evans Hall, Newark, DE 19716, USA
e-mail: hfang@ece.udel.edu

1 Introduction

The growing amount of scientific literature in genomics and related biomedical disciplines has led to an increasing need of using effective retrieval systems to access relevant information from biomedical literature. Many information needs in this domain center around genes, such as "the function of a gene" and "the interaction of two genes in some disease." However, a gene can be described with many variants, such as gene name, gene symbols and its acronyms. For example, gene "prion protein" can be described with "prnp", "Prn-p", "CD230", "PrPL-P1-like", "prion protein PrP", etc., in the biomedical literature. Unfortunately, most existing retrieval models rely on *exact* term matching, which makes it hard to retrieve relevant documents that contain the synonyms but not the one mentioned in the query. Thus, with the omnipresence of gene synonyms in biomedical literature (Buttcher et al. 2004), it is clear that we need to consider the synonyms in order to achieve optimal retrieval performance.

Due to its importance, this problem has attracted much attention recently in the TREC genomic track (Hersh 2003, 2004, 2005, 2006, 2007). Many groups have explored how to use synonym resources such as Entrez Gene[1] to improve retrieval accuracy. However, this body of previous work has mixed findings and experiment conditions are not controlled, making it impossible to compare different results. In particular, expanding queries directly with synonyms often leads to negative results (Goldberg et al. 2006; Divoli et al. 2006; Buttcher et al. 2004), while performance improvement is often achieved by manual synonym selection (Huang et al. 2006) or the use of special heuristics (Zhou et al. 2006). Thus, it is still unclear (1) whether automatic gene synonym expansion helps and (2) what is the best way to perform gene synonym expansion.

In general, synonym expansion is often achieved through query expansion. Unfortunately, the performance improvement of query expansion based on only hand-crafted thesaurus is often limited (Voorhees 1994; Stairmand 1997). The major challenge is how to assign appropriate weights to the synonyms.

In this paper, we conduct a systematic study of the gene synonym expansion problem in the language modeling framework. The language modeling framework provides a principled way to model retrieval problems and has also been shown to perform well empirically (Ponte and Bruce Croft 1998; Bruce croft and Lafferty 2003; Zhai and Lafferty 2004, 2006). We propose two methods for synonym expansion: single query language model (SQLM) and multiple query language models (MQLM). The idea of SQLM is similar to the traditional query expansion in the sense that synonyms are combined with the original query terms and documents are ranked based on the combined (i.e., expanded) query, but it provides flexibility to systematically adjust the weights of different aspects in the combined query. SQLM does not capture the disjunctive semantics of synonyms and the original genes, so in order to explicitly model the disjunctive relation among synonyms, we propose another method, i.e., MQLM, which combines results returned by using different gene variants to formulate queries including both gene information in the original query and other synonymous terms. We further propose and study several synonym weighting strategies. We perform a comprehensive evaluation of these methods on all available TREC Genomics data collections. Experiment results show that (1) Both SQLM and MQLM have the potential to significantly improve the retrieval performance (e.g., from 9.55% to 38.14% in MAP). (2) It is important to adjust weights on different aspects in verbose queries. (3) The proposed synonym weighting methods are more effective for

[1] http://www.ncbi.nlm.nih.gov/sites/entrez?db=gene.

gene-only queries than for verbose queries, though they can also increase recall without hurting MAP performance for verbose queries. (4) Applying pseudo relevance feedback after synonym expansion for verbose queries can further improve performance.

The rest of the paper is organized as follows. In Sect. 2, we survey the synonym expansion strategies explored in previous work on biomedical information retrieval. In Sect. 3, we briefly overview the language modeling approach for retrieval. We discuss two proposed synonym expansion methods in Sect. 4 and then evaluate their effectiveness over TREC Genomic collections in Sect. 5. Finally, we conclude in Sect. 6.

2 Related work

Query expansion is a commonly used strategy to improve retrieval performance. The main challenge of synonym expansion in both general and biomedical domains is how to assign appropriate weights to the synonyms. Recent studies (Fang and Zhai 2006; Fang 2008) used axiomatic approach to provide guidance on the weighting of expanded terms, and the results showed that the performance can be significantly improved with appropriate term weighting strategy. However, it is unclear whether their approach would work in biomedical domain.

An early study on biomedical information retrieval (Hersh et al. 2000) shows that retrieval performance does not improve from adding even manually selected synonyms. Recently, most work on biomedical information retrieval appeared in the Genomics Track of TREC 2003–2007 (Hersh 2003, 2004, 2005, 2006, 2007). To address the problem of synonymous terms in biomedical text, participating groups take different kinds of approaches. Most groups (Huang et al. 2007; Stokes et al. 2007; Cohen et al. 2007; Huang et al. 2006; Buttcher et al. 2004; Zhou et al. 2007; Ruiz 2006; Demner-Fushman et al. 2006; Lin et al. 2006; Dorff et al. 2006; Wan et al. 2006; Tsai et al. 2005; Abdou et al. 2005; Guo et al. 2004; Fujita 2004) automatically query gene synonyms from existing knowledge bases, such as Entrez Gene, MeSH, UMLS, AcroMed; some (Buttcher et al. 2004; Abdou et al. 2005; Huang et al. 2006) use heuristics to generate lexical variants for gene names; some (Stokes et al. 2007; Huang et al. 2006; Demner-Fushman et al. 2006) manually select appropriate synonyms from knowledge bases. After that, some groups (Jimeno and Pezik 2007; Buttcher et al. 2004; Guo et al. 2004) connect those synonyms with original gene name using disjunctive query semantics; others just add those synonyms to the set of original query terms. Most groups give synonyms same weights as gene names, while a few others (Guo et al. 2004; Buttcher et al. 2004) arbitrarily discount the weights of synonyms or boost the weights of gene names. And the results are mixed: some groups (Fautsch and Savoy 2007; Stokes et al. 2007; Cohen et al. 2007; Huang et al. 2006; Buttcher et al. 2004; Zhou et al. 2007; Ruiz 2006) report positive results of query expansion with synonyms, while others do not see significant improvement (Huang et al. 2007; Lin et al. 2006; Dorff et al. 2006; Wan et al. 2006; Tsai et al. 2005; Guo et al. 2004; Fujita 2004) or even report detrimental results (Huang et al. 2007; Jimeno and Pezik 2007; Demner-Fushman et al. 2006; Abdou et al. 2005). Since the previous work used different retrieval models, different synonym databases, and different heuristics, it is difficult to draw conclusions on the effectiveness of different methods. More importantly, none of them has studied how to automatically weight the added synonyms with respect to the original query and how to weight the synonyms among themselves.

Our work extends the previous work in two ways: (1) We propose general methods in the language modeling framework to perform synonym expansion; most strategies

explored in previous studies have more or less been covered by our general strategies. (2) We systematically compare and evaluate the major gene synonym expansion methods using all the existing TREC genomic collections.

3 Language models for information retrieval

KL-Divergence (Zhai and Lafferty 2001) is one of the most effective retrieval models derived in the language modeling framework. In the KL-Divergence retrieval model, queries and documents are all represented by unigram language models, which are essentially multinomial word distributions. Assuming that these language models can be appropriately estimated, KL-divergence retrieval model scores a document D with respect to a query Q by computing the Kullback-Leibler divergence between the query language model θ_Q and the document language model θ_D as follows:

$$-D_{KL}(\theta_Q||\theta_D) = -\sum_{w\in V} p(w|\theta_Q) log \frac{p(w|\theta_Q)}{p(w|\theta_D)}$$

where V is the set of all words in the vocabulary.

What remains to be solved is how to appropriately estimate the query language model θ_Q and the document language model θ_D. θ_D can be estimated from the document D with Dirichlet prior smoothing method (Zhai and Lafferty 2004) as shown below:

$$p(w|D) = \frac{c(w,D) + \mu \cdot p(w|\mathcal{C})}{|D| + \mu} \tag{1}$$

where $c(w,D)$ is the number of occurrences of word w in document D, $p(w|\mathcal{C})$ is the empirical word distribution in the whole collection C, and μ is a parameter that controls the degree of smoothing.

The simplest way to estimate the query language model θ_Q is to use maximum likelihood estimation which equals the empirical distribution:

$$p(w|\theta_Q) = p(w|Q) = \frac{c(w,Q)}{|Q|} \tag{2}$$

where $c(w,Q)$ is the number of occurrences of word w in query Q, and $|Q|$ is the length of query Q.

Clearly, the estimation of the query language model directly affects the retrieval performance. We will discuss a few methods to improve the query model estimation based on gene synonyms in the next section.

4 Gene synonym query expansion methods

We first introduce the notations that will be used in the paper. A query Q in biomedical domain often contains two aspects: gene aspect G and non-gene aspect NG, i.e., $Q = G \cup NG$. The gene aspect includes all query terms related to genes and is represented as $G = \{g_1, \ldots, g_n\}$. For each g_i, we can retrieve a set of synonyms S_i from the knowledge databases. The synonym set of the whole query is $S = \{S_1, \ldots, S_n\}$. Note that both S and G include information related to genes.

For example, assume we have query "what is the role of prnp in mad cow disease". The gene aspect is $G = \{\text{prnp}\}$, the non-gene aspect is $NG = \{\text{what}, \text{is}, \text{the}, \text{role}, \text{of}, \text{in}, \text{mad}, \text{cow}, \text{disease}\}$, and the synonym set would be $S = \{\{\text{Prnp}, \text{PrPLP1like}, \text{CD230}\}\}$.

Our main idea of performing gene synonym expansion in the KL-divergence retrieval model is to use the synonyms S to estimate a potentially more informative (effective) query language model θ_Q than the maximum likelihood estimate shown in Eq. 2. We now propose two general ways of constructing a query language model based on S.

4.1 Single query LM

The idea of single query language model (SQLM) is to combine the original query with synonym information and then estimate a language model from the combined information. Conceptually, this is to "expand" the original query with synonym terms, which is precisely the strategy adopted in most existing work. However, SQLM goes beyond existing work to offer a general probabilistic model that allows us to vary the weights of different components systematically as will be further discussed.

An expanded query logically includes two aspects: gene aspect $G \cup S$ and non-gene aspect NG, and there are two types of information in the gene aspect: gene information in the original query G and the synonym sets S. Thus, we would like to estimate the query language model using three sources of information: NG (non-gene aspect), G (gene description in the query), and S (synonyms):

$$p(w|\theta_Q) = p(w|Q, S) = p(w|G, NG, S)$$

One natural way to combine all the three sources of information is to define the query language model as the following mixture model, which gives us a linear combination of three unigram language models estimated using the three sources of information, respectively:

$$p(w|\theta_Q) = (1-\beta)p(w|NG) + \beta[(1-\alpha)p(w|G) + \alpha p(w|S)], \quad (3)$$

where $\beta \in [0, 1]$ is used to balance the gene aspect and non-gene aspect and $\alpha \in [0, 1]$ is the weight of balancing original gene information with its synonyms.

This mixture model can be interpreted as to capture the following process of sampling a word according to θ_Q: We first determine which part of the expanded query to use to generate a word, and then sample a word using the corresponding component language model to the part chosen. Specifically, with probability $1-\beta$, we would use the non-gene part (NG) and generate a word according to $p(w|NG)$. With probability β, we would use the gene part, but still need to decide whether to use G or S. With probability α, we would use S and generate a word according to $p(w|S)$, and with probability $1-\alpha$, we would use G and generate a word according to $p(w|G)$. Clearly, if we set $\alpha = 0$ and $\beta = \frac{|G|}{|G|+|NG|}$, we basically get the baseline as in Eq. 2 where no expansion of synonym is applied and maximum likelihood estimation is used to estimation the query model.

The estimated model shown in Eq. 3 can then be used directly in the KL-Divergence model to score documents in the collection, achieving the effect of query expansion based on synonyms. The estimation of the three component models will be discussed in Sect. 4.3.

4.2 Multiple query LMs

Although SQLM can naturally combine synonyms with the original query, it does not capture the desired disjunctive relationship between the original gene and its synonyms. As

a result, a document matching many synonym terms, but not the non-gene part may be scored higher than one matching one synonym term plus the non-gene part of the query. Intuitively, however, since the original gene and its synonyms represent exactly the same semantic aspect, matching any one of them would already imply matching the semantic aspect, thus matching multiple synonyms on top of that should not further contribute too much to the score.

To capture the disjunctive semantics in query expansion, we propose another way to incorporate synonyms into the KL-divergence retrieval model, called multiple query language models (MQLM). The main idea of MQLM is to generate multiple query models, corresponding to multiple ways of describing a gene in the query, and then combine the ranking lists from all these models in a certain way to generate the final ranking for the documents. Each query model can be regarded as modeling one query variant. A query variant Q_{ij} is generated by replacing an original gene query term g_i with one of its synonyms $s_{ij}(s_{ij} \in S_i)$. More formally,

$$Q_{ij} = \{Q \backslash g_i\} \bigcup \{s_{ij}\} \tag{4}$$

where $Q\backslash g_i$ means the set of query terms excluding g_i.

Now let $s(D;Q_{ij})$ be the relevance score of D w.r.t. query variant Q_{ij} and $s(D;Q)$ the relevance score of D w.r.t. the original query Q. We use $RS(D;Q,S)$ to denote the following set of adjusted relevance scores of document D w.r.t. the original query and all the query variants generated using S:

$$RS(D;Q,S) = \{(1-\alpha) \times s(D;Q)\} \cup (\bigcup_{s_{ij} \in S} \{\alpha \times \lambda_{ij} \times s(D;Q_{ij})\}),$$

where $\alpha \in [0,1]$ is a parameter to balance the trust on the original gene query terms and their synonyms, $\lambda_{ij} \in [0,1]$ measures the reliability of synonym $s_{ij} \in S$ and $p(w|Q_{ij})$ is estimated as

$$p(w|Q_{ij}) = (1-\beta)p(w|NG) + \beta p(w|s_{ij}) \tag{5}$$

where β balances the weight of gene aspect and that of the non-gene aspect.

In order to combine the results returned by these multiple query models, we propose to compute the relevance score of D as follows:

$$s(D;Q,S) = F(RS(D;Q,S)),$$

where F is an aggregation function that can combine a set of values. In the paper, we consider two types of F, i.e., *MAX* and *AVG*. *MAX* returns the maximal values in the set, while *AVG* returns the average. Intuitively, *MAX* better captures the disjunctive semantics while *AVG* is less aggressive than *MAX*.

Since the relevance scores computed with KL-Divergence function are negative and query dependent, we can not combine these scores directly in MQLM. In order to solve this problem, we use the normalized KL-Divergence results to score documents in the following way:

$$s(D;Q) = H(-D_{KL}(\theta_Q||\theta_D)),$$

where H is a function to transform the original relevance score to a positive value. We consider two transformation functions. The first is a simple exponential transformation function, i.e., $H(x) = exp(x)$. However, the results of this transformation are still not comparable because the scale of scores may vary from a query variant to another. We thus

further normalize the exponential transformation function into the range of [0,1] with a "min-max normalizer":

$$H(x) = \frac{exp(x) - a}{b - a},$$

where a and b are the minimal and maximal relevance score of the corresponding ranking list respectively. The normalized score would be a value between 0 and 1, thus becomes comparable across different ranking lists. Since the values of the second transformation function are more comparable, the results with this function are expected to be better, which has been confirmed in the experiments.

4.3 Parameter estimation

We use the maximum likelihood estimator to estimate $p(w|NG)$ and $p(w|G)$, which gives us normalized frequencies of words in NG and G, respectively:

$$p(w|NG) = \frac{c(w, NG)}{|NG|}$$

$$p(w|G) = \frac{c(w, G)}{|G|}$$

We now discuss how to estimate $p(w|S)$. Note that S is the synonym set that includes the synonyms of all gene aspects in the original query. Intuitively, all the synonyms in the set are not equally important for improving retrieval performance. Some of them might be more reliable than others. For example, a gene has different synonyms in different species. Given a query, appropriate synonym weighting might be able to give synonyms in the correct species higher weights. Assuming $S = \{s_{11}, \ldots, s_{nm}\}$, we propose the following general mixture model to estimate $p(w|S)$.

$$p(w|S) = \sum_{s_{ij} \in S} \lambda_{ij} p(w|s_{ij}) \; s.t. \; \sum_{s_{ij} \in S} \lambda_{ij} = 1$$

where $p(w|s_{ij})$ is estimated with MLE. Note that s_{ij} is a synonym, which may contain multiple words. A synonym with higher λ_{ij} means that this synonym is more reliable. If λ_{ij} is the same, all the synonyms in S receive equal weights.

SQLM is similar to the query expansion concept in the sense that expanded terms are combined with the original query. However, it is different from existing query expansion method in the way of query model estimation. None of the existing work has considered the three aspects we discussed above.

There are three types of parameters in both proposed methods: (1) α balances the weights of original gene term with its synonyms; (2) β controls the weights between gene aspect and non-gene aspect; and (3) λs are the weights for different synonyms.

Both α and β are set empirically, and the performance sensitivity for these parameters will be discussed in next section. We now focus our discussion on how to estimate the weight λ_{ij} for a synonym $s_{ij} \in S$.

In the previous work, people have tried to validate the synonyms by some heuristics, e.g. exclude synonyms with short length because they are more likely to be ambiguous, or avoid using synonyms that do not co-occur with the gene in the query frequently enough. Here, we propose a more general way to estimate the reliability of a synonym. Intuitively, a reliable synonym should be able to replace the gene information in the original query

without changing the meaning of the query. A good indicator of such "replaceability" is the similarity of their contexts; if two terms are interchangeable, the language used around them (we call it context) can be expected to be similar.

Formally, given a gene g_i in the original query, we define its context $C(g_i)$ as the language model learned from top K documents returned based on the original query Q. Given a synonym $s_{ij} \in S_i$ for g_i in query Q, its context $C(s_{ij})$ is defined as the language model learned from top K documents returned based on the query variant Q_{ij}. We introduce a "context similarity weighting strategy" by defining the weight λ_{ij} of synonym s_{ij} as the cosine similarity of the contexts of s_{ij} and g_i:

$$\lambda_{ij} = \text{cosine}(C(g_i), C(s_{ij})) \tag{6}$$

$$= \frac{\sum_{w \in V} p(w|C(g_i))p(w|C(s_{ij}))}{\sqrt{\sum_{w \in V} p(w|C(g_i))^2}\sqrt{\sum_{w \in V} p(w|C(s_{ij}))^2}} \tag{7}$$

However, a problem with the method above is that it might over-weight synonyms that frequently co-occur with the original gene term. One way to address this problem is to define a novelty context $C'(s_{ij})$ by excluding documents included in $C(g_i)$ from $C(s_{ij})$. This method is referred to as "novelty similarity weighting strategy". In this strategy, the weight λ_{ij} is the cosine similarity of the novel context of s_{ij} and the context of g_i:

$$\lambda_{ij} = \text{cosine}(C(g_i), C'(s_{ij})) \tag{8}$$

$$= \frac{\sum_{w \in V} p(w|C(g_i))p(w|C'(s_{ij}))}{\sqrt{\sum_{w \in V} p(w|C(g_i))^2}\sqrt{\sum_{w \in V} p(w|C'(s_{ij}))^2}} \tag{9}$$

Clearly, this strategy measures not only the context similarity of two terms, but also the novelty of the synonym, i.e. the ability of bringing in documents which original gene terms fail to retrieve.

5 Experiments

5.1 Experiment design

We construct three evaluation collections from all the available TREC Genomics Track collections (from 2003 to 2007). Since we only focus on the problem of gene synonym expansion, we filter out the topics that do not contain gene or protein names. The three collections are:

1. **G03**: 1-year medline abstract collection and 50 topics from TREC 2003;
2. **G0405**: 10-year medline abstract collection and 41 topics from TREC 2004 and 2005;
3. **G0607**: full text collection and 29 topics from TREC 2006 and 2007.

Table 1 shows the statistics of the three collections. The query type used in TREC 2003 is different from other collections. The 50 topics from TREC 2003 consist of only gene names and aim at finding all MEDLINE references that focus on the basic biology of the gene or its protein products from the designated organism. However, the topics from TREC 2004–2007 are often verbose in the sense that they contain non-gene terms and common

Table 1 Description of three evaluation collections

Collection	Documents	# Docs	Query type	# Queries
G03	Medline abstracts (April 2002–2003)	525,938	Gene-only	50
G0405	10 year Medline abstracts (1994–2003)	3,479,798	Verbose	41
G0607	Full text biomedical corpus	162,259	Verbose	29

English words in the query, such as "Find articles about Ferroportin-1, an iron transporter, in humans."

Since we are only interested in comparing different synonym expansion methods instead of optimizing the overall performance, preprocessing is minimized: we did not perform stemming or stop word removal; only some basic tokenization techniques have been applied, e.g. replacing hyphens with spaces. The gene names in the queries are manually labeled, and synonyms are looked up in NCBI Entrez Gene table for each gene names identified. The Dirichlet prior smoothing parameter μ in Eq. 1 is set to 1000, according to our earlier experiments. And we empirically use top $K = 100$ documents to estimate the context language model.

The results are evaluated with document mean average precision (MAP). MAP has so far been the standard measure used to evaluate ad hoc retrieval results and has also been used in the TREC Genomics Track evaluation (Hersh 2003, 2004, 2005, 2006, 2007). It has the advantage of being sensitive to the rank of every relevant document, thus it rseflects the overall ranking accuracy. We also report the results evaluated by P@30 (i.e, precision at top 30 documents), and R@1000 (Recall at top 1000 documents). Note that in TREC Genomics 2006 and 2007, the official tasks focus on passage retrieval, but we focus on document performance in this paper. Since the choice of retrieval unit is presumably orthogonal to the query expansion method, we may expect our conclusions to be applicable to passage retrieval as well. We thus leave further experiments with passage retrieval as a future work.

The specific goals of our experiments include: (1) evaluate the effectiveness of the two synonym expansion methods, i.e. SQLM and MQLM; (2) check whether synonym weighting helps improve performance and whether our weighting scheme is effective; (3) examine how sensitive the performance is to the settings of parameters, and try to provide some guidance of parameter setting; (4) recommend the most effective synonym expansion method for each different type of query; (5) compare and combine synonym weighting with pseudo relevance feedback, which is another commonly used strategy to improve retrieval performance.

5.2 Result analysis

5.2.1 Comparison of SQLM and MQLM

Table 2 shows the optimal performance of single query LM (SQLM) and multiple query LMs (MQLM) with different estimation strategies. The results are measured with MAP, precision at 30 (P@30) and recall at 1000 (R@1000). As discussed in Sect. 4.3, there are two parameters, i.e., α and β, that need to be set empirically. They are set based on the optimal performance measured by MAP in this table.

BL denotes the baseline method using MLE to estimate the query model without synonym expansion as shown in Eq. 2. **NoExp** denotes the proposed models discussed in

Table 2 Performance comparison of SQLM and MQLM

			BL	NoExp	UniformExp	NoveltyExp
G03	SQLM	Optimal parm		$\alpha = 0, \beta = 1$	$\alpha = 0.6, \beta = 1$	$\alpha = 0.7, \beta = 1$
		MAP	0.1193	0.1193	0.1562	**0.1648** (38.14%)*
		P@30	0.0653	0.0653	**0.0833**	0.08 (22.51%)*
		R@1000	0.6852	0.6852	0.8245	**0.8411** (22.75%)*
	MQLM	Optimal parm		$\alpha = 0, \beta = 1$	$\alpha = 0.3, \beta = 1$	$\alpha = 0.4, \beta = 1$
		MAP	0.1193	0.1193	0.1274	**0.1396** (17.02%)*
		P@30	0.0653	0.0653	0.0773	**0.0793** (21.44%)*
		R@1000	0.6852	0.6852	0.8008	**0.8266** (20.64%)*
G0405	SQLM	Optimal parm		$\alpha = 0, \beta = 0.2$	$\alpha = 0.4, \beta = 0.3$	$\alpha = 0.4, \beta = 0.3$
		MAP	0.2992	0.3653	0.3656	**0.367** (22.66%)*
		P@30	0.3732	**0.4398**	0.4358	0.4374 (17.20%)*
		R@1000	0.6372	0.6673	0.6813	**0.682** (7.03%)*
	MQLM	Optimal parm		$\alpha = 0, \beta = 0.2$	$\alpha = 0.2, \beta = 0.2$	$\alpha = 0.4, \beta = 0.2$
		MAP	0.2992	0.3653	0.3692	**0.3673** (22.76%)*
		P@30	0.3732	**0.4398**	0.4325	0.4293 (15.03%)*
		R@1000	0.6372	0.6673	0.6798	**0.6871** (7.83%)*
G0607	SQLM	Optimal parm		$\alpha = 0, \beta = 0.3$	$\alpha = 0.6, \beta = 0.5$	$\alpha = 0.5, \beta = 0.5$
		MAP	0.2755	0.2986	**0.3199**	0.3127 (13.50%)
		P@30	0.3264	**0.3609**	0.3494	0.3483 (6.71%)
		R@1000	**0.7288**	0.6791	0.718	0.6838 (−6.18%)*
	MQLM	Optimal parm		$\alpha = 0, \beta = 0.3$	$\alpha = 0.2, \beta = 0.3$	$\alpha = 0.4, \beta = 0.3$
		MAP	0.2755	0.2986	0.2997	**0.3018** (9.55%)
		P@30	0.3264	0.3609	0.3609	**0.3632** (11.27%)*
		R@1000	**0.7288**	0.6791	0.683	0.7015 (−3.75%)

Eqs. 5 and 3 with $\alpha = 0$, i.e., gene information in the original query is weighted and no synonym is used for expansion. **UniformExp** denotes the proposed models where λs are set to the same value. **NoveltyExp** denotes the proposed models where λs are set by using novelty similarity weighting strategy as shown in Eq. 9. The percentage of improvement of NoveltyExp over BL is also presented in parentheses. An asterisk indicates that the performance improvement is statistically significant at the 95% confidence level based on Wilcoxon test. Note that queries in G03 contain only gene aspect, so β is always 1 for G03.

The table brings us the following interesting messages:

1. Compared with the baseline, both of the proposed models (SQLM and MQLM) have the potential to improve the retrieval accuracy significantly.
2. The performance comparison between BL and NoExp shows that even before expansion with synonyms the performance can be improved by tuning β, the weight of gene aspect.
3. Comparing UniformExp and NoveltyExp with NoExp shows that gene synonym expansion is effective for improving retrieval performance on all collections.
4. Given appropriate gene aspect β value, weighting synonyms in the expansion helps short gene-only queries, but not the verbose queries.
5. SQLM outperforms MQLM on gene-only queries, but they perform comparably on verbose queries.

5.2.2 *Analysis of SQLM*

We now further analyze SQLM for different types of queries.

Gene-only queries: The previous subsection shows that SQLM outperforms MQLM on short gene-only queries. Here we further examine the details of SQLM expansion on G03 gene-only queries. With gene-only queries ($\beta = 1$), there is only one parameter α that controls the contribution of the synonyms in SQLM.

We examine three variations for estimating the values of λ: UniformExp, ContextExp and NoveltyExp. **ContextExp** and **NoveltyExp** denote that proposed models where λs are set by using context similarity and novelty similarity weighting strategies and discussed in Eqs. 7 and 8 respectively. We plot the performance of these three variations over different values of α as shown in Fig. 1. It shows that (1) Synonym expansion always improves performance compared with NoExp (i.e., $\alpha = 0$). (2) Both proposed synonym weighting strategies (i.e., ContextExp and NoveltyExp) are effective. NoveltyExp is better than ContextExp, and when optimized, it improves over NoExp by 38%. (3) NoveltyExp is robust in the sense that it improves over the baseline by 33% to 38% when α is set between 0.4 and 0.8.

Verbose queries: SQLM synonym expansion is more complicated in verbose queries than in gene-only queries, because both parameters, α and β, are involved. We plot MAP performance curves of SQLM expansion without synonym weighting for different β values on both G0405 and G0607 in Fig. 2. The x axis is the value of α, the degree that we trust synonyms.

Recall that in SQLM the query model is estimated using Eq. 3. It seems that performance is more sensitive to β than α, because the performance for curves mostly stay at the same level. The only exception is when β is small ($\beta = 0.2$) in which case the retrieval performance decreases quickly as α goes bigger, i.e. the gene aspect does not receive enough weight ($\beta(1 - \alpha)$ is too small). From another perspective, even if we do not expand queries with synonyms by setting $\alpha = 0$, we can still get near-optimal performance by setting an appropriate β value ([0.2, 0.4]). All these evidences point to the conclusion that it is very important to assign an adequate weight β to the gene aspect in order to achieve good performance. After achieving a strong baseline by choosing the optimal β, it is difficult to further improve performance by synonym expansion with SQLM. Further study is needed to understand why this is the case.

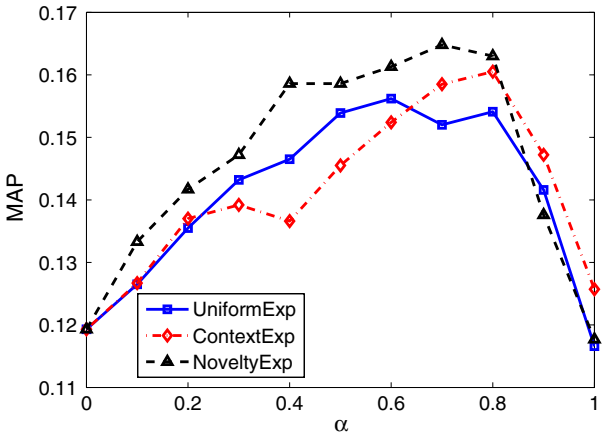

Fig. 1 SQLM expansion on G03 data

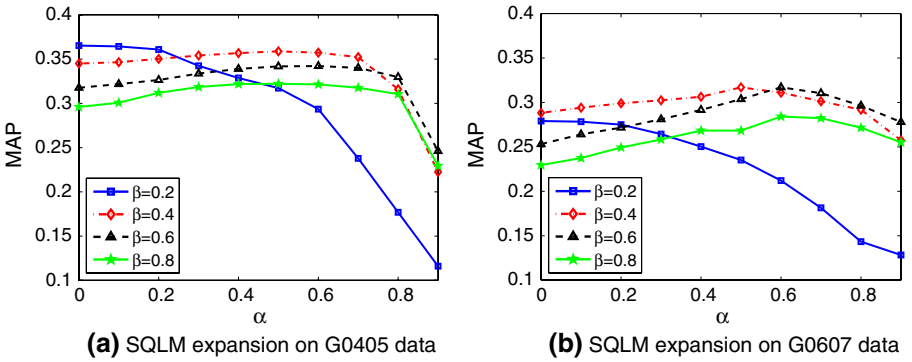

Fig. 2 SQLM analysis **a** SQLM expansion on G0405 data **b** SQLM expansion on G0607 data

Table 3 SQLM expansion on verbose queries

		NoExp	UniformExp	ContextExp	NoveltyExp	RecallExp	MapExp
G0405	Optimal α	$\alpha = 0$	$\alpha = 0.4$	$\alpha = 0.4$	$\alpha = 0.4$	$\alpha = 0.4$	$\alpha = 0.4$
	Optimal β	$\beta = 0.2$	$\beta = 0.3$	$\beta = 0.3$	$\beta = 0.3$	$\beta = 0.3$	$\beta = 0.3$
	MAP	0.3653	0.3656	0.3659	0.367	0.3729	**0.3773**
	P@30	0.4398	0.4358	0.4398	0.4374	0.4415	**0.4463**
	R@1000	0.6673	0.6813	0.6824	0.682	**0.7062**	0.6914
G0607	Optimal α	$\alpha = 0$	$\alpha = 0.6$	$\alpha = 0.6$	$\alpha = 0.6$	$\alpha = 0.4$	$\alpha = 0.7$
	Optimal β	$\beta = 0.3$	$\beta = 0.5$	$\beta = 0.5$	$\beta = 0.5$	$\beta = 0.4$	$\beta = 0.5$
	MAP	0.2986	0.3199	0.3061	0.3127	0.321	**0.3567**
	P@30	0.3609	0.3494	0.3483	0.3483	0.3563	**0.369**
	R@1000	0.6791	0.718	0.7098	0.7077	0.7145	**0.7884**

In order to further examine the effectiveness of synonym weighting, we report optimal performance for different weighting strategies of verbose queries in Table 3. It shows that our context similarity weighting (ContextExp) performs similarly to novelty similarity weighting scheme (NoveltyExp), but they do not help for verbose queries. However, there seem to be some differences between the two data sets. In order to deeply understand the differences between the two data sets, we perform some analysis experiments based on "gold standard weighting." In "gold standard weighting", each synonym s_{ij} is weighted based on the MAP or recall performance of its corresponding query variant Q_{ij} (Eq. 4). It is called "gold standard weighting", because the performance of Q_{ij} is evaluated on gold standard relevance judgement and in some sense is a good indicator of the best we could do in estimating the reliability of s_{ij}. We use **MapExp** and **RecallExp** as the abbreviation for gold standard weighting based on MAP and recall at 1000. They could be used to estimate the upper bound of synonym weighting on the given data set. As shown in Table 3, even gold standard MAP weighting could gain only 3.28% MAP improvement on G0405 data, which indicates that there is indeed little potential for improvement. On the contrary, there could be 19.46% increase in MAP by using gold standard MAP weighting in G0607 data. This clearly shows that our weighting scheme is not optimal on verbose queries.

5.2.3 Analysis of MQLMs

Synonym expansion with MQLM on gene-only queries is not as successful as with SQLM, so we will only discuss verbose queries in this subsection. As discussed in Sect. 4.2, there are multiple ways to combine the results of multiple query models, which depends on the choices of F and H, where F can be either exponential transformation or min-max normalization and H can be either *MAX* or *AVG*.

We now examine the effectiveness of these four variations. We fix β in Eq. 5 to the optimal value for the corresponding collection (i.e., $\beta = 0.2$ on G0405 and $\beta = 0.3$ on G0607). Since the performance decrease significantly if we trust synonyms more than the gene information in the query by setting $\alpha > 0.5$, we only report the result on $\alpha \in [0, 0.5]$. Figure 3 shows the performance sensitivity of all variants for the parameter α on G0405 and G0607 collections. We can clearly see that *MAX* with min-max normalization is the best combination among all, which means that it is important to capture the disjunctive semantics among synonyms by using *MAX* and to make the scores comparable by normalization.

We further check different weighting schemes using the best performing variation on verbose queries, i.e. minmax normalization + MAX, in Figs. 3, 4. It can be observed that although synonym weighting cannot improve MAP performance, it makes the MAP performance more robust, less sensitive to parameter settings. Furthermore, synonym expansion significantly increases recall performance and with our weighting schemes, the recall improvement is more robust in the sense that recall performance keeps going up even if we "over"-trust synonyms by setting a large α value. Note that although the uniform weighting achieves higher recall on G0607 data (Fig. 4) when $\alpha \in [0.35, 0.4]$, the corresponding MAP performance is falling below optimal when α is in such range (Fig. 4). So using our synonym weighting strategies is a good way to balance precision and recall and it saves the trouble of controlling the parameter value.

5.2.4 Pseudo relevance feedback

Pseudo feedback is another widely used strategy to expand queries with feedback information estimated from top K ranked documents. We conduct experiments to examine whether we can combine both synonym expansion and feedback methods to further improve performance. We use the model-based feedback method (Zhai and Lafferty 2001).

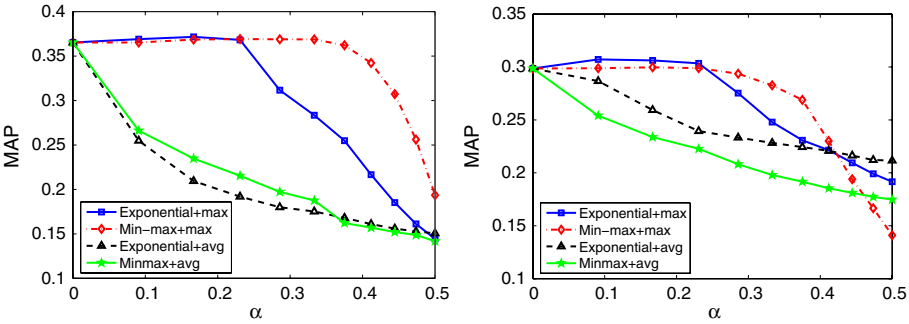

Fig. 3 Variations of MQLM **a** Variations of MQLM on G0405 **b** Variations of MQLM on G0607

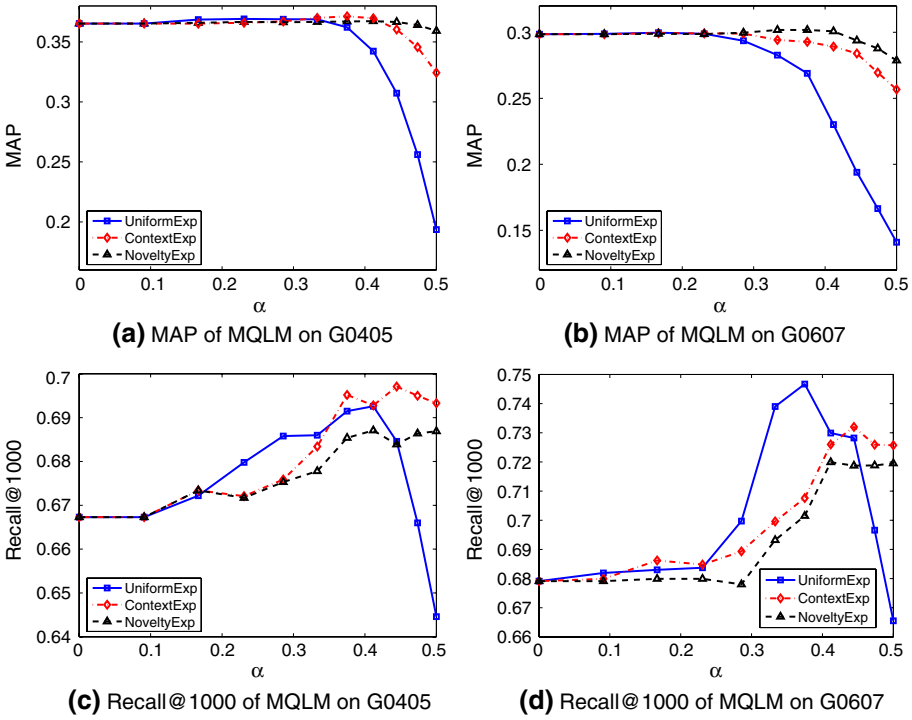

Fig. 4 MQLM analysis **a** MAP of MQLM on G0405 **b** MAP of MQLM on G0607 **c** Recall@1000 of MQLM on G0405 **d** Recall@1000 of MQLM on G0607

In Table 4, we report the evaluation results of (1) **FB on baseline**: directly applying pseudo relevance feedback on the no expansion baseline (2) **Syn-Exp**: one of our representative synonym expansion run (i.e., SQML + NoveltyExp) (3) **FB on Syn-Exp**: applying pseudo relevance feedback after our representative synonym expansion run. We use top 10 documents and a maximum of 50 terms in the pseudo relevance feedback estimation. We also mark the best performance in bold font for each measure in each data set.

Table 4 Comparison of synonym expansion with pseudo relevance feedback

	Run	MAP	P@30	R@1000
G03	FB on baseline	0.1211	0.0673	0.6936
	Syn-Exp	**0.1648**	0.0800	**0.8411**
	FB on Syn-Exp	0.1540	**0.0827**	0.8393
G0405	FB on baseline	0.3428	0.4106	0.6834
	Synonym Exp	0.3670	**0.4374**	0.6820
	FB on Syn-Exp	**0.3824**	0.4358	**0.7228**
G0607	FB on baseline	0.3144	**0.3540**	**0.7298**
	Synonym Exp	0.3127	0.3483	0.6838
	FB on Syn-Exp	**0.3231**	0.3483	0.6927

In the gene-only query data set (i.e., G03), the synonym expansion method significantly outperforms pseudo relevance feedback expansion in all measures. And further applying pseudo relevance feedback after synonym expansion does not help. Since there is only the gene aspect in the query, the results indicate that the terms brought in by pseudo feedback is not as reliable as our weighted synonyms.

In the verbose query data sets (i.e., G0405 and G0607), compared with pseudo relevance feedback, synonym expansion achieves higher MAP on G0405 data but similar MAP on G0607 data. However, on these two verbose query data sets, combining two expansion strategies always outperforms applying only one expansion method in terms of MAP performance. The intuition is that that synonym expansion helps on the gene aspect while pseudo relevance feedback further helps by bringing in terms related to the non-gene aspect.

But again, there is some difference between two data sets. The MAP improvement of the combined approach is significant on G0405 but not on G0607 according to Wilcoxon test. So we further look into the G0607 data set. Since we use the top 10 documents to do pseudo feedback; intuitively, if we apply pseudo feedback on a basic run that returns more relevant documents among the top 10 (i.e. achieves a higher precision@10), we expect to get better results. However, it is not the case. The average precision@10 for **Syn-Exp** is 0.4552 which is a significant improvement compared with 0.4069 for baseline, but **FB on Syn-Exp** is not significantly better than **FB on baseline**. Take topic 174 for example: applying pseudo feedback on Syn-Exp reduces MAP from 0.2079 to 0.1252 while applying pseudo feedback on baseline increases MAP from 0.1305 to 0.1880, even if **Syn-Exp** has a much higher precision@10, i.e. 0.4 compared with 0.1 of baseline. We do not have a good explanation for this observation. It is possible that some relevant documents contain distracting terms which when used to expand the original query could decrease the retrieval performance. Since G0607 is a full text data set, using the whole document for doing pseudo feedback presumably brings in more distracting terms than if we use a data set of abstracts such as G0405. But we need future experiments to test this hypothesis. For example, we may try to locate the relevant part in the relevant document and then use the local context terms to expand the query instead of using all the terms in the whole document.

5.2.5 Summary

From the comparison and analysis above, we can draw the following conclusions and recommendations. For short gene-only queries, using synonym expansion and novelty similarity weighting scheme can significantly improve performance based on all measures. For verbose queries, in order to achieve good MAP performance, assigning an appropriate weight for the gene aspect (i.e., $\beta = 0.2$ or 0.3) is of top importance. After achieving optimal performance by setting the correct β, it is difficult to improve MAP with synonym expansion. But recall can be improved with some price paid in precision. MQLM is more robust than SQLM on verbose queries, because it is less sensitive to the parameter value. Finally, applying pseudo relevance feedback after synonym expansion on verbose queries could further improve retrieval performance.

Our recommendation for short gene-only queries is to use SQLM expansion (Eq. 3) with novelty similarity weighting scheme where $\alpha \in [0.4, 0.8]$. For verbose queries, it is recommended that we first assign appropriate wights to the gene aspect ($\beta \in [0.2, 0.3]$) to achieve strong MAP performance, and then apply MQLM (with min-max normalization and *MAX* aggregation together with context similarity or novelty similarity weighting scheme) to further increase recall.

These recommendations, however, have to be taken cautiously with the understanding that they are based on limited experiments and optimal performance comparisons. A major limitation of our work is that we mostly compared optimal performance, which can show the potential of a method. In practice, we will have to set each parameter to a specific value. Thus an important future research direction is to further study how to set these parameters.

6 Conclusions and future work

The variations of gene names, symbols and acronyms in biomedical literature, along with the exact term matching characteristic of existing retrieval models, make it necessary to leverage synonyms to improve information retrieval performance. In this paper, we propose two principled methods for synonym expansion in the language modeling framework, i.e. single query language model (SQLM) and multiple query language models (MQLM). We also propose several synonym weighting strategies, and perform a systematic evaluation of these methods on all the available TREC Genomics data collections. Experiment results show that both SQLM and MQLM have the potential to significantly improve the retrieval performance. Our proposed synonym weighting scheme is effective for short gene-only queries on all measures and is able to increase recall without hurting precision for verbose queries.

There are many interesting future research directions worth exploring. First, in order to focus on understanding gene synonym query expansion, we intentionally controlled all other variables. It is known that the top performances in TREC were achieved through some additional heuristics other than gene synonym expansion, such as stemming. So one interesting future research direction would be to combine synonym expansion with other heuristics to see if synonym expansion can further improve performance on top of those other heuristics. In addition, our synonym weighting scheme cannot achieve optimal MAP performance on verbose queries. Further studies are needed to develop a more effective synonym weighting strategy for verbose queries. Another future direction is to test the current methods on other biological entities and even in the general domain. Finally, our evaluation is mostly based on optimal performance of the methods. In practice, we will need to set the parameters, so an important future research direction is to further study how to set the parameters.

Acknowledgments This material is based in part upon work supported by the National Science Foundation under award number 0425852 and work supported by NIH/NLM grant 1 R01 LM009153-01.

References

Abdou, S., Savoy, J., & Ruck, P. (2005). Evaluation of stemming, Query expansion and manual indexing approaches for the genomic task. In *Proceedings of TREC*.

Bruce croft, W., & Lafferty, J. (2003). *Language modeling and information retrieval*. Kluwer Academic Publishers.

Buttcher, S., Clarke, C. L. A., & Cormack, G. V. (2004). Domain-specific synonym expansion and validation for biomedical information retrieval (MultiText Experiments for TREC 2004). In *Proceedings of TREC*.

Cohen, A. M., Yang, J., Fisher, S., Roark, B., & Hersh, W. R. (2007). The OHSU biomedical question answering system framework. In *Proceedings of TREC*.

Demner-Fushman, D., Humphrey, S. M., Ide, N. C., Loane, R. F., Smith, L. H., Tanabe, L. K., Wilbur, W. J., Ruch, P., & Ruiz, M. E. (2006). Finding relevant passages in scientific articles: Fusion of automatic approaches vs. an interactive team effort. In *Proceedings of TREC*.

Divoli, A., Hearst, M. A., Nakov, P. I., & Schwartz, A. (2006). Biotext team report for the TREC 2006 Genomics Track. In *Proceedings of TREC*.

Dorff, K. C., Wood, M. J., & Campagne, F. (2006). Twease at TREC 2006: Breaking and fixing BM25 scoring with query expansion, a biologically inspired double mutant recovery experiment. In *Proceedings of TREC*.

Fang, H. (2008). A re-examination of query expansion using lexcial resources. In *ACL'08: Proceedings of the 46th Meetings of the Association for Computational Linguistics*.

Fang, H., & Zhai, C. (2006). Semantic term matching in axiomatic approaches to information retrieval. In *SIGIR '06: Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 115–122.

Fautsch, C., & Savoy, J. (2007). IR-specific searches at TREC 2007: genomics & blog experiments. In *Proceedings of TREC*.

Fujita, S. (2004). Revisiting again document length hypotheses TREC 2004 genomics track experiments at patolis. In *Proceedings of TREC*.

Goldberg, A. B., Andrzejewski, D., Van Gael, J., Settles, B., Zhu, X., & Craven, M. (2006). Ranking biomedical passages for relevance and diversity: University of Wisconsin, Madison at TREC Genomics 2006. In *Proceedings of TREC*.

Guo, Y., Harkema, H., & Gaizauskas, R. (2004). Sheffield University and the TREC 2004 genomics track: Query expansion using synonymous terms. In *Proceedings of TREC*.

Hersh, W. R., et al. (2003). TREC genomics track overview. In *Proceedings of TREC*.

Hersh, W. R., et al. (2004). TREC 2004 genomics track overview. In *Proceedings of TREC*.

Hersh, W. R., et al. (2005). TREC 2005 genomics track overview. In *Proceedings of TREC*.

Hersh, W. R., et al. (2006). TREC 2006 genomics track overview. In *Proceedings of TREC*.

Hersh, W. R., et al. (2007). TREC 2007 genomics track overview. In *Proceedings of TREC*.

Hersh, W. R., Price, S., & Donohoe, L. (2000). Assessing thesaurus-based query expansion using the UMLS Metathesaurus. In *Proceedings of the 2000 Annual AMIA Fall Symposium*.

Huang, X., Hu, B., & Rohian, H. (2006). York University at TREC (2006): Genomics track. In *Proceedings of TREC*.

Huang, X., Sotoudeh-Hosseinii, D., Rohian, H., & An, X. (2007). York University at TREC 2007: Genomics track. In *Proceedings of TREC*.

Jimeno, A., & Pezik, P. (2007). Information retrieval and information extraction in TREC genomics 2007. In *Proceedings of TREC*.

Lin, K. H.-Y., Hou, W.-J., & Chen, H.-H. (2006). NTU at TREC 2006 genomics track. In *Proceedings of TREC*.

Ponte, J. M., & Bruce Croft, W. (1998). A language modeling approach to information retrieval. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 275–281.

Ruiz, M. E. (2006). UB at TREC Genomics 2006: Using passage retrieval and pre-retrieval query expansion for genomics IR. In *Proceedings of TREC*.

Stairmand, M. A. (1997). Textual conext analysis for information retrieval. In *Proceedings of the 1997 ACM SIGIR Conference on Research and Development in I nformation Retrieval*.

Stokes, N., Li, Y., Cavedon, L., Huang, E., Rong, J., & Zobel, J. (2007). Entity-based relevance feedback for genomic list answer retrieval. In *Proceedings of TREC*.

Tsai, T.-H., Wu, C.-W., Hung, H.-C., Wang, Y.-C., He, D., Lin, Y.-F. , Lee, C.-W., Sung, T.-Y., & Hsu, W.-L. (2005). Enhance genomic IR with term variation and expansion: Experience of the IASL group at genomic track 2005. In *Proceedings of TREC*.

Voorhees, E. M. (1994). Query expansion using lexical-semantic relations. In *Proceedings of the 1994 ACM SIGIR Conference on Research and Development in Information Retrieval*.

Wan, R., Takigawa, I., Mamitsuka, H., & Ngoc Anh, V. (2006). Combining vector-space and word-based aspect models for passage retrieval. In *Proceedings of TREC*.

Zhai, C., & Lafferty, J. (2001). Model-based feedback in the language modeling approach to information retrieval. In *Proceedings of the 10th ACM International Conference on Information and Knowledge Management (CIKM'01)*, pp. 403–410.

Zhai, C., & Lafferty, J. (2004). A study of smoothing methods for language models applied to information retrieval. *ACM Transactions on Information Systems, 2*(2), 179–214.

Zhai, C., & Lafferty, J. (2006). A risk minimization framework for information retrieval. *Information Processing and Management, 42*(1), 31–55.

Zhou, W., Yu, C., Smalheiser, N., & Torvik, V., & Hong, J. (2007). Knowledge-intensive conceptual retrieval and passage extraction of biomedical literature. In *SIGIR '07: Proceedings of the 30th Annual International ACM SIGIR Conference On Research And Development In Information Retrieval*, pp. 655–662.

Zhou, W., Yu, C. T., Torvik, V. I., & Smalheiser, N. R. (2006). A concept-based framework for passage retrieval at genomics. In *Proceedings of TREC*.