A Bayesian Learning Approach to Promoting Diversity in Ranking for Biomedical Information Retrieval

Xiangji Huang
Information Retrieval and Knowledge
Management Research Lab
School of Information Technology
York University
Toronto, Canada
jhuang@yorku.ca

Qinmin Hu
Information Retrieval and Knowledge
Management Research Lab
Department of Computer Science & Engineering
York University
Toronto, Canada
vhu@cse.yorku.ca

ABSTRACT

In this paper, we propose a Bayesian learning approach to promoting diversity for information retrieval in biomedicine and a re-ranking model to improve retrieval performance in the biomedical domain. First, the re-ranking model computes the maximum posterior probability of the hidden property corresponding to each retrieved passage. Then it iteratively groups the passages into subsets according to their properties. Finally, these passages are re-ranked from the subsets as our output. There is no need for our proposed method to use any external biomedical resource. We evaluate our Bayesian learning approach by conducting extensive experiments on the TREC 2004-2007 Genomics data sets. The experimental results show the effectiveness of the proposed Bayesian learning approach for promoting diversity in ranking for biomedical information retrieval on four years TREC data sets.

Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**]: Text Mining

General Terms

Algorithm

Keywords

Bayesian Learning, Promoting Diversity, Biomedical IR

1. INTRODUCTION

Advances in computational and biological methods during the last decade have remarkably changed the scale of genomic research. Current genomic research is characterized by immense volume of data, accompanied by a tremendous increase in the number of genomics and biomedical related publications. This wealth of information has led to an increasing amount of interest and need for applying information retrieval techniques to access the scientific literature in genomics and related biomedical disciplines.

Information Retrieval (IR) in the context of biomedical databases has the following three major problems [2]: the frequent use of (possibly non-standardized) acronyms, the presence of homonyms (the same word referring to two or more different entities) and synonyms (two or more words referring to the same entity). How to deal with an abundant number of lexical variants of the same term is a challenging task in biomedical IR. These problems have raised many new data analysis and search challenges in the field of biomedical information retrieval, especially given that the genomic and biomedical literature is expanding in an unprecedented rate.

The Genomics Track of Text REtrieval Conference (TREC) provide a common platform to evaluate the methods and techniques proposed by various research groups for biomedical IR. The motivation of the TREC Genomics Track is to propose something in a system that attempts to provide short, specific answers to questions and put them in context by providing linking to original sources [7]. Systems are tasked with extracting out passages and grouped them into aspects identified by one or more MeSH terms. The performance is scored using mean average precision (MAP) at the document-level, passage-level, aspect-level and passage2-level [7]. The aspect-level is to address a question from different aspects, which indicates how comprehensive the question is answered. For example, the question "what is the role of gene PRNP in the Mad cow disease?" can be answered from aspects like "Diagnosis", "Neurologic manifestations", or "Prions/Genetics".

In this paper, we focus on promoting diversity in ranking using a re-ranking model. The central idea of the model is to compute the maximum probability of its hidden properties corresponding to each retrieved passage iteratively until all subsets achieve stability, and then re-rank these passages from different subsets. To the best of our knowledge, there is little work devoted to genomics aspect search for promoting diversity in ranking and a systematic comparison for biomedical information retrieval. In the rest of this paper, we use "aspect search" and "promoting diversity in ranking" exchangeable. In our work, we conducted an extensive and careful evaluation with different tuning constant values of our IR system. All the results show stable improvements in terms of the document-level, passage-level, aspect-level and passage2-level MAP over the baseline results on the TREC 2004-2007 Genomics data sets.

The remainder of this paper is organized as follows. We first give a brief survey in previous work in Section 2. In Section 3, a Bayesian learning approach is proposed. Then

SIGIR'09, July 19–23, 2009, Boston, Massachusetts, USA.

we describe the IR environment in Section 4. Following that, we present the experiments that we have conducted on the TREC Genomics data sets in Section 5 and 6. Finally, conclusions are given in Section 7.

2. RELATED WORK

A lot of work has been done on biomedical information retrieval in the past five years. In 2003 as the first year of the TREC Genomics Track, two tasks are featured: ad hoc retrieval and information extraction. Only the document-level performance is evaluated and a total of 25 groups have submitted 49 official runs for scoring with the highest document-level performance of 0.4165 [6]. The two tasks in the second year of the TREC Genomics Track are the standard ad hoc retrieval task and categorization of full-text documents. The document-level performance is also used. The best document-level performance is 0.4075 [6]. The tasks in 2005 are more detailed than in 2004. Retrieval performance is measured with Bpref (Binary PREFerence relations) [6]. The Bpref score for the topic is the average of the scores of its relevant documents. The score for each relevant document is the percentage of non-relevant documents (among the top R non-relevant) that it ranks better than. Bpref tracks closely to MAP (Mean Average Precision) with complete judgments, but degrades much more gracefully than MAP as judgments become more incomplete.

In the 2006 TREC Genomics Track, an emphasis is placed on returning relevant passages that discuss different aspects of the topic [7]. The participants' submissions are scored in three different ways. First, the passage-level retrieval performance is found by measuring the amount of overlap between returned passages and passages the judges deem relevant. Second, the aspect-level retrieval performance is scored by computing how diverse the set of passages returned is. Third, the document-level retrieval performance is calculated by essentially counting the number of relevant documents for which a passage is returned. In the 2007 TREC Genomics Track, an alternative passage MAP "Passage2" that calculated MAP as if each character is each passage were a ranked document is defined to compared the accuracy of the extracted answers [7]. Passage2 was used as the primary passage retrieval evaluation measure in 2007. Aspect retrieval was measured using the average precision for the aspects of a topic averaged across all topics in 2007.

However, there is not too much previous work conducted on Genomics aspect search with promoting diversity in ranking. In the 2006 TREC Genomics Track, University of Wisconsin re-ranked the passages using a naive clustering-based approach called GRASSHOPPER to promote diversity [5]. Existing methods to improve diversity in ranking include maximum marginal relevance (MMR) [3], mixture models [17], subtopic diversity [16] and diversity penalty [18]. The basic idea is to penalize redundancy by lowering an item's rank if it is similar to items already ranked. GRASSHOPPER is an alternative to MMR and variants with a principled mathematical model and strong empirical performance on artificial data set [21]. Unfortunately, this re-ranking method actually hurt promoting diversity and its aspect-level performance was not as good as the original results [5].

Later in the 2007 TREC Genomics Track, most teams tried to obtain the aspect-level performance through their passage2-level results, instead of working on the aspect-level search directly. For example, National Library of Medicine (NLM) combined resources to find passages containing answers to biomedical questions [4]. University of Illinois at Chicago interpreted queries into two types for passage extraction and they kept the same passage ranking list for aspect search [20].

3. A BAYESIAN LEARNING APPROACH

Bayesian learning is a learning process based on Bayes rule which is used to update the prior distribution of the parameters of the learning model and compute the posterior distribution for prediction purpose. In this section, we will first introduce our model in Section 3.1 which iteratively calculates the hidden properties for the retrieved passages and re-ranks the passages into a new list. Then the pseudo codes for an algorithm are presented in Section 3.2.

3.1 A Re-ranking Model

A re-ranking model is proposed to calculate the hidden properties of the retrieved passages. In the model, we build up a probability space $(\Omega^\star, \mathcal{F}^\star, \mathrm{P})$ for the probability calculations. Then retrieved passages are iteratively grouped into different aspect subsets according to their hidden aspect estimations. Finally a re-ranking list is generated from different aspect subsets. Here each aspect subset corresponds to a group of retrieved passages that contain the common hidden property.

A probability space $(\Omega, \mathcal{F}, \mathrm{P})$ is proposed as our whole measure space [13]. The sample space Ω is the genomics data set presented as the index, which is introduced in Section 4.2. The events set $\mathcal{F}$ is a $\sigma - algebra$ of subsets of Ω, whose elements are the keyword sequences of the topics. The probability measure P is a function from $\mathcal{F}$. The targets of the re-ranking model are N retrieved passages outputted by the IR system, but not the whole data set. Therefore as a subset of Ω, the subspace $\Omega^\star$ is generated by the N retrieved passages. $\mathcal{F}^\star$ is a $\sigma - algebra$ of subsets of $\Omega^\star$. Then in this model, all calculations are under the probability space $(\Omega^\star, \mathcal{F}^\star, \mathrm{P})$.

For each topic, the system outputs N retrieved passages. Each passage can be expressed by $\mathbf{x_i}$ as a vector and the retrieved results for each topic can be presented as a matrix $\mathbf{X}$. Topics and the retrieved passages are represented as the keyword sequences. As follows $\mathbf{t}$ shows a topic where w stands for a keyword.

$$\mathbf{t} = \{w_1, w_2, ..., w_m\} \tag{1}$$

$$\mathbf{X} = \{\mathbf{x_1}, \mathbf{x_2}, ..., \mathbf{x_N}\}^T \tag{2}$$

$$\mathbf{x_i} = \{x_{i1}, x_{i2}, ..., x_{in}\} \tag{3}$$

Θ is a set of properties with $\Theta = \{\theta_1, \theta_2, ..., \theta_k\}$, where each θ_j $(j = 1, 2, ..., k)$ is a hidden property of some retrieved passages and each θ_j is independent. It is reasonable to assume that Θ has a Poisson distribution such that the expected parameter of passages' occurrences is θ_j. Then the prior density is shown in Equation 4, where n is the number of passages' occurrences. As the input, the probability of $\mathbf{x_i}$ is known based on the original results. In Equation 5, the likelihood function is presented by $P(\mathbf{x_i}|\theta)$. The posterior probability is presented by Equation 6 which can be interpreted by $l(\theta_j|\mathbf{x_i})$ and $P(\theta_j)$ because $P(\mathbf{x_i})$ is known.

$$P(\theta_j) = f(n, \theta_j) = \frac{\theta_j^n e^{-\theta_j}}{n!} \tag{4}$$

$$l(\theta_j|\mathbf{x_i}) = P(\mathbf{x_i}|\theta_j) \tag{5}$$

$$\begin{aligned} P(\theta_j|\mathbf{x_i}) &= \frac{P(\mathbf{x_i}|\theta_j)P(\theta_j)}{P(\mathbf{x_i})} \\ &\propto P(\mathbf{x_i}|\theta_j)P(\theta_j) \\ &\propto l(\theta_j|\mathbf{x_i})P(\theta_j) \end{aligned} \tag{6}$$

The maximum probability of property θ_j^1 for the 1st iteration is estimated by Equation 7 according to the posterior probabilities of θ_j given $\mathbf{x_i}$.

$$\begin{aligned} \theta_j^1 &= \arg\max_{\theta_j \in \Theta}\{P(\theta_j|\mathbf{x_i})\} \\ &\propto \arg\max_{\theta_j \in \Theta}\{l(\theta_j|\mathbf{x_i})P(\theta_j)\} \end{aligned} \tag{7}$$

```
0. Input
     The original results for the topics on each data set;
     Keywords for each retrieved passages.

1. Output
     New re-ranking results for the topics on each data set.

2. Initialize
     $k = N = 1000$;
     $l(\theta_j|\mathbf{x_i}) = P(\mathbf{x_i}|\theta_j) = logP(\mathbf{x_i})$;
     Set $\theta_i = \theta_j$ through comparing keywords of each passages
     and group the $n$ passages to compute the prior probabilities;
     $\varepsilon = 0.001$;

3. Iterate
     for each topic {
        Compute the posterior probabilities in Equation 6 until
          convergence;
        Calculate $\theta_j^l$ in Equation 7;
        Group $\theta_j^l$ and update $n$;
        if($P(\theta_j^{p+1}|\mathbf{x_i}) - P(\theta_j^p|\mathbf{x_i}) < \varepsilon$)
           exit;          }

4. Re-ranking
     //After p iterations, k aspect subsets are grouped from 3.
     Rank aspect subsets {
        List aspect subsets $\Theta^p = \{\theta_1^p, \theta_2^p, ..., \theta_k^p\}$;
        Compute $v_j$ for $\theta_j^p$ in Equation 9;
        Sort $v_j$;
        Rank aspect subset $\Theta^{p+1} = \{\theta_1^{p+1}, \theta_2^{p+1}, ..., \theta_k^{p+1}\}$
          according to $v_j$;
     }
     Re-rank passages {
        for (j=1; j < k; j++) {
           Select the top passage from aspect subset $\theta_j^{p+1}$ as the $j^{th}$
             passage in the new list;
           Remove this passages from $\theta_j^{p+1}$;
           Until $\theta_j^{p+1}$ is empty;     }
      }
```

Figure 1: A Re-ranking Algorithm

Initially we set the aspect subset number $k = N$ so that each retrieved passage corresponds to one of these N subsets. After the computation of Equation 7, a new k value is generated and each passage is assigned to the subset which has the same estimation parameter θ_j^1. This computation process is repeated until there are no passages moving among the subsets. As follows are the final property set and passages $\{x_{j1}, x_{j2}, ..., x_{jq}\}$ in each subset θ_j^p after p iterations.

$$\Theta^p = \{\theta_1^p, \theta_2^p, ..., \theta_k^p\} \tag{8}$$

$$v_j = \frac{\sum_{i=1}^{q} w(x_{ji})}{q} \tag{9}$$

$$\Theta^{p+1} = \{\theta_1^{p+1}, \theta_2^{p+1}, ..., \theta_k^{p+1}\} \tag{10}$$

3.2 A Re-ranking Algorithm

For re-ranking results, Θ^{p+1} is generated by sorting θ_j^p according to its value v_j. We choose the top passage in θ_1^{p+1} as the first one in a new list then remove it from the θ_1^{p+1} subset. Similarly we select the top one from θ_2^{p+1} as the second one in the new list and remove it from this subset. Every time we select and remove the top passage from each subset and rank them in the new list. This re-ranking process is repeated until all the passages are chosen from the above k subsets.

Our algorithm is presented in Figure 1. There are three major phases in this iterative algorithm. For the first phase, we initialize the settings for the algorithm. The initial k value is set to be N, different n values are computed for the prior probabilities, and ε is set for convergence. For the second phase, an iterative process is described to compute and update θ_j^p after p iterations. Here the prior probabilities, posterior probabilities, the subsets are updated in each iteration until the exit condition is satisfied. For the third phase, we present how to do re-ranking. In this phase, the property subsets are sorted by their weights v_j in Equation 9. $w(x_{ji})$ of passage x_{ji} is refined by a result combination model which is introduced in our paper [8]. Then passages are selected and removed from the subsets in turns and formed to be a new list as the output.

4. IR ENVIRONMENT

4.1 The System

We used Okapi BSS (Basic Search System) [15] as our main search system and conducted our information retrieval experiments using the improved Okapi system [9, 10, 11, 12, 19]. Okapi is an information retrieval system based on the probability model of Robertson and Sparck Jones [1, 15]. The retrieval documents are ranked in the order of their probabilities of relevance to the query. Search term is assigned weight based on its within-document term frequency and query term frequency. The weighting function used is BM25.

$$\begin{aligned} w = &\frac{(k_1+1)*tf}{K+tf} * \log\frac{(r+0.5)/(R-r+0.5)}{(n-r+0.5)/(N-n-R+r+0.5)} \\ &* \frac{(k_3+1)*qtf}{k_3+qtf} \quad \oplus \quad k_2 * nq * \frac{(avdl-dl)}{(avdl+dl)} \end{aligned} \tag{11}$$

where N is the number of indexed documents in the collection, n is the number of documents containing a specific term, R is the number of documents known to be relevant to a specific topic, r is the number of relevant documents containing the term, tf is within-document term frequency, qtf is within-query term frequency, dl is the length of the document, $avdl$ is the average document length, nq is the number of query terms, the k_is are tuning constants (which depend on the database and possibly on the nature of the queries and are empirically determined), K equals to $k_1 * ((1-b) + b * dl/avdl)$, and $\oplus$ indicates that its following component is added only once per document, rather than for each term.

In our experiments, the tuning constants k_1 and b are set to be different values. k_2 and k_3 are set to be 0 and 8 respectively.

4.2 Indexing

One important issue that IR systems have to deal with is the size of the retrieved passages and the granularity of the indexed information. In the context of text retrieval, the granularity of the indexed text can be defined as the length of the indexed text unit and the size can be defined as the length of the retrieved passage. In this paper, we call an indexed text unit as a *passage*.

Three indices are built on the genomics 2007 and 2006 data sets according to three passage extraction methods [8] and a paragraph-based index is built on the genomics 2005 and 2004 data sets. Sentence-based indexing is based on passages each of which has up to 3 sentences. Paragraph-based indexing is generated on passages each of which is a paragraph. Here a paragraph is defined as the sequence of sentences between the <p> and </p> tags from the HTML data set. Word-based indexing forms passages that contain one, two or three sentences each, where the number of words in the passages may only slightly exceed 47.

4.3 Data Sets and Evaluation Measures

We evaluated our models on four TREC data sets: the Genomics 2007 and 2006 data sets with 36 topics in 2007 and 28 topics in 2006, the Genomics 2004 and 2005 data sets with 50 topics respectively.

TREC 2007 and 2006 Genomics data sets provide a test collection of 162,259 full-text documents. The TREC 2007 queries are in the form of questions asking for lists of specific entities. The definitions for these entity types are based on controlled terminologies from different sources, with the source of the terms depending on the entity type [7]. The TREC 2006 queries are derived from the set of biologically relevant questions based on the Generic Topic Types (GTTs) [6, 7]. All these queries are listed on the official genomics website at: http://ir.ohsu.edu/genomics.

TREC 2005 and 2004 Genomics data sets consist of a document collection for the ad hoc retrieval task which is a 10-year subset of MEDLINE with completed citations from the database inclusive from 1994 to 2003. This provides a total of 4,591,008 records [6]. Each record is an abstract of a document. In this paper, we take the abstract as a paragraph-based passage. There are 50 queries for each year respectively. More information can be found at: http://www.ncbi.nlm.nih.gov/entrez/query/static/help/pmhelp.html#MEDLINEDisplayFormat

Evaluation Measures in terms of the document-level, passage-level, aspect-level and passage2-level are presented in this paper. Each of these provides insight into the overall performance for a user trying to answer the given queries and measured by some variant of mean average precision (MAP). Their definitions can be found in [6, 7].

5. EXPERIMENTAL RESULTS

In this section, we describe a series of experiments that have been conducted to evaluate the effectiveness of the proposed model on the TREC 2004-2007 genomics data sets. First, we present the original results on word-based, sentence-based and paragraph-based indices under five parameter settings (k_1,b). Second, the re-ranking results and the relative improvements are shown corresponding to the original results.

5.1 Original Performance

Table 1 shows the performance for five parameter settings with three different indices in terms of the document-level, passage-level, aspect-level and passage2-level on the genomics 2004-2007 data sets respectively. The first and second columns are the parameters settings. The third one is for the indices and the rest columns are the evaluation measure for the 2004-2007 data sets.

5.2 Re-ranking Performance

Corresponding to the original results, we generate the improved re-ranking results using our proposed algorithm in Figure 1. The performance and improvements are presented in Table 2. The values in the parentheses are the relative rates of improvement over the original results.

6. DISCUSSION AND ANALYSIS

In this section, we first investigate the influence of using different indices. Then, we investigate the influence of different tuning constant values. Furthermore, the number k_r of aspect subsets generated by the proposed re-ranking model is discussed. Finally, we make a comparison and analysis with the K-mean algorithm in terms of the aspect-level performance.

6.1 Influence of Different Indices

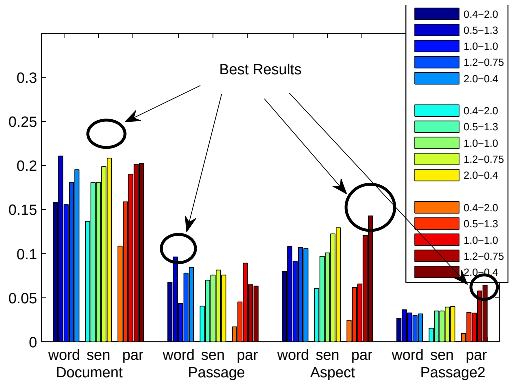

Figure 2: Performance of Baselines 2007

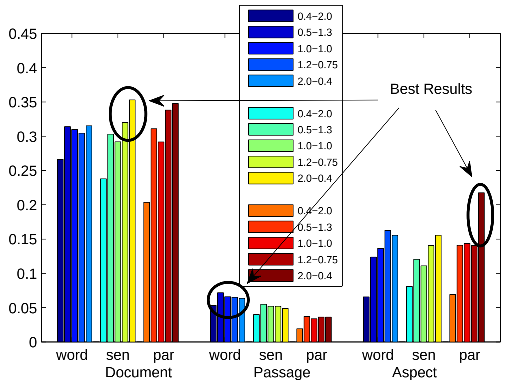

Figure 3: Performance of Baselines 2006

In order to investigate the influence of different indices, we will continue to analyze the experimental results presented in Section 5.1. To illustrate the results in Table 1 graphically, we re-plot these data in Figure 2, 3 and 4. The performance of the original results is shown in terms of the document-level, passage-level, aspect-level and passage2-level. The x-axis represents the evaluation measures, where word, sen and par stand for word-based, sentence-based and paragraph-based indices. The y-axis shows the MAP performance.

Table 1: Performance of Original Baselines

k_1	b	Indices	Baseline 2007				Baseline 2006			Baseline 2005	Baseline 2004
			document	passage	aspect	passage2	document	passage	aspect	document	document
0.4	2.0	word	0.1584	0.0675	0.0801	0.0267	0.2662	0.0532	0.0657	-	-
		sentence	0.1368	0.0406	0.0605	0.0154	0.2378	0.0398	0.0808	-	-
		paragraph	0.1086	0.0170	0.0244	0.0094	0.2036	0.0192	0.0690	0.1964	0.2952
		AVERAGE	0.1346	0.0417	0.0550	0.0172	0.2359	0.0374	0.0718	0.1964	0.2952
0.5	1.3	word	**0.2108**	**0.0963**	0.1080	0.0364	0.3140	**0.0718**	0.1237	-	-
		sentence	0.1805	0.0700	0.0970	0.0350	0.3030	0.0550	0.1206	-	-
		paragraph	0.1588	0.0452	0.0616	0.0333	0.3109	0.0369	0.1410	0.2602	0.3404
		AVERAGE	0.1834	0.0705	0.0889	0.0349	0.3093	0.0546	0.1284	0.2602	0.3404
1.0	1.0	word	0.1556	0.0434	0.0916	0.0328	0.3097	0.0659	0.1365	-	-
		sentence	0.1809	0.0758	0.1009	0.0350	0.2918	0.0521	0.1110	-	-
		paragraph	0.1902	0.0893	0.0656	0.0327	0.2916	0.0337	0.1438	0.2547	0.3425
		AVERAGE	0.1756	0.0695	0.0860	0.0335	0.2977	0.0506	0.1304	0.2547	0.3425
1.2	0.75	word	0.1809	0.0780	0.1070	0.0295	0.3045	0.0651	0.1626	-	-
		sentence	0.1987	0.0814	0.1225	0.0394	0.3202	0.0522	0.1405	-	-
		paragraph	0.2013	0.0648	0.1208	0.0578	0.3381	0.0362	0.1407	**0.2874**	**0.3584**
		AVERAGE	0.1936	0.0747	0.1168	0.0423	0.3209	0.0512	0.1479	0.2874	0.3584
2.0	0.4	word	0.1953	0.0844	0.1057	0.0317	0.3152	0.0637	0.1556	-	-
		sentence	0.2084	0.0758	0.1294	0.0401	**0.3529**	0.0490	0.1557	-	-
		paragraph	0.2025	0.0633	**0.1428**	**0.0641**	0.3476	0.0362	**0.2176**	0.2779	0.3483
		AVERAGE	0.2021	0.0745	0.1260	0.0453	0.3386	0.0496	0.1763	0.2779	0.3483

Table 2: Performance of The Bayesian Approach

k_1	b	Indices	Improved 2007				Improved 2006			Improved 2005	Improved 2004
			document	passage	aspect	passage2	document	passage	aspect	document	document
0.4	2.0	word	0.2575 (62.56%)	0.1620 (140.00%)	0.2261 (182.27%)	0.0657 (146.07%)	0.3100 (16.45%)	0.0957 (79.89%)	0.1258 (91.48%)	-	-
		sentence	0.2138 (56.29%)	0.1194 (194.09%)	0.1740 (187.60%)	0.0412 (167.53%)	0.2752 (15.73%)	0.0873 (119.35%)	0.1162 (43.81%)	-	-
		paragraph	0.1887 (73.76%)	0.0907 (433.53%)	0.1265 (418.44%)	0.0419 (345.74%)	0.2401 (17.93%)	0.0456 (137.50%)	0.1174 (70.14%)	0.2443 (24.39%)	0.3022 (2.37%)
0.5	1.3	word	0.3460 (64.14%)	0.2014 (109.14%)	0.2331 (115.83%)	0.0813 (123.35%)	0.3828 (21.91%)	**0.1262** (75.77%)	0.2370 (91.59%)	-	-
		sentence	**0.3280** (81.72%)	**0.1740** (148.57%)	0.2629 (171.03%)	0.0774 (121.14%)	0.3751 (23.80%)	0.1111 (102.00%)	0.1981 (64.26%)	-	-
		paragraph	0.2442 (53.78%)	0.1300 (187.61%)	0.1727 (180.36%)	0.0797 (139.34%)	0.3633 (16.85%)	0.0733 (98.64%)	0.2122 (50.50%)	0.3048 (17.14%)	0.3484 (2.35%)
1.0	1.0	word	0.2981 (91.58%)	0.1782 (310.60%)	0.2515 (174.56%)	0.0657 (100.30%)	0.3808 (22.96%)	0.1218 (84.83%)	0.2462 (80.37%)	-	-
		sentence	0.2814 (55.56%)	0.1681 (121.77%)	0.2425 (140.34%)	0.0688 (96.57%)	0.3558 (21.93%)	0.1053 (102.11%)	0.1954 (76.04%)	-	-
		paragraph	0.2319 (21.92%)	0.1211 (35.61%)	0.1639 (149.85%)	0.0705 (115.60%)	0.3410 (16.94%)	0.0662 (96.44%)	0.2143 (49.03%)	0.2788 (9.46%)	0.3506 (2.36%)
1.2	0.75	word	0.3086 (70.59%)	0.1726 (121.28%)	0.2567 (139.91%)	0.0684 (130.30%)	0.3579 (17.54%)	0.1101 (69.12%)	0.2279 (40.16%)	-	-
		sentence	0.3372 (69.70%)	0.1854 (127.76%)	0.3241 (164.57%)	0.0834 (111.68%)	0.3841 (19.96%)	0.1029 (97.13%)	0.2618 (86.33%)	-	-
		paragraph	0.2935 (45.80%)	0.1444 (122.84%)	0.2390 (97.85%)	0.0952 (68.17%)	0.3666 (8.43%)	0.0555 (53.31%)	0.1549 (10.09%)	**0.3123** (8.66%)	**0.3667** (2.32%)
2.0	0.4	word	0.3062 (56.78%)	0.1694 (100.71%)	0.2780 (163.01%)	0.0710 (123.97%)	0.3587 (13.80%)	0.0953 (49.61%)	0.1979 (27.19%)	-	-
		sentence	0.3256 (56.24%)	0.1659 (118.87%)	0.3077 (137.79%)	0.0778 (94.01%)	**0.3970** (11.76%)	0.0871 (77.76%)	0.2353 (51.12%)	-	-
		paragraph	0.2884 (42.42%)	0.1355 (114.06%)	**0.3485** (144.05%)	**0.0972** (48.52%)	0.3944 (14.21%)	0.0652 (80.11%)	**0.3052** (40.26%)	0.2958 (6.44%)	0.3569 (2.47%)

The three figures show that sentence-based index produces the best results in terms of the document-level, word-based index for the best results in terms of the passage-level and paragraph-based index for the best results in terms of the aspect-level and passage2-level. Table 2 shows that the best re-ranking aspect results always come from the best Okapi baseline results in Table 1. The best baseline results and their corresponding best re-ranking results have been highlighted as boldface.

For all the baseline results, their re-ranking results and their relative rates of improvement are presented in Table 2, where the relative rates of improvement in the parentheses are always positive. In other words, no matter which baseline is chosen, the proposed Bayesian learning approach can always generate the better re-ranking results. The baseline results used in this paper are generated on four data sets. We believe that a good aspect-level MAP performance can be achieved if a good baseline result is available.

6.2 Influence of Different Parameter Settings

In order to investigate the influence of the tuning constant values, the experimental results in Section 5.1 are further discussed and analyzed. We re-plot the results in Table 1 and 2 graphically as Figure 6. Nine sub-figures stand for the performance in terms of four evaluation measures on genomics 2007 data set, three on genomics 2006 data set, one on genomics 2005 data set and one on genomics 2004 data set respectively. The performance of the original and re-ranking results is shown and compared. The x-axis represents the indices under five parameter settings. The y-axis shows the MAP performance.

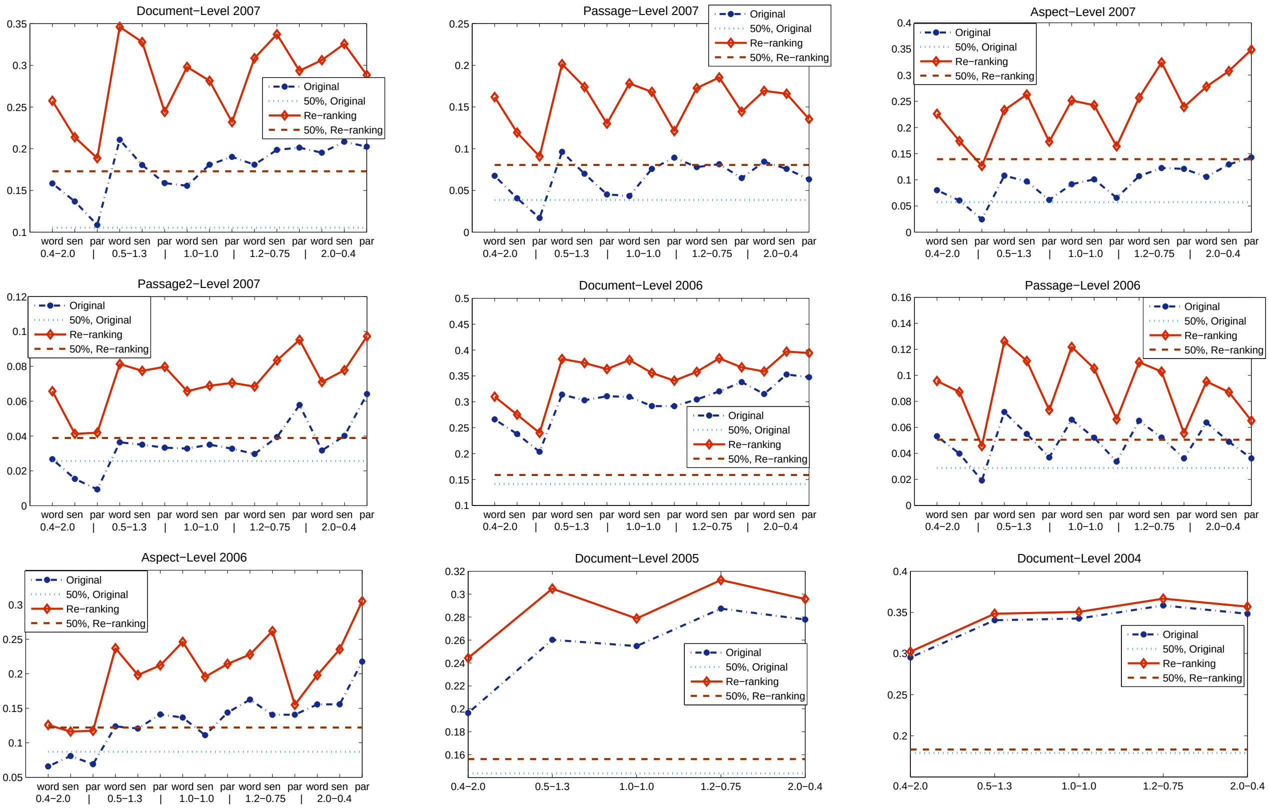

Figure 6: Re-ranking and Original Results, 50% Lines

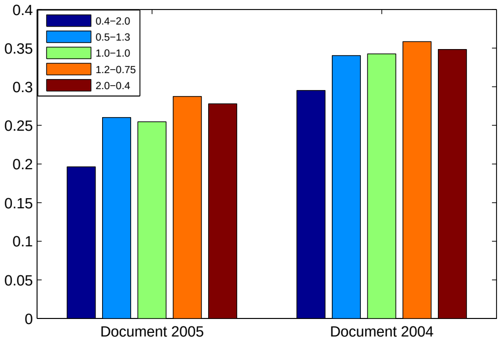

Figure 4: Performance of Baselines 2005 and 2004

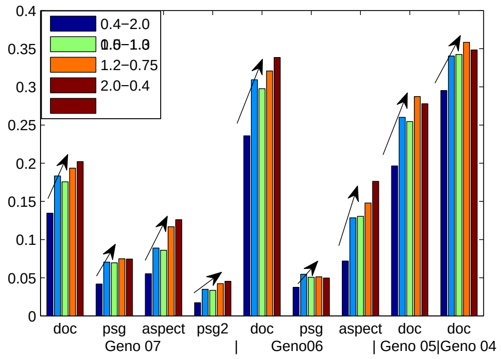

Figure 5: Average Performance

We can observe the following three phenomena. First, when the parameter setting values (k_1, b) is equal to (2.0,0.4), the best aspect-level and passage2-level performance can be obtained for the 2007 and 2006 topics respectively. When the parameter setting values (k_1, b) is equal to (0.5,1.3), the best passage-level performance can be obtained. Second, the average performance of both Okapi 2007 and 2006 baseline results on three indices is increasing when k_1 increases from 0.4 to 2.0 and b decreases from 2.0 to 0.4. Figure 5 shows this trend. Third, in order to discriminate the good and bad performance obtained in the experiments, we define a border line. The border line is to differ the results on four data sets. For all the measures, the results above the border line are treated as good results and the results below the border line are considered as bad results. If we consider the performance interval [50%, 100%] of the best baseline performance as the good performance interval, the 50% border lines in Figure 6 can be used to discriminate the good and bad performance. "50%, Original" and "50%, Re-ranking" indicate the border lines for the original results and re-ranking results respectively. We can see that the interval $(0.5 \sim 2.0, 1.3 \sim 0.4)$ for the parameter settings (k_1,b) should be chosen in order to obtain better MAP performance for all the measures on four data sets.

6.3 Influence of Aspect Number K_r

In this paper, we focus on promoting diversity in ranking using a re-ranking model. In the re-ranking model, we generate hidden aspect subsets by iteratively computing the maximum posterior probabilities. The exact aspect num-

Table 3: Dynamic Aspect Number k_r

k_1	b	Index	Ave # of Re-ranking		Ave # of Re-ranking		Ave # of Re-ranking		Ave # of Re-ranking	
			2007		2006		2005		2004	
0.4	2.0	word	**40**	best	38		-		-	
		sentence	44		**49**	best	-		-	
		paragraph	48		41		29	best	34	best
0.5	1.3	word	35		38		-		-	
		sentence	**40**	best	41		-		-	
		paragraph	49		**49**	best	31	best	37	best
1.0	1.0	word	35		36		-		-	
		sentence	**37**	best	40		-		-	
		paragraph	48		**47**	best	29	best	35	best
1.2	0.75	word	35		**45**	best	-		-	
		sentence	**41**	best	40		-		-	
		paragraph	48		37		30	best	36	best
2.0	0.4	word	38		37		-		-	
		sentence	43		40		-		-	
		paragraph	**40**	best	**50**	best	31	best	38	best
Average of best			40		48		30		36	

ber k_r, which is the number of hidden aspect subsets in the re-ranking model, is determined dynamically for each topic. Different topic usually has different aspect number k_r. We calculate the average number k_r for all topics by simple averaging the aspect numbers on each data set.

As shown in Table 3, the average aspect numbers obtained from the best runs for each setting are independent of the tuning constant values (k_1,b) and indexing granularities. For different tuning constant values (k_1,b) and indexing granularities, the average numbers k_r from the best runs are around 40, 48, 31 and 36 respectively for the genomics 2007-2004 data sets. This shows that the number of topic aspects is determined by the document collection given, which matches our intuition. We have done an extensive experiments with many different settings. Due to space limitation, we did not present all experimental results for the parameter settings within intervals of given values. However, all other results are consistent with those presented here.

As defined in Section 3.1, $\Theta = \{\theta_1, \theta_2, ..., \theta_k\}$ is a set of aspects, where θ_j is a hidden aspect vector. Θ is determined by Bayesian learning approach statistically, which is not necessarily the same as the set of real aspects defined by the human genomics experts. This explains why k_r may not be equal to the aspect number provided by the gold standard of the TREC Genomics Track.

6.4 Comparisons with K-Mean Algorithm

In order to further evaluate our proposed approach to promoting diversity in ranking, we study how K-mean algorithm [14] performs on four data sets. We also choose a genomics topic as an example for more discussion in Section 6.5.

Here we only present the average improvements of K-mean algorithm on the data sets. However the experiments of K-mean algorithm are conducted under five parameter settings and three indices. The average improvements are calculated by the relative rates of K-mean algorithm over their corresponding original results. A series of K values are tested as 5, 10, 20, 50, 80, 100, 200 and 500. For the 2007 and 2006 data sets, we focus on the aspect-level. For the 2005 and 2004 data sets, we focus on the document-level. Although the proposed re-ranking model mainly contributes to promote diversity in ranking, it also works for the document-level retrieval. Table 4 presents us the improvements for both re-ranking model and K-mean algorithm in terms of average aspect-level and document-level MAP. The improvements of the re-ranking model are the relative rates over the corresponding original results. We can conclude that the re-ranking model is much stabler than K-mean algorithm.

6.5 An Example

For the proposed method, the aspect number k is determined dynamically for each topic. However, the number k in K-mean algorithm is set to a fixed number at the beginning. K-mean may not perform well if a non-optimal k is given. Let us take the Topic 220 of the Genomic 2007 Track as an example and consider it on paragraph-based index. As shown in Table 5, the aspect number $k = 32$ is obtained by the proposed method and the corresponding aspect-level MAP performance is 0.9167. This result is significantly better than the original result and all results acquired by K-mean.

K-mean algorithm achieve the best results when k is initialized to 20 or 50, but are still worse than the original result. That is, for Topic 220, K-mean algorithm makes negative contributions for re-ranking the original results using different $k's$. We believe this is because K-mean is not running on an optimal k, which is difficult to find without good prior knowledge. By going through the re-ranking results from K-mean algorithm, we find that this typically happens on many other genomics topics. So we conclude this is one of the reasons why K-mean always performs worse than our approach.

7. CONCLUSIONS

The contribution of this paper is three-fold. First, we propose a re-ranking model for promoting diversity in ranking in the biomedical domain. We build up this model by generating the aspect subsets through iteratively computing the maximum probability of the hidden property for each passages and re-ranking the results based on these subsets. We find that performance can be improved by re-ranking when hidden properties can be properly captured. Second, the exact number of k_r subsets for each topic is determined dy-

Table 4: Average Improvements Comparisons of Re-ranking and K-mean

		Re-ranking	K-mean Algorithm							
			k=5	k=10	k=20	k=50	k=80	k=100	k=200	k=500
Genomics 2007	Aspect	172.67%	6.29%	6.83%	**8.36%**	6.33%	4.82%	6.66%	7.26%	6.77%
Genomics 2006	Aspect	58.16%	6.73%	7.38%	7.11%	6.97%	7.80%	7.99%	8.35%	**9.53%**
Genomics 2005	Document	13.22%	0.59%	**1.03%**	0.32%	0.33%	-0.80%	0.66%	0.21%	0.76%
Genomics 2004	Document	2.37%	**0.73%**	0.35%	0.11%	-0.97%	-0.80%	0.26%	0.48%	0.55%

Table 5: An Example: Performance for Topic 220

Baseline results	Paragraph-based k=32	K-mean Algorithm							
		k=5	k=10	k=20	k=50	k=80	k=100	k=200	k=500
0.6625	**0.9167**	0.5268	0.5576	**0.6220**	0.6115	0.5884	0.5884	0.5417	0.5403

namically when there are no passages moving among the subsets. We find that the subset number k_r is independent of the indices and the parameter settings. Third, a series of comprehensive experiments have been conducted to evaluate and analyze the results under different parameter settings (k_1,b) on different indices. The experimental results on the 2004-2007 Genomics data sets show that the Bayesian learning approach is promising. We also compare the performance of our method with K-mean algorithm in terms of the aspect-level. The experimental results show the stability of our proposed model. K-mean algorithm can generally improve performance over base line results. However, its problem is in that we usually do not have good prior knowledge about k. In addition, in order to obtain the better performance, how to choose the parameter settings (k_1,b) are carefully analyzed and the interval of (k_1,b) for good performance has been found. Simple rules have been found regarding how to adjust them to reach better performance.

Our future work includes investigating the effectiveness of the Bayesian learning approach on other data sets such as TREC blog data sets and better baseline results from other groups. We will work on the opinion retrieval for blogs and focus on searching diversity of blogs. In addition, we plan to apply the EM method and PLSA model to promoting diversity on Genomics research. This is also our ongoing work.

8. ACKNOWLEDGEMENTS

This research is supported in part by NSERC of Canada and the Early Researcher Award/Premier's Research Excellence Award. We thank four anonymous reviewers for their excellent comments on this paper.

9. REFERENCES

[1] M. Beaulieu, M.Gatford, X. Huang, S. Robertson, S. Walker and P. Williams (1997). Okapi at TREC-5. In *Proc. of TREC-5*, NIST Special Publication, Nov. 1997. pp.143-166.

[2] Stefan Buttcher, Charles L. A. Clarke and Gordon V. Cormack (2004). Domain-Specific Synonym Expansion and Validation for Biomedical Information Retrieval (MultiText Experiments for TREC 2004). *Proc. of TREC-13*, 2004.

[3] J. Carbonell and J. Goldstein (1998). The Use of MMR, Diversity-based Reranking for Reordering Documents and Producing Summaries. In *Proc. of the 21st ACM SIGIR Conference*, Melbourne, Australia, August 24-28, 1998.

[4] Dina Demner-Fushman et al (2007). Combining resources to find answers to biomedical questions. In *Proc. of TREC-16*, NIST Special Publication, Nov. 2007.

[5] Andrew B. Goldberg, David Andrzejewski, Jurgen Van Gael, Burr Settles, Xiaojin Zhu and Mark Craven (2006). Ranking Biomedical Passages for Relevance and Diversity. In *Proc. of TREC-15*, NIST Special Publication, Nov. 2006.

[6] William Hersh et al (2005). TREC 2005 Genomics Track Overview. In *Proc. of TREC-14*, NIST Special Publication, Gaithersburg, MD, Nov. 2005.

[7] William Hersh, Aaron M. Cohen, Lynn Ruslen and Phoebe M. Roberts (2007). TREC 2007 Genomics Track Overview. In *Proc. of TREC-16*, NIST Special Publication, Gaithersburg, MD, Nov. 2007.

[8] Qinmin Hu and Xiangji Huang (2009). Passage Extraction and Result Combination for Genomics Information Retrieval. To appear in *Journal of Intelligent Information Systems*, 2009.

[9] Xiangji Huang, Ming Zhong and Si Luo (2005). York University at TREC 2005: Genomics Track. In *Proc. of TREC-14*, NIST Special Publication, Gaithersburg, MD, November 2005.

[10] Xiangji Huang, Bin Hu and Hashmat Rohian (2006). York University at TREC 2006: Genomics Track. In *Proc. of TREC-15*, NIST Special Publication, Gaithersburg, MD, November 2006.

[11] Xiangji Huang, Yan Huang, Miao Wen, Aijun An, Yang Liu and Josiah Poon (2006). Applying Data Mining to Pseudo-Relevance Feedback for High Performance Text Retrieval. In *Proc. of ICDM'06*, Hong Kong, December 2006.

[12] Xiangji Huang, Miao Wen, Aijun An and Yan-Rui Huang (2006). A Platform for Okapi-Based Contextual Information Retrieval. *Proc. of the 29th ACM SIGIR Conference*, Washington, August 6-11, 2006.

[13] Harold Jeffreys (1961). Theory of Probability, 3rd Edition, Oxford University Press, 1961.

[14] J. B. MacQueen (1967). Some Methods for Classification and Analysis of Multivariate Observations. *Proc. of 5th Berkeley Symposium on Mathematical Statistics and Probability*, Berkeley, University of California Press, pp 281-297, 1967.

[15] Stephen E. Robertson and Steve Walker (1994). Some Simple Effective Approximations to the 2-Possion Model for Probabilistic Weighted Retrieval. *Proc. of the 17th ACM SIGIR Conference*, Dublin, Ireland, July 2-6, 1994.

[16] C. Zhai, W. Cohen and J. Lafferty (2003). Beyond Independent Relevance: Methods and Evaluation Metrics for Subtopic Retrieval. *Proc. of the 26th ACM SIGIR Conference*, Toronto, Canada, July 28-August 1, 2003.

[17] Y. Zhang, J. Callan and T. Minka (2002). Novelty and Redundancy Detection in Adaptive Filtering. *Proc. of the 25th ACM SIGIR Conf.*, Tampere, Finland, August 11-15, 2002.

[18] B. Zhang, H. Li, Y. Liu, L. Ji, W. Xi, W. Fan, Z. Chen and W.Y. Ma (2005). Improving Web Search Results Using Affinity Graph. *Proc. of the 28th ACM SIGIR Conference*, Salvador, Brazil, August 15-19, 2005.

[19] Ming Zhong and Xiangji Huang (2006). Concept-Based Biomedical Text Retrieval. *Proc. of the 29th ACM SIGIR Conference*, Washington, August 6-11, 2006.

[20] Wei Zhou and Clement Yu (2007). TREC Genomics Track at UIC. In *Proc. of TREC-16*, NIST Special Publication, Gaithersburg, MD, Nov. 2007.

[21] X. Zhu, A. Goldberg, Van Gael and D. Andrzejewski (2007). Improving Diversity in Ranking Using Absorbing Random Walks. *Proc. of NAACL-HLT*, Rochester, USA, April 22-27, 2007.