Investigation of Web Query Refinement via Topic Analysis and Learning with Personalization *

Lidong Bing Wai Lam
Department of Systems Engineering and Engineering Management
The Chinese University of Hong Kong
Shatin, Hong Kong
{ldbing, wlam}@se.cuhk.edu.hk

ABSTRACT

We investigate the benefits of latent topic analysis and learning with personalization for Web query refinement. Our proposed framework exploits a latent topic space, which is automatically derived from a query log, to leverage the semantic dependency of terms in a query. Another major characteristic of our framework is an effective mechanism to incorporate personal topic-based profile in the query refinement model. Moreover, such profile can be automatically generated achieving personalization of query refinement. Preliminary experiments have been conducted to investigate the query refinement performance.

Categories and Subject Descriptors

I.5.1 [**Pattern Recognition**]: Models—*Statistical*

General Terms

Algorithms

Keywords

query refinement, personalization, bookmark data, query log

1. INTRODUCTION

Web query refinement aims at reformulating a given Web query to improve search result quality. There are three broad types of refinement, namely, substitution [8, 13], expansion [1, 4, 15, 6], and deletion [9, 10, 16]. Besides these broad types, some other fine-grained types include stemming [12], spelling correction [3], abbreviation expansion [14], etc. In [5], a linear chain Conditional Random Field model is employed to deal with these fine-grained refinement operations. For the broad types of refinement mentioned above, a common approach is to generate some candidate queries first, and then a scoring method is designed to assess the quality of these candidates. For example, Wang and Zhai proposed a contextual model by investigating the context similarity of terms in history queries [13]. Two terms with similar context are used to substitute each other in candidate query generation. Then a context based translation model is employed to score the candidate queries. Jones et al. employed hypothesis likelihood ratio to identify those highly related query phrases or term pairs in user sessions [8]. The above existing methods make use of history queries in a query log, and exploit the term context information to generate term pairs and score new candidate queries. One shortcoming of existing context-based methods is that they cannot deal with some ambiguous terms effectively especially when a term has very diverse contexts in history queries.

This paper focuses on term substitution in query refinement. Our proposed framework also consists of two phases, namely, candidate query generation and candidate query scoring. For candidate query generation, we employ a method similar to the term substitution pattern mining presented in [13], which generates a candidate query by substituting one term in the input query and is capable of keeping the semantic relation between the candidate and the input query. For candidate query scoring, we propose a framework that considers semantic dependency of latent topics of term sequence in a given query. Our proposed model exploits a latent topic space, which is automatically derived from a query log, to leverage the semantic dependency of terms in a query. When we score a candidate query, the latent topic sequence of the query is used as hidden evidence to guide the semantic dependency assessment. Another major characteristic of our framework is an effective mechanism to incorporate personal topic-based profile in the query refinement model. Moreover, such profile can be automatically generated from a query log achieving personalization of query refinement. Our final hybrid scoring model combines latent topic evidence and a bigram-based language model. Preliminary experiments have been conducted to investigate the query refinement performance.

2. CANDIDATE QUERY SCORING

Since we currently focus on the investigation of query term substitution, our candidate query scoring method aims at comparing queries of the same length.

*The work described in this paper is substantially supported by grants from the Research Grant Council of the Hong Kong Special Administrative Region, China (Project Codes: CUHK4128/07 and CUHK413510) and the Direct Grant of the Faculty of Engineering, CUHK (Project Codes: 2050442 and 2050476). This work is also affiliated with the Microsoft-CUHK Joint Laboratory for Human-centric Computing and Interface Technologies.

SIGIR Workshop on Query Representation and Understanding'11 July 28, 2011, Beijing, China.

2.1 Latent Topic Analysis

A typical record in query log can be represented as a 4-tuple $(anonymous\_user\_id, query, clicked\_url, time)$. One direct method for using query log data for latent topic analysis is to treat each *clicked_url* as a single document unit. This approach will suffer from the data sparseness problem since most URLs only involve very small number of queries. Instead of adopting such a simple strategy, we aggregate all the queries related to the same host together and construct one pseudo-document for each host. For example, the pseudo-document of "www.mapquest.com" consists of the queries "mapquest", "travel map", "driving direction", and so on. Some general Web sites such as "en.wikipedia.org" are not suitable for latent topic analysis because they involve large amount of queries as well as many query terms, and cover very diverse semantic aspects. To tackle this problem, we first sort the host pseudo-documents in descending order according to the number of distinct query terms they have. Then, the top ranked pseudo-documents are eliminated in our latent topic discovery process.

We employ the standard Latent Dirichlet Allocation (LDA) algorithm [2] to conduct the latent semantic topic analysis on the collection of host-based pseudo-documents. In particular, GibbsLDA[1] package is used to generate the set of latent topics. Let Z denote the set of latent topics. Each topic z_i is associated with a multinomial distribution of terms. The probability of each term t_k given a topic z_i is denoted by $P(t_k|z_i)$.

2.2 Latent Topic Based Scoring with Personalization

A candidate query $q : t^1 \cdots t^n$ is composed of a sequence of terms, where the superscript indicates the position of the term and $t^r (1 \leq r \leq n)$ may take any admissible term in T where T denotes the vocabulary set. The topic of t^r is denoted by z^r, which denotes an admissible topic in Z. We investigate a Hidden Markov Model (HMM) as shown in Figure 1 to score the candidate query. The query terms are observable and represented by filled nodes. The latent topics are unobservable and represented by empty nodes. Different from a common application of HMM that solves the tagging or decoding problem, we make use of this model to compute the marginal joint probability of the term sequence denoted as $P(t^{1:n})$ for scoring the candidate query. Let $z^{1:n}$ denote the topic sequence. The candidate query score can be computed by:

$$P(t^{1:n}) = \sum_{z^{1:n}} P(t^{1:n}, z^{1:n}) \tag{1}$$

Due to the dependency structure of the model, the joint probability of the topic sequence and term sequence can be expressed as:

$$P(t^{1:n}, z^{1:n}) = \prod_{r=1}^{n} P(t^r|z^r) P(z^1) \prod_{r=2}^{n} P(z^r|z^{r-1}) \tag{2}$$

This model involves the term emission probability from a topic $P(t_k|z_i)$ which can be regarded as considering the relationship of terms and topics. Another model parameter $P(z_j|z_i)$ captures the relationship of two topics in the modeling and scoring process. This pairwise topic relationship offers a means to incorporate different strategies of considering topic contexts governing neighboring terms in the candidate query. One strategy is to favor neighboring terms in a query sharing similar topic context. To facilitate this objective, we design a scheme that favors the pairwise topic probability if the semantic content of the two topics has high degree of similarity.

[1] http://gibbslda.sourceforge.net/

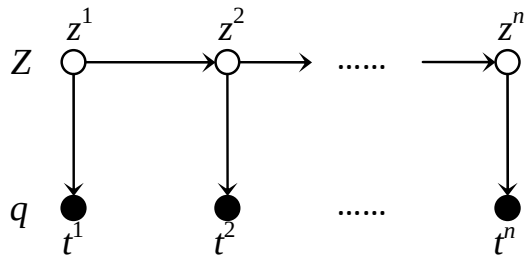

Figure 1: Latent topic model for a candidate query.

To provide a better quality for query refinement, we investigate the capability of personalization. Different users have different personalized preference which can be revealed by the fact that their queries concentrate on a particular set of topics. For example, teenagers usually issue queries related to basketball, computer game, and so on. Therefore, each user has an inherent preference on different topics. This personalized interest can be encapsulated in a topic-based profile. Then the profile can be taken into account in the calculation of the topic-based score. We describe how to incorporate a topic-based profile in the scoring model.

Let $\Pi^u = \{\pi_1^u, \pi_2^u, \cdots, \pi_{|Z|}^u\}$ denote the profile of the user u, where $\pi_i^u = P(z_i|u)$ is the probability that the user u prefers the topic z_i. Considering the probability constraint, we have $\sum_i \pi_i^u = 1$. Equation 2 can be modified as:

$$P(t^{1:n}, z^{1:n}) = \prod_{r=1}^{n} P(t^r|z^r) \pi_{z^1}^u \prod_{r=2}^{n} P(z^r|z^{r-1}), \tag{3}$$

where $\pi_{z^1}^u = P(z^1|u)$. Thus, when we score a query, the preference of u is taken into account.

To compute efficiently the score, we make use a dynamic programming technique commonly adopted for tackling the computational requirement of HMM models. The forward variable $\alpha_r(i)$ is defined as:

$$\alpha_r(i) \triangleq P(t^{1:r}, z^r = z_i), \tag{4}$$

which is an intermediate score of the partial query $t^1 \cdots t^r$ given the topic z_i at the position r. When $r = 1$, we set $\alpha_1(i) = \pi_i^u P(t^1|z_i)$. The recursive calculation of α is:

$$\alpha_r(i) = \Big[\sum_{z_j \in Z} \alpha_{r-1}(j) P(z^r = z_i | z^{r-1} = z_j) \Big] P(t^r|z^r = z_i), \tag{5}$$

where $P(z^r = z_i|z^{r-1} = z_j)$ is the pairwise topic probability at the position r.

The topic-based score $S_t(q)$ is calculated by summing over all possible z_i of $\alpha_n(i)$:

$$S_t(q) = P(t^{1:n}) = \sum_{z_i \in Z} \alpha_n(i). \tag{6}$$

2.3 Model Parameter Design

The model parameter $P(t_k|z_i)$ can be readily obtained from the probability of a term given a topic in the LDA analysis mentioned above.

For the parameter $P(z_j|z_i)$ corresponding to the pairwise topic relationship, we consider the objective of query refinement. The topics of terms in the same query tend to remain consistent from semantic point of view because of the unique search intention of the user for the given query. Different strategies can be developed to achieve this objective. As a preliminary investigation, we examine the degree of semantic similarity of the pair of topics. Basically, the more similar between two latent topics, the higher is this probability. Therefore, we calculate $P(z_j|z_i)$ as:

$$P(z_j|z_i) = \frac{sim(z_i, z_j)}{\sum_{z_k \in Z} sim(z_k, z_i)}, \quad (7)$$

where $sim(z_i, z_j)$ is a similarity measure of the topics z_i and z_j. If z_i and z_j are highly related, the value of $P(z_j|z_i)$ is large. Specifically, we adopt the cosine similarity as the similarity measure. The semantic similarity of two topics is calculated as:

$$sim(z_i, z_j) = \frac{\sum_{t_k} P(t_k|z_i)P(t_k|z_j)}{\sqrt{\sum_{t_k} P(t_k|z_i)^2}\sqrt{\sum_{t_k} P(t_k|z_j)^2}}. \quad (8)$$

Another option for calculating $sim(z_i, z_j)$ is Kullback-Leibler (KL) divergence, which will be explored in future work.

2.4 Deriving Personal Topic Profiles Automatically

The topic-based profile Π^u defined in the above model supports personalized query refinement. This profile can be automatically derived from a query log using the inference algorithm of LDA which computes the posterior distribution of the hidden topics given a document. First, we construct a personal pseudo-document for a particular user u. Then we generate a personal topic profile by conducting inference on the latent topic model.

Following the idea of the construction of host-based pseudo-documents in latent topic analysis, this time we aggregate all the queries issued by the user u, and Let $\mathcal{U}$ denote the generated pseudo-document. The inference algorithm in GibbsLDA package based on Gibbs sampling is invoked for the pseudo-document $\mathcal{U}$ with the model parameters obtained in the latent topic analysis process. Then, the probability distribution $\{P(z_1|\mathcal{U}), P(z_2|\mathcal{U}), \cdots\}$ obtained from the inference algorithm is used as the profile of u.

2.5 Final Hybrid Scoring

In the above topic-based personalized scoring method, the term context is considered indirectly with the successive topics. Although the term context information can be captured to some extent, we can further enhance the quality. To capture the term context dependency directly, as well as the topic-based score, we develop a hybrid method which combines these two scores together as follows:

$$S_h(q) = \lambda \log S_t(q) + (1 - \lambda) \log S_b(q), \quad (9)$$

where $S_b(q)$ is a bigram-based score of q, and calculated as:

$$S_b(q) = \prod_{i=1}^{n} P(t^i|t^{i-1}). \quad (10)$$

$P(t^1|t^0)$ is set to $P(t^1)$. λ is the parameter for controlling their relative weights.

3. PRELIMINARY EXPERIMENTS

3.1 Experimental Setup

We use the AOL query log [11] from 1 March, 2006 to 31 May, 2006. The raw data contains a lot of noise, so some cleansing operations are performed, such as navigation query removal and stop words removal. We adopt a hybrid method to detect user session boundary [7] and remove those sessions without any clicking. The data set is split into two sets, namely, the history set and the test set. The history set contains the first two months' log and the test set contains the third month's log. The pseudo-documents for latent topic analysis are constructed with the history set. We select hosts involving at least 5 queries and we remove the top 0.1% of the hosts according to the distinct query terms they have. Finally, 189,670 pseudo-documents are obtained and the number of topics is set to 30. The value of λ used is 0.4 for the personalized model and 0.2 for the non-personalized model, with which the best performance is achieved. The bigram language model in our hybrid scoring method is estimated from the queries in the history set of the query log, and the Dirichlet smoothing technique [17] is employed to tackle the problem of data sparseness.

For conducting the comparison, we implement a context based term association (CTA) method presented in Wang and Zhai [13]. We denote the query scoring step in CTA as "CTA-SCR". We generate the contextual and translation models for the top 100,000 terms in the history set, and apply the threshold in [13] to filter out the noise.

We conduct an automatic method by utilizing the session information to evaluate the performance of the refinement models. In a search session, when users feel unsatisfied with the results of current query, they may refine current query and search again. We differentiate two kinds of queries, namely, satisfied queries and unsatisfied queries. In a user session, the query which causes at least one URL clicked and is located in the end of the session is called a satisfied query. The query which is located just ahead of the satisfied query in the same user session is called an unsatisfied query. We collect a set of unsatisfied queries for conducting the experiment and their corresponding satisfied queries are treated as the benchmark answer for the refinement task. The scoring method will return a ranked list of the candidate queries. Then we evaluate the performance of the method at top m. If the true answer can be found in the top m candidates, that query is considered as successful. Accuracy is defined as the total number of successful queries divided by the total number of test queries.

For generating user profile, we randomly select 400 users who have more than 100 sessions in the history set. Then the queries issued by the same user in the history set are aggregated together to generate user profile. For each user, we select one of his unsatisfied queries from the test set which has at least 3 terms, and use this query as the input of the refinement models.

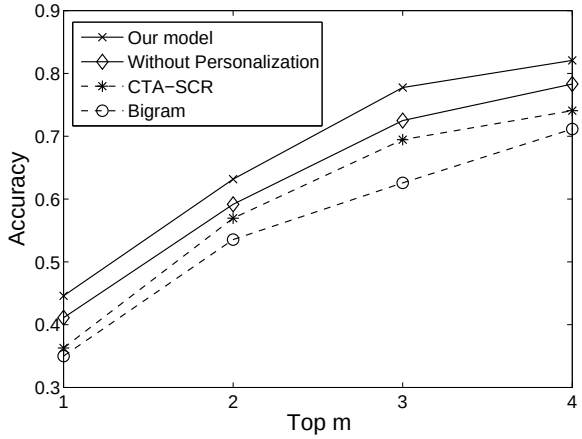

Figure 2: Performance of different scoring methods.

3.2 Results

The performance of different scoring methods is given in Figure 2. "Our model" is the framework presented in this paper combining personalized topic scoring and bigram scoring. "Without Personalization" is our model without personalization. "CTA-SCR" is a baseline method as described above. "Bigram" is another baseline model using a pure bigram method. It can be observed that our framework that considers personalization achieves the best performance. It indicates that our method can rank good suggestions of query refinement higher. We also find that with user profiles, the topic-based scoring part is more reliable and it plays a more important role. "CTA-SCR" performs better than the pure bigram method, but not as effective as our method.

4. CONCLUSIONS AND FUTURE WORK

We present a framework for performing term substitution in Web query refinement based on a personalized topic-based hybrid scoring method. Our method can detect and consider the semantic dependency of terms in queries. From the experimental results, we observe that taking both the semantic dependency and personalization into account can help offer better query refinement quality.

In this preliminary work, the model parameters, namely, emission probability and transition probability, are directly estimated from the results of latent topic analysis. In our future work, we intend to employ Expectation Maximization (EM) algorithm to conduct more precise parameter estimation.

5. REFERENCES

[1] B. Billerbeck, F. Scholer, H. E. Williams, and J. Zobel. Query expansion using associated queries. In *Proceedings of the twelfth international conference on Information and knowledge management*, CIKM '03, pages 2–9, 2003.

[2] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022, March 2003.

[3] S. Cucerzan and E. Brill. Spelling correction as an iterative process that exploits the collective knowledge of web users. In *Proceedings of EMNLP*, pages 293–300, 2004.

[4] H. Cui, J.-R. Wen, J.-Y. Nie, and W.-Y. Ma. Probabilistic query expansion using query logs. In *Proceedings of the 11th international conference on World Wide Web*, WWW '02, pages 325–332, 2002.

[5] J. Guo, G. Xu, H. Li, and X. Cheng. A unified and discriminative model for query refinement. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '08, pages 379–386, 2008.

[6] B. He and I. Ounis. Combining fields for query expansion and adaptive query expansion. *Inf. Process. Manage.*, 43:1294–1307, September 2007.

[7] D. He, A. Göker, and D. J. Harper. Combining evidence for automatic web session identification. *Inf. Process. Manage.*, 38:727–742, September 2002.

[8] R. Jones, B. Rey, O. Madani, and W. Greiner. Generating query substitutions. In *Proceedings of the 15th international conference on World Wide Web*, WWW '06, pages 387–396, 2006.

[9] G. Kumaran and J. Allan. A Case for Shorter Queries, and Helping User Create Them. In *Proceedings of Human Language Technologies: The Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 220–227, 2006.

[10] G. Kumaran and J. Allan. Effective and efficient user interaction for long queries. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '08, pages 11–18, 2008.

[11] G. Pass, A. Chowdhury, and C. Torgeson. A picture of search. In *Proceedings of the 1st international conference on Scalable information systems*, InfoScale '06, 2006.

[12] F. Peng, N. Ahmed, X. Li, and Y. Lu. Context sensitive stemming for web search. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '07, pages 639–646, 2007.

[13] X. Wang and C. Zhai. Mining term association patterns from search logs for effective query reformulation. In *Proceeding of the 17th ACM conference on Information and knowledge management*, CIKM '08, pages 479–488, 2008.

[14] X. Wei, F. Peng, and B. Dumoulin. Analyzing web text association to disambiguate abbreviation in queries. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '08, pages 751–752, 2008.

[15] J. Xu and W. B. Croft. Query expansion using local and global document analysis. In *Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '96, pages 4–11, 1996.

[16] X. Xue, S. Huston, and W. B. Croft. Improving verbose queries using subset distribution. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, CIKM '10, pages 1059–1068, 2010.

[17] C. Zhai and J. Lafferty. A study of smoothing methods for language models applied to information retrieval. *ACM Trans. Inf. Syst.*, 22:179–214, April 2004.