5J-1

Name Disambiguation Using Topics Extracted from Web Directories in Information-explosion Era

ヴークァンミン[†]　高須淳宏[‡]　安達淳[‡]

[†] 東京大学情報理工学系研究科　　[‡] 国立情報学研究所

1 Introduction

Information in the World Wide Web (WWW) is increasing explosively and the needs to search for useful information are increasing. A certain amout of these needs are to search for people using personal name queries. Since a name is often shared by several people, search results for a name query often contain documents relevant to more than one person.

In this research, we use a reranking method to differentiate the person of interest from other people. First, users select from the result set a document mentioning the person of their interests. Then, the system reranks documents in the order of relevances to the selected document. We use web directories to improve the measurements of document similarities. We use the latent Dirichlet allocation method(LDA)[2] to extract topics in web directories and use extracted topics to modify documents in search results.

2 Related works

In [1], they used the vector space model method to create context vectors of people and used inner products of context vectors for document similarities. In [5], they built contexts of people using the second order context vector method. This method used statistical calculations on documents and terms to recognize important terms related to contexts of people. In [3], they extracted key phrases in documents and used information from search engines to enrich contexts for key phrases. In [4], they used a named entity recognition method to extract entities related to people to and built contexts of people.

3 Our approach

3.1 Extraction of topics in web directories

We use the LDA method[2] to extract latent topics in web directories. Let D, T, and W be the numbers of directories, latent topics, and different words, respectively. For a directory dir, its topic distribution is represented by a vector $\Theta_{dir} = (\vartheta_{dir,1}, \vartheta_{dir,2}, ...\vartheta_{dir,T})$. In the conventional LDA method[2], the probability distribution of topic vectors is assumed to follow the same Dirichlet prior for all documents. However, since we know that each directory has its own specific topic, we assume that documents in a directory are influenced mainly by the specific topic associated with that directory, while receiving small influences from topics of other directories. A hyper-parameter vector $\vec{\alpha}^{(dir)}$ for a directory dir is set to have a large hyper-parameter $\alpha_{dir} = k\alpha$ for its associated specific topic, and small hyper-parameters $\alpha_j = \alpha$ for other topics $j \neq dir$: $\vec{\alpha}^{(dir)} = (\alpha, \alpha, ..., \alpha_{dir} = k\alpha, ..., \alpha)$.

The topic distribution vector of the directory dir has the distribution density as follows.

$$P(\Theta_{dir}|\vec{\alpha}^{(dir)}) = \frac{\Gamma((k+T-1)\alpha)}{\Gamma(k\alpha)\Gamma(\alpha)^{T-1}} \vartheta_1^{\alpha-1} \cdot ... \vartheta_{dir}^{k\alpha-1} ... \cdot \vartheta_T^{\alpha-1} \tag{1}$$

We call k the bias factor of directories.

For a topic t, its word distribution is represented by a vector $\Phi_t = (\varphi_{t,1}, \varphi_{t,2}, ...\varphi_{t,W})$. The probabilitistic density of Φ_t is also assumed to follow a Dirichlet distribution.

$$P(\Phi_t|\vec{\beta}) = \frac{\Gamma(\sum_{i=1}^{W} \beta_i)}{\prod_{i=1}^{W} \Gamma(\beta_i)} \varphi_1^{\beta_1 - 1} ... \varphi_W^{\beta_W - 1} \tag{2}$$

where $\vec{\beta} = (\beta_1, \beta_2, ..., \beta_W)$ and $\beta_1, \beta_2, ..., \beta_W$ are hyper-parameters.

The Gibbs sampling method is used to assign a topic ID for each word in documents and to calculate topic distribution vectors. In our research, since we use different sets of hyperparameters for different directories, the new update formula in the Gibbs sampling procedure is as follows.

$$P(t_i = t|\overrightarrow{t_{-i}}, \overrightarrow{w}) \propto$$
$$\begin{cases} \frac{n_{-i,dir_d}^{(t)} + k\alpha}{[\sum_{t'=1}^{T} n_{-i,dir_d}^{(t')}] + (k+T-1)\alpha} \frac{n_{-i,topic_t}^{(w)} + \beta_w}{\sum_{w'=1}^{W} [n_{-i,topic_t}^{(w')} + \beta_{w'}]}, \\ \quad \text{if } t = dir_d \\ \frac{n_{-i,dir_d}^{(t)} + \alpha}{[\sum_{t'=1}^{T} n_{-i,dir_d}^{(t')}] + (k+T-1)\alpha} \frac{n_{-i,topic_t}^{(w)} + \beta_w}{\sum_{w'=1}^{W} [n_{-i,topic_t}^{(w')} + \beta_{w'}]}, \\ \quad \text{if } t \neq dir_d \end{cases} \tag{3}$$

where $w = w_i$, dir_d is the directory contains w_i, $n_{-i,dir_d}^{(t)}$ is the number of words in dir_d to be assigned topic t except w_i, and $n_{-i,topic_t}^{(w)}$ is the total number of times the word w is assigned topic t except w_i.

3.2 Document similarities

Topic feature vectors of words

We calculate a feature vector for word w using the probability $P(w|t)$ as follows.

$$\overrightarrow{p_w} = (p_{w,1}, p_{w,2}, ..., p_{w,T}) \tag{4}$$

where $p_{w,t} \propto P(w|t)$. We normalize $\overrightarrow{p_w}$ so that the summarization of $p_{w,t}$ is equal to 1.

Topic feature vectors of documents

Quang Minh VU[†]　Atsuhiro TAKASU[‡]　Jun ADACHI[‡]

[†]Graduate School of Information Science and Technology, The University of Tokyo, 7-3-1 Hongo, Bunkyo-ku, Tokyo 113-8656, Japan

[‡]National Institute of Informatics, 2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo 101-8430, Japan

Email: {vuminh, takasu, adachi}@nii.ac.jp

Given a word w is observed, we learn the topic distribution attached with w: $(p_{w,1}, p_{w,2}, ..., p_{w,T})$. If we do not observe w, the topic distribution is the same for all topics: $(\frac{1}{T}, \frac{1}{T}, ..., \frac{1}{T})$. Therefore, the information amount conveyed by w is the difference of information amounts between these two distributions.

$$\text{weight}(w) = \log T + \sum_{t=1}^{T} p_{w,t} \log p_{w,t} \qquad (5)$$

The topic feature vector for a new document d is a combination of word feature vectors.

$$\begin{aligned} \overrightarrow{\rho_d} &= \sum_{w \in d} \text{weight}(w) \cdot \overrightarrow{p_w} \\ &= (\rho_1, \rho_2, ..., \rho_T) \qquad (6) \end{aligned}$$

Modification of documents

We use a topic t to modify a document d as follows. Denote $\overrightarrow{d} = (tf_1, tf_2, ..., tf_W)$ as the original document vector, where tf_w is the number of times word w appears in d. Denote the modified document as d_t and its vector as $\overrightarrow{d_t} = (tf_1^{(t)}, tf_2^{(t)}, ..., tf_W^{(t)})$. We assume that terms in the modified document are generated by either the original document d or the topic t. The probability that the modified document d_t generates a word w is derived as follows.

$$\begin{aligned} &P(w|d_t) \\ &= 1 - P(d_t \text{ not generate } w) \\ &= P(w|d) + P(w|t) - P(w|d)P(w|t) \qquad (7) \end{aligned}$$

$$tf_w^{(t)} = P(w|d_t) \cdot \text{length}(d) \qquad (8)$$

Measurement of similarities

Denote (d_1, d_2) as a pair of document. For each document $d_i (i = 1, 2)$, we select m topics that have the top m values of $\rho_{i,t}$. We call these topics as representative topics for the document and denote the set of topics as $R_i = \{t_{i,1}, t_{i,2}, ..., t_{i,m}\}$. We calculate a document similarity via a topic $t \in R_1 \cup R_2$.

$$Sim(d_1, d_2, t) = \rho_{1,t}\rho_{2,t}\overrightarrow{d_{1,t}}\overrightarrow{d_{2,t}} \qquad (9)$$

Next, the document similarity of (d_1, d_2) is defined as the summarization of document similarites via all representative topics $t \in R_1 \cup R_2$.

$$Sim(d_1, d_2) = \sum_{t \in R_1 \cup R_2} \rho_{1,t}\rho_{2,t}\overrightarrow{d_{1,t}}\overrightarrow{d_{2,t}} \qquad (10)$$

4 Experiments

We sent 24 name queries to the Google search engine[1] to get the top 100 documents for each query and disambiguated personal names in each result set. We collected web directories from three web directories: the Dmoz directory[2], the Google directory[3] and the Yahoo directory[4]. We disambiguated personal names by document reranking. First, users selected a document referring to the person of interests and notified the system. Then, the system reranked documents based on the similarities to the selected document. We evaluated the disambiguation performance by measuring precision values with respect to recalls. We compared our approach with the vector space model method and the named entity recognition method. The results were shown in Fig. 1. In terms of averaged precisions, the vector space model method got the performance of 68.6%, the named entity recognition method got the performance of 67.6% and our approach got the performances from 73.1% to 75.0%.

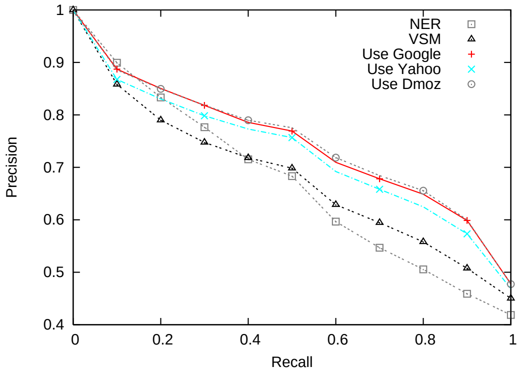

Fig. 1: Disambiguation performances by methods

5 Conclusions

In this paper, we reported our research on name disambiguation in the web. We proposed a new approach that used web directories to measure document similarities more effectively. These similarity results were used to disambiguate personal names by the mean of document reranking. We carried experiments on real documents from the web and verified the improvements of our approach over the vector space model method and the named entity recognition method.

References

[1] Amit Bagga and Breck Baldwin. Entity-based cross-document coreferencing using the vector space model. In *ACL1998*, 1998.

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

[3] D. Bollegala, Y. Matsuo, and M. Ishizuka. Extracting key phrases to disambiguate personal name queries in web search. In *Proceedings of the workshop "How can Computational Linguistics improve Information Retreival?", COLING-ACL 2006*, 2006.

[4] Gideon S. Mann and David Yarowsky. Unsupervised personal name disambiguation. In *Proceedings of Computational Natural Language Learning 2003*, 2003.

[5] T. Pedersen, A. Kulkarni, R. Angheluta, Z. Kozareva, and T. Solorio. Name discrimination by clustering similar contexts. In *Proceedings of the Sixth International Conference on Intelligent Text Processing and Computational Linguistics*, 2005.

[1] http://www.google.com
[2] http://www.dmoz.org
[3] http://directory.google.com
[4] http://dir.yahoo.com