Citation Data Clustering for Author Name Disambiguation

Tomonari Masada
Nagasaki University
Bunkyo-machi 1-14,
Nagasaki, Japan
masada@cis.nagasaki-u.ac.jp

Atsuhiro Takasu
National Institute of Informatics
Hitotsubashi 2-1-2,
Chiyoda-ku
Tokyo, Japan
takasu@nii.ac.jp

Jun Adachi
National Institute of Informatics
Hitotsubashi 2-1-2,
Chiyoda-ku
Tokyo, Japan
adachi@nii.ac.jp

ABSTRACT

In this paper, we propose a new method of citation data clustering for author name disambiguation. Most citation data appearing in the reference section of scientific papers include the coauthor first names with their initials. Hence, we often search citation data by using such an abbreviated name, e.g. "S. Lee" or "J. Chen", and consequently obtain many irrelevant data in the search result, because such an abbreviated name refers to many different persons. In this paper, we propose a method of citation data clustering to construct clusters each of which includes only citation data corresponding to a unique author. Our clustering method is based on a probabilistic model which is an extension of the naive Bayes mixture model. Since our model has two hidden variables, we call it *two-variable mixture model*. In the evaluation experiment, we used the well-known DBLP data set. The results show that the two-variable mixture model can achieve a better balance between precision and recall than the naive Bayes mixture model.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval

Keywords

Name Disambiguation, Unsupervised Learning

1. INTRODUCTION

When we manage a large-scale database of the real-world data, we often face a problem called *name disambiguation*. The name ambiguities can be classified into the following two cases: 1) the same object or the same person is referred to by different names; and 2) many objects or many persons are referred to by the same name. As for 1), we should make groups of the different names so that each group corresponds to a unique object or person. As for 2), we should make groups of the *instances* of the same name so that each group corresponds to a unique object or person. In this paper, we cope with the name ambiguity of case 2) with respect to the instances of abbreviated author names appearing in the citation data. Most citation data in the reference section of scientific papers include coauthor names after abbreviating first names to their initials. Therefore, many different authors can be referred to by the same abbreviated name. Consequently, we will have many irrelevant search results when we use such an abbreviated name as a query passed to the citation database, e.g. Citeseer [1] and DBLP [2]. In this paper, we propose a new method of citaiton data clustering. Our clustering method divides a citation data set, which is obtained as a result of a search using an abbreviated name as a query, into disjoint clusters. When each cluster includes all citation data corresponding to a unique author, we have a complete solution.

However, our problem is difficult because we have only a few clues to make our clustering effective. Citation data are too short a string, and several data fields (e.g. volume, number, and pages) give almost no help. Longer documents (e.g. e-mails, Web pages, and newspaper articles) can provide abundant clues when we try to correctly assign a unique person to each name instance. In contrast, citation data only provide poor clues for disambiguating author names. This difficulty may be overcome by using additional information sources about authors, journals, relevant research fields, etc. However, this option will reduce the scalability of citation database, because we should pay much effort to keep such additional data reliable and consistent. Therefore, we do not take into account such an option. In this paper, we suppose that each citation data consists of the following three fields: coauthor names, title words, and journal or conference name, as in the preceding papers [5][6][7]. These fields appear in almost all citation data and provide stronger clues than other data fields. Our procedure for the evaluation of author name disambiguation is as follows.

- *Retrieval phase.* We collect all citation data including a given abbreviated name, e.g. "J. Smith", from the prepared set of citation data. We call this abbreviated name *query name*, because it can be regarded as a query for retrieval. We denote the query name by q and denote the set of retrieved citation data by $D^q = \{d_1^q, \ldots, d_I^q\}$. We will omit the superscript q when no confusion arise. We use the DBLP citation data set [2] as the prepared set of citation data. Most citation data in this data set include author names with their full names. Therefore, we abbreviate all first names to initials, and use the full names as the

INFOSCALE 2007, June 6-8, Suzhou, China

DOI 10.4108/infoscale.2007.203

correct answers for the evaluation. This abbreviation is artificial but necessary to design a valid evaluation procedure based on the reliable correct answers. The citation data originally including abbreviated names are discarded, because we have no correct answers for such data. We denote the set of all full names abbreviated to q by $F^q = \{f_1^q, \ldots, f_L^q\}$. We say that a citation data d_i^q *corresponds to* a full name f_l^q when d_i^q originally includes q as a full name f_l^q. While the same full name can refer to different authors, we assume that each full name stands for a unique author. This assumption makes us apart from realistic situations. However, it is difficult to prepare the correct answers taking into consideration the fact that the same full name can refer to different authors. We believe that we can evaluate the performance of name disambiguation accurately enough even under this assumption.

- *Disambiguation phase.* We divide D^q into disjoint clusters $\mathcal{G}^q = \{G_1^q, \ldots, G_M^q\}$. This is the very process of name disambiguation. If any two citation data from the same cluster correspond to the same full name, and if any two data from different clusters correspond to different full names, our problem is perfectly solved. In this paper, we use two probabilistic models for clustering: the naive Bayes mixture model (NBM) [11] and a new probabilistic model which is an extension of NBM. The latter model proposed by us has two hidden variables. We call this *two-variable mixture model (TVM)*. Both of these models are trained in an unsupervised manner. Unsupervised learning is desirable to keep our method scalable, because the cost for preparing training data for all possible query names is expensive.

- *Evaluation phase.* We evaluate clustering results with their precisions and recalls. We say that a full name f_l^q *dominates* a cluster G_m^q when the number of the citation data which belong to G_m^q and correspond to f_l^q is larger than any other full names. We denote the dominating full name of a cluster G_m^q by $f^q(G_m^q)$. The precision of G_m^q is the ratio of the number of citation data which belong to G_m^q and correspond to $f^q(G_m^q)$ to the cluster size of G_m^q. The recall of G_m^q is the ratio of the number of citation data which belong to G_m^q and correspond to $f^q(G_m^q)$ to the number of citation data which belong to D^q and correspond to $f^q(G_m^q)$. The precision gets larger when the cluster sizes get smaller. In the extreme case where every cluster is singleton, the precision is equal to 1, but the recall is disastrously small. It is important to achieve a good balance between precision and recall. Our results will show that TVM realizes a better balance than NBM.

This paper is organized as follows. Section 2 presents the previous work concerning name disambiguation. Section 3 provides the formalizations of NBM and TVM. This section also provides EM algorithms for parameter estimation. Section 4 describes the details of the evaluation experiment and presents the results. Section 5 summarizes the paper.

2. PREVIOUS WORK

Name disambiguation is a focal point of recent researches on real-world information integration and data maining. Han et al. [5] provide a supervised method and use the DBLP data and evaluate their method by disambiguating abbreviated author names. However, to prepare training data for all possible abbreviated names is not a realistic requirement. Therefore, recent researches mainly propose unsupervised methods. Dong et al. [4] and Kalashnikov et al. [9] cope with both cases of name ambiguities presented in Section 1 by adopting unsupervised learning framework. However, these two researches assume that we can use additional author information which cannot be extracted from citation data. In this paper, we assume that no additional information sources are available. As a result, we should solve more difficult problem. However, we restrict the scope of our disambiguation method to the ambiguities of case 2) presented in Section 1. We think that the requirement of additional information sources reduces the scalability of citation database by introducing unnegligible cost for keeping such information sources reliable and consistent.

Han et al. [7] provide an unsupervised method based on the spectral clustering. Further, Han et al. [6] propose an unsupervised method based on a probabilistic model which subtly distinguishes various coauthoring patterns. Both of these researches use the DBLP citation data and evaluate their methods by disambiguating abbreviated author names. This is the same setting as our experiment. However, both researches assume that the true number of clusters, i.e., the number of full names which can be abbreviated to the given query name, is known. In this paper, we conduct not only the experiment under the assumption that we know the true number of clusters, but also the experiment under the assumption that we do not know the true number of clusters. In the latter experiment, we set the number of clusters to a constant value larger than the true number for all query names. Moreover, these two researches only use the microaveraged precision for the evaluation. Since these researches use the true number of clusters as an input of clustering, we cannot arbitrarily increase the microaveraged precision by reducing cluster sizes. Therefore, we can obtain a reliable evaluation only with the microaveraged precision. In contrast, we also conduct the experiment under the assumption that we do not know the true number of clusters. Therefore, the evaluation based only on the precision is not reliable. We will use the following four evaluation measures: microaveraged precision/recall and macroaveraged precision/recall.

Our citation data clustering is based on a probabilistic model which is a modification of the naive Bayes mixture model (NBM) [11]. NBM has one hidden variable whose value tells to which cluster each citation data belongs. Each value of the hidden variable corresponds to different multinomial distributions defined over the words appearing in the citation data. Roughly speaking, citation data showing similar distributions of word frequencies are likely to belong to the same cluster. NBM is theoretically simple and practically effective in comparison with k-means [6]. Moreover, the time complexity for parameter estimation is small enough to obtain clusters at the query time. NBM is suitable for name disambiguation on the large-scale citation database. Our new method for name disambiguation is based on a probabilistic model having two hidden variables. We call this *two-variable mixture model (TVM)*. Since TVM is a slight modification of NBM, the time complexity is still small enough to disambiguate a given query name at the query time. The probabilistic model proposed in [6] is also a slight modification of NBM and is effective in its execution time. Both of

this model and TVM are based on the same intuition: coauthor relationship is the most important factor for author name disambiguation. However, our assumption and solution are different from [6]. Their model aims to achieve a higher precision under the assumption that the true number of clusters is known. Our model aims to achieve a better balance between precision and recall with no regard to whether we know the true number of clusters or not.

3. GENERATIVE MODEL FOR CITATION DATA CLUSTERING

The input for our name disambiguation problem is a set $D^q = \{d_1^q, \ldots, d_I^q\}$ of all citation data that include a given query name q. We assume that each citation data consists of the following three fields: coauthor names, title words, and journal or conference name. Let $A^q = \{a_1^q, \ldots, a_U^q\}$ be the set of coauthor names appearing in D^q. We exclude q from A^q. Let $B^q = \{b_1^q, \ldots, b_V^q\}$ be the set of journal or conference names appearing in D^q. Further, let $W^q = \{w_1^q, \ldots, w_J^q\}$ be the set of title words appearing in D^q. We neglect the order of coauthor names and that of title words. Our aim is to disambiguate q by splittting D^q into disjoint clusters. In the ideal clustering, any two citation data from the same cluster correspond to the same full name, and any two citation data from different clusters correspond to different full names. We will omit the superscript q when no confusion arise.

3.1 Naive Bayes Mixture Model (NBM)

The naive Bayes mixture model (NBM) has one hidden variable. Let the set of values this hidden variable takes be $C = \{c_1, \ldots, c_K\}$. Each of these K values can be regarded as a cluster ID. K should be given as an input for parameter estimation. NBM generates each citation data as described below. First, a hidden variable value is randomly selected from C according to the multinomial distribution $P(c_k)$. Let the selected value be c_k. Second, coauthor names are randomly selected from A according to the multinomial distribution $P(a_u|c_k)$ which is determined by the selected hidden variable value c_k. Title words are also randomly selected from W according to the multinomial distribution $P(w_j|c_k)$ which is determined by the selected hidden variable value c_k. A journal name or a conference name is randomly selected from B according to the multinomial distribution $P(b_v|c_k)$ which is also determined by the hidden variable value c_k. In this paper, we assume that the number of coauthor names and the number of title words are given, and do not explicitly model these numbers as in [11]. Let o_{iu} be the number of coauthor names in d_i, and let c_{ij} be the number of title word w_j in d_i. Further, δ_{iv} is defined to be 1 if the journal or conference name of d_i is b_v and 0 otherwise. Then, the probability of generating d_i can be written as $P(d_i) = \sum_{k=1}^{K} P(c_k)P(d_i|c_k)$, where

$$P(d_i|c_k) = \prod_{u=1}^{U} P(a_u|c_k)^{o_{iu}} \prod_{j=1}^{J} P(w_j|c_k)^{c_{ij}} \prod_{v=1}^{V} P(b_v|c_k)^{\delta_{iv}}. \quad (1)$$

The probability of generating D is $P(D) = \prod_{i=1}^{I} P(d_i)$.

The E step of EM algorithm for NBM can be written as $P(c_k|d_i) = \bar{P}(d_i, c_k)/\sum_{k=1}^{K} \bar{P}(d_i, c_k)$ where $\bar{P}(d_i, c_k)$ is equal to $\bar{P}(c_k)\bar{P}(d_i|c_k)$. $\bar{P}(c_k)$ is a parameter value obtained in the previous M step. $\bar{P}(d_i|c_k)$ can be computed by using parameter values obtained in the previous M step with Equation 1. The M step of EM algorithm for NBM is as follows:

$$\begin{aligned}
P(c_k) &= \frac{\sum_{i=1}^{I} P(c_k|d_i)}{\sum_{k=1}^{K}\sum_{i=1}^{I} P(c_k|d_i)} \\
P(a_u|c_k) &= \frac{\sum_{i=1}^{I} P(c_k|d_i)o_{iu}}{\sum_{u=1}^{U}\sum_{i=1}^{I} P(c_k|d_i)o_{iu}} \\
P(b_v|c_k) &= \frac{\sum_{i=1}^{I} P(c_k|d_i)\delta_{iv}}{\sum_{v=1}^{V}\sum_{i=1}^{I} P(c_k|d_i)\delta_{iv}} \\
P(w_j|c_k) &= \frac{\sum_{i=1}^{I} P(c_k|d_i)c_{ij}}{\sum_{j=1}^{J}\sum_{i=1}^{I} P(c_k|d_i)c_{ij}}, \quad (2)
\end{aligned}$$

where $P(c_k|d_i)$ is obtained in the previous E step. In our experiment, 30 times iteration was enough for convergence. The cluster membership is determined by $\arg\max_k P(c_k|d_i)$ for each d_i. When k does not satisfy $k = \arg\max_k P(c_k|d_i)$ for any d_i, c_k corresponds to an empty cluster. Hence, the number of non-empty clusters can be less than K.

3.2 Two-Variable Mixture Model (TVM)

In this paper, we propose a new probabilistic model, called *two-variable mixture model (TVM)*. Let $Y = \{y_1, \ldots, y_S\}$ be the set of values the one hidden variable takes. Let $Z = \{z_1, \ldots, z_T\}$ be the set of values the other hidden variable takes. By combining these two types of values, we represent the cluster membership of citation data. TVM generates each citation data as follows. First, a value of the one hidden variable is randomly selected from Y according to the multinomial distribution $P(y_s)$. Let the selected value be y_s. Second, a value of the other hidden variable is randomly selected from Z according to the multinomial distribution $P(z_t|y_s)$. We denoted this value by z_t. The multinomial $P(z_t|y_s)$ is determined by y_s. Further, a journal or conference name is randomly selected from B according to the multinomial distribution $P(b_v|y_s)$ which is also determined by y_s. Third, title words are randomly selected from W according to the multinomial distribution $P(w_j|z_t)$. This multinomial is determined by the value z_t selected for the latter hidden variable. Finally, coauthor names are randomly selected from A according to the multinomial $P(a_u|y_s, z_t)$ which is determined by the value pair (y_s, z_t) of the two hidden variables. The generation order of the values of the two hidden variables is irrelevant to the generation of coauthor names. As for TVM, the probability of generating a citation data d_i can be written as $P(d_i) = \sum_{s=1}^{S}\sum_{t=1}^{T} P(y_s)P(z_t|y_s)P(d_i|z_t, y_s)$, where

$$\begin{aligned}
&P(d_i|z_t, y_s) \\
&= \prod_{u=1}^{U} P(a_u|z_t, y_s)^{o_{iu}} \prod_{j=1}^{J} P(w_j|z_t)^{c_{ij}} \prod_{v=1}^{V} P(b_v|y_s)^{\delta_{iv}}. \quad (3)
\end{aligned}$$

With respect to TVM, the E step of EM algorithm is $P(y_s, z_t|d_i) = \bar{P}(d_i, y_s, z_t)/\sum_{s=1}^{S}\sum_{t=1}^{T} \bar{P}(d_i, y_s, z_t)$ where $\bar{P}(d_i, y_s, z_t)$ is equal to $\bar{P}(y_s)\bar{P}(z_t|y_s)\bar{P}(d_i|z_t, y_s)$. $\bar{P}(y_s)$ and $\bar{P}(z_t|y_s)$ are parameter values obtained in the previous M step, and $\bar{P}(d_i|z_t, y_s)$ can be computed by using parameter values obtained in the previous M step with Equation 3.

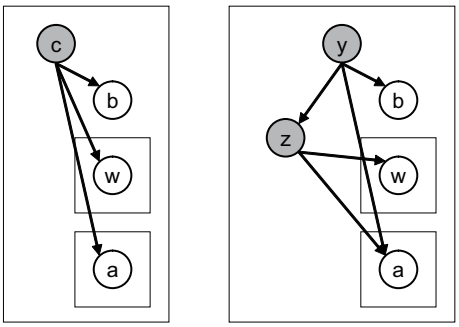

Figure 1: Graphical representations of NBM (right panel) and TVM (left panel).

The M step of EM algorithm for TVM is given by

$$P(y_s) = \frac{\sum_{t=1}^{T}\sum_{i=1}^{I} P(y_s, z_t|d_i)}{\sum_{s=1}^{S}\sum_{t=1}^{T}\sum_{i=1}^{I} P(y_s, z_t|d_i)}$$

$$P(z_t|y_s) = \frac{\sum_{i=1}^{I} P(y_s, z_t|d_i)}{\sum_{t=1}^{T}\sum_{i=1}^{I} P(y_s, z_t|d_i)}$$

$$P(b_v|y_s) = \frac{\sum_{t=1}^{T}\sum_{i=1}^{I} P(y_s, z_t|d_i)\delta_{iv}}{\sum_{v=1}^{V}\sum_{t=1}^{T}\sum_{i=1}^{I} P(y_s, z_t|d_i)\delta_{iv}}$$

$$P(w_j|z_t) = \frac{\sum_{s=1}^{S}\sum_{i=1}^{I} P(y_s, z_t|d_i)c_{ij}}{\sum_{j=1}^{J}\sum_{s=1}^{S}\sum_{i=1}^{I} P(y_s, z_t|d_i)c_{ij}}$$

$$P(a_u|y_s, z_t) = \frac{\sum_{i=1}^{I} P(y_s, z_t|d_i)o_{iu}}{\sum_{u=1}^{U}\sum_{i=1}^{I} P(y_s, z_t|d_i)o_{iu}} \quad (4)$$

where $P(y_s, z_t|d_i)$ is obtained in the previous E step. Also for TVM, 30 times iteration of E and M steps was enough for convergence. We regard $\arg\max_{(s,t)} P(y_s, z_t|d_i)$ as the ID of the cluster to which d_i belongs. There are ST possible pairs of the values of the two hidden variables. When a value pair (s, t) has no d_i satisfying $\arg\max_{(s,t)} P(y_s, z_t|d_i)$, the pair corresponds to an empty cluster. In the evaluation experient, we set $S = T$, because the preliminary experiments provide no interesting results for the cases $S \neq T$. TVM generates title words and a conference or journal name according to a value selected for one of the two hidden variables. Only coauthor names are generated according to a pair of selected values. Consequently, only coauthor names are used as a direct clue for citation data clustering, because cluster membership is determined by value pairs of the two hidden variables. Title words and a journal or conference name work only as an indirect clue. This construction of TVM is based on the intuition that coauthor names are the most important factor for author name disambiguation. This intuition is also shared by the previous studies [5][6].

We can have another construction of TVM by exchanging the roles of the two hidden variables. In this alternative model, $P(d_i)$ is equal to $\sum_s \sum_t P(z_t)P(y_s|z_t)P(d_i|z_t, y_s)$. Preliminary experiments provide no interesting differences between the original TVM and this alternative. Hence, we only use the TVM shown above in the experiment. Figure 1 presents graphical representations of NBM and TVM.

3.3 Smoothing and Annealing

In estimating parameters, we use two standard techniques: smoothing and annealing. We realize a smoothing by modifying Equation 2 as follows:

$$P(a_u|c_k) = (1-\gamma)\frac{\sum_i P(c_k|d_i)o_{iu}}{\sum_u \sum_i P(c_k|d_i)o_{iu}} + \gamma\frac{\sum_i o_{iu}}{\sum_u \sum_i o_{iu}}$$

$$P(b_v|c_k) = (1-\gamma)\frac{\sum_i P(c_k|d_i)\delta_{iv}}{\sum_v \sum_i P(c_k|d_i)\delta_{iv}} + \gamma\frac{\sum_i \delta_{iv}}{\sum_v \sum_i \delta_{iv}}$$

$$P(w_j|c_k) = (1-\gamma)\frac{\sum_i P(c_k|d_i)c_{ij}}{\sum_j \sum_i P(c_k|d_i)c_{ij}} + \gamma\frac{\sum_i c_{ij}}{\sum_j \sum_i c_{ij}} \quad (5)$$

where we linearly mix the clusterwise probability and the background probability. As for Equation 4, we also use the linear mixture of the clusterwise probability and the background probability. This kind of smoothing is required due to the fact that citation data are quite sparse. γ was set to 0.5 after an appropriate tuning.

Further, we apply an annealing method proposed by Rose et al. [10] to prevent our EM algorithm from being quickly caught by local maxima. We modify the E step for NBM as $P(c_k|d_i) = \{\bar{P}(d_i, c_k)\}^\beta / \sum_k \{\bar{P}(d_i, c_k)\}^\beta$ and that for TVM as $P(y_s, z_t|d_i) = \{\bar{P}(d_i, y_s, z_t)\}^\beta / \sum_{s=1}^{S}\{\sum_{t=1}^{T} \bar{P}(d_i, y_s, z_t)\}^\beta$. β is initialized to 0.5, and is powered to 0.8 for every iteration. As the number of iterations increases, β gets near to 1.0, and the differences of the probabilities $\bar{P}(d_i, c_k)$ or $\bar{P}(d_i, y_s, z_t)$ come to stand out.

Table 1: Abbreviated names used in the experiment.

Abbr. name	# of full names	# of data	Abbr. name	# of full names	# of data
s.lee	161	971	j.park	68	376
j.lee	134	892	y.liu	67	313
j.kim	129	769	c.wang	65	360
j.wang	112	575	s.chen	64	297
s.kim	108	598	z.wang	63	142
y.wang	101	533	j.liu	63	406
h.kim	100	506	h.li	63	220
h.lee	99	346	j.li	61	314
x.wang	86	322	j.zhang	60	308
j.chen	86	487	s.li	59	242
s.wang	84	274	z.li	56	210
y.zhang	83	391	j.wu	56	320
y.chen	81	525	j.lin	55	196
k.lee	81	380	z.zhang	54	255
h.wang	79	380	s.liu	54	122
y.li	76	261	h.liu	54	197
c.lee	75	468	d.kim	53	304
y.kim	74	373	y.yang	51	250
h.chen	74	419	x.liu	51	187
x.zhang	72	287	c.chen	51	462
y.lee	71	385	m.lee	50	309
k.kim	71	330	l.wang	50	253
x.li	69	315	j.yang	50	301
s.park	69	379			

4. EVALUATION EXPERIMENT

4.1 Experiment Procedure

In the evaluation experiment, we used a citation dataset published by the DBLP bibliographic database [2]. We used the data file `dblp20040213.xml.gz`, because this version is kept uploaded at the DBLP Web site with no modification for a long period of time. First, we removed the citation data

lacking any one of the following three data fields: coauthor names, title words, and journal or conference name. We also removed the citation data originally including coauthor names with their abbreviated first names, because we have no correct answer, i.e., no corresponding full names, for such citation data. Then, we removed data fields other than the above three fields from the remaining citation data and abbreviated all first names to their initials. Among the abbreviated author names in the resulting citation data, we selected 47 names in Table 1. To each of these 47 names, more than or equal to 50 full names correspond. In Table 1, the first and the fourth columns show the abbreviated author names, the second and the fifth columns show the number of corresponding full names, and the third and the sixth columns show the number of citation data including each abbreviated name. As a preprocessing, we removed a standard set of stop words from title words and applied a porter stemmer [3] to the remaining title words.

With respect to each abbreviated name in Table 1, we conducted an evaluation experient in the procedure described below. For example, suppose that we conduct an experiment for "S. Lee". First, we collect all citation data including "S. Lee" to make a citation data set D. Second, we subdivide D into disjoint clusters by the following three methods. a) Apply the naive Bayes mixture model to D. We simply denote this disambiguation method by *NBM*. b) Remove title words and journal or conference name from every citation data in D, and apply the naive Bayes mixture model to this modified D. We denote this method by *NBMa*, because we only use author names. c) Apply the two-variable mixture model to D. We denote this method by *TVM*. For any of these three methods, we randomly initialized model parameter values and executed EM algorithm from 20 different sets of initial parameter values. Consequently, we have 20 results for each of NBM, NBMa, and TVM. We also used k-means as a baseline method. We ran k-means algorithm 20 times from randomly initialized cluster assignments. The feature vector for k-means includes the frequencies of coauthor names, title words, and journal or conference name.

When we assumed that the true number of clusters was not known, the number K of clusters was set to 256 for all query names. As for TVM, we set $S = T = 16$. Then, $ST = K$ holds, and we set the same cluster granularity for NBM, NBMa, and TVM. On the other hand, when we assumed that the true number of clusters was known, we set $S = T = \lceil \sqrt{\text{true number of clusters}} \rceil$ for TVM, and set $K = ST$ for NBM and NBMa. Also for this case, we set the same cluster granularity for NBM, NBMa, and TVM. When we used "S. Lee" as a query name, the actual execution time of 30 iterations of EM algorithm was about 19 seconds for NBM, 16 seconds for TVM, and 6 seconds for NBMa, where all data were loaded on the main memory, and the CPU was Intel Xeon 3.20GHz.

4.2 Evaluation Method

We evaluated clustering results as follows. Suppose that we have a clustering $\mathcal{G}$ of D. For each cluster $G \in \mathcal{G}$, we can obtain the dominating full name $f(G)$ by checking the full names appearing in the original citation data. Let $N_{pos}(G)$ be the number of the citation data in G which correspond to the dominating full name $f(G)$. Let $N_{size}(G)$ be the size of G. Further, let $N_{cor}(G)$ be the number of the citation data in D which correspond to the dominating full name $f(G)$.

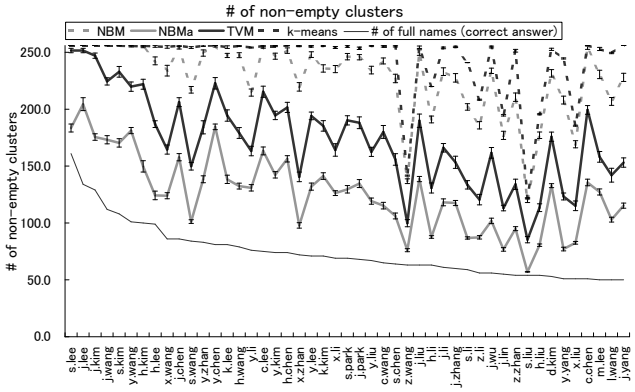

Figure 2: True number of clusters (lowermost graph) and number of non-empty clusters.

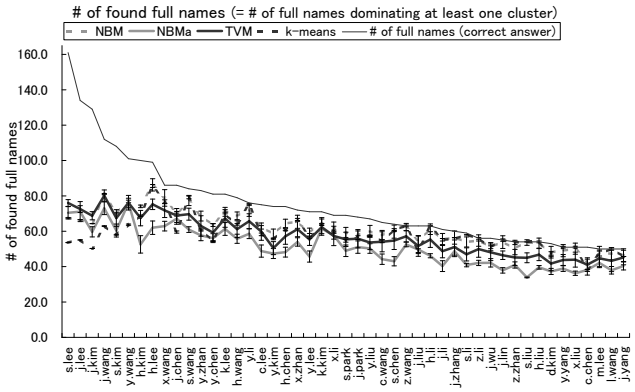

Figure 3: True number of clusters (uppermost graph) and number of found full names.

Then, the precision of G is $N_{pos}(G)/N_{size}(G)$, and the recall of G is $N_{pos}(G)/N_{cor}(G)$. We can obtain an averaged precision/recall of $\mathcal{G}$ in two different ways. *Macroaveraged* precision/recall is computed based on a simple sum of the precisions/recalls of all clusters. On the other hand, *microaveraged* precision/recall is computed based on a weighted sum of the precisions/recalls of all clusters. To be precise, the macroaveraged precision $P_{mac}(\mathcal{G})$ is defined to be $\sum_{G\in\mathcal{G}} \frac{N_{pos}(G)}{N_{size}(G)}/|\mathcal{G}|$ and the macroaveraged recall $R_{mac}(\mathcal{G})$ is $\sum_{G\in\mathcal{G}} \frac{N_{pos}(G)}{N_{cor}(G)}/|\mathcal{G}|$. The microaveraged precision $P_{mic}(\mathcal{G})$ is $\sum_{G\in\mathcal{G}} N_{pos}(G)/\sum_{G\in\mathcal{G}} N_{size}(G)$ and the microaveraged recall $R_{mic}(\mathcal{G})$ is $\sum_{G\in\mathcal{G}} N_{pos}(G)/\sum_{G\in\mathcal{G}} N_{cor}(G)$. We compute these four evaluation measures with respect to 20 clustering results obtained for each of the three disambiguation methods: NBM, NBMa, and TVM. Then, we compute the mean and the standard deviation of 20 values for each of the four evaluation measures. We regard these means and standard diviations as our evaluation of each disambiguation method with respect to a given query name.

While there is a study contending that microaveraged precision is identical with microaveraged recall for the clustering result evaluation [8], our definition of R_{mic} is different from that of P_{mic}. Both of P_{mic} and R_{mic} are equal to 1 for an ideal clustering result. Our P_{mic} is identical with "disambiguation accuracy" in [6] for clustering result evaluation.

Table 2: Evaluation results under the assumption that the true number of clusters is unknown.

method	P_{mic}	P_{mac}	R_{mic}	R_{mac}	F_{mic}	F_{mac}
NBMa	0.7034	0.9026	0.1314	0.3586	0.2140	0.5085
NBM	0.8501	0.8890	0.0741	0.2488	0.1343	0.3859
TVM	0.7807	0.8653	0.0985	0.2995	0.1710	0.4408
k-means	0.7686	0.8055	0.0612	0.2133	0.1115	0.3340

4.3 Results of Evaluation Exepriment

4.3.1 Non-empty clusters and dominating full names

We present the number of non-empty clusters for each of the 47 query names in Figure 2 with respect to the case where we assume that the true number of clusters is not known. In this case, we set $K = 256$ (for NBM and NBMa) and $S = T = 16$ (for TVM) for all 47 query names. Each number of non-empty clusters is the average of 20 numbers obtained by executing an EM algorithm from 20 randomly initialized sets of parameter values. The marker shows plus /minus one standard deviation of these 20 numbers of non-empty clusters. $\times$ stands for the true number of clusters, i.e., the number of full names which can be abbreviated to each query name. In case of NBMa, the numbers of non-empty clusters are close to the true numbers. On the other hand, NBM results in oversegmentation. The numbers of clusters of TVM lie halfway between those of NBMa and those of NBM for all query names. Since NBMa only uses the coauthor name field, the variety of citation data is largely reduced. As for NBM, the variety of citation data increases due to the title word field where a wide variety of words appear. Moreover, the generation of title words in NBM depends on a hidden variable taking one value from 256 values. Consequently, citation data are likely to be dispersed into many different clusters. In contrast, the generation of title words in TVM depends on a hidden variable taking one value from only 16 values. This restriction of the number of hidden variable values with respect to the generation of title words results in a moderate cluster granularity given by TVM. Figure 2 also shows that the clusters given by k-means method are most severely oversegmented.

Figure 3 provides the number of full names dominating at least one cluster, i.e., the number of full names clustering algorithms could find. Each marker shows plus/minus one standard deviation of 20 numbers of found full names for 20 executions of EM algorithm. The graph of NBMa is in the lowermost position for many query names. That is, NBMa missed more full names than NBM, TVM, and k-means for many query names. This means that NBMa often results in a clustering where many clusters are dominated by the same full names. Therefore, the fact that NBMa can provide cluster numbers close to the true numbers is not necessary a good result.

4.3.2 Evaluation by precision and recall

We evaluate the clustering results by precision and recall. Figure 4 presents P_{mic} for each query name. P_{mic} is likely to be large when the clusters are oversegmented, and is strongly affected by the precisions of large clusters. Figure 5 presents R_{mic}, which is likely to be small when the clusters are oversegmented, and is strongly affected by the recalls of clusters dominated by the full names to which many citation data correspond. Figure 6 presents P_{mac}, which is likely to be large when the clusters are oversegmented just as P_{mic}, but is equally affected by the precisions of all clusters. Figure 7 presents R_{mac}, which is likely to be small when the clusters are oversegmented just as R_{mic}, but is equally affected by the recalls of all clusters. In all these four figures, the marker shows plus/minus one standard deviation of 20 values of P_{mic}, R_{mic}, P_{mac}, and R_{mac}, respectively, obtained by executing an EM algorithm from 20 randomly initialized sets of parameter values.

While NBM and NBMa show no remarkable differences in P_{mac}, NBM is superior to NBMa in P_{mic}. However, as for R_{mic} and R_{mac}, NBMa gives better results than NBM. Since NBM uses all of the three data fields: coauthor names, title words, and journal or conference name, the input data shows a wider variety than that used by NBMa, and, consequently, NBM is likely to result in oversegmentation and to provide lower recalls. TVM shows halfway results between NBM and NBMa with respect to P_{mic}, R_{mic}, and R_{mac}. We can conclude that TVM gives a good balance between precision and recall. This is because the title word field, which shows the widest variety among the three fields, is generated depending only on one of the two hidden variables in TVM. This model structure of TVM reduces oversegmentation.

The problem that the recall is low is shared by all clustering methods in our experiment. R_{mac}s for most abbreviated names nearly range from 0.25 to 0.5 as depicted in Figure 7. Roughly speaking, this result corresponds to the situation that full names are scattered in from two to four clusters in average for most abbreviated names. On the other hand, R_{mic}s for most abbreviated names nearly range from 0.1 to 0.2 in Figure 5. This result corresponds to the situation that the full names to which many citation data correspond are scattered in from five to ten clusters. These two situations are not so bad as long as the precisions are large enough and the number of found full names is close to the actual number of full names. As for some query names (e.g. "Z. Wang" and "S. Liu") the recalls are large, and Figure 2 shows that the number of non-empty clusters is very close to the true number for all three methods. We can conclude that the similarity among citation data was correctly explained by the naive Bayes mixture model or by the two-variable mixture model for these query names.

Table 2 shows the averages of P_{mic}s, R_{mic}s, P_{mac}s, and R_{mac}s taken over all abbreviated names. The sixth column (resp. the seventh column) includes the harmonic mean F_{mic} (resp. F_{mac}) of P_{mic} and R_{mic} (resp. P_{mac} and R_{mac}). When we do not need to mind the fact that many full names cannot be found, NBMa is the most favorable. However, if we would like to find as many full names as possible, we should choose TVM. Moreover, while TVM is not the best with respect to both F_{mic} and F_{mac}, we think that improving recalls by sacrificing precisions is not a good strategy when it is intrinsically difficult to improve recalls as in our case. Table 2 also shows that k-means gives precisions even lower than TVM. As k-means results in the most severe oversegmentation (cf. Figure 2), we can say that k-means seems not suitable for our problem.

Table 3 summarizes the evaluation results when we know the true number of clusters. For TVM, we set $S = T = \lceil\sqrt{\text{true number of clusters}}\,\rceil$, and, for NBM and NBMa, we set $K = \lceil\sqrt{\text{true number of clusters}}\,\rceil^2$. By comparing Table 3 with Table 2, we can find that the precision largely decreases and that the recall largely increases. This is because oversegmentation is reduced by using the true num-

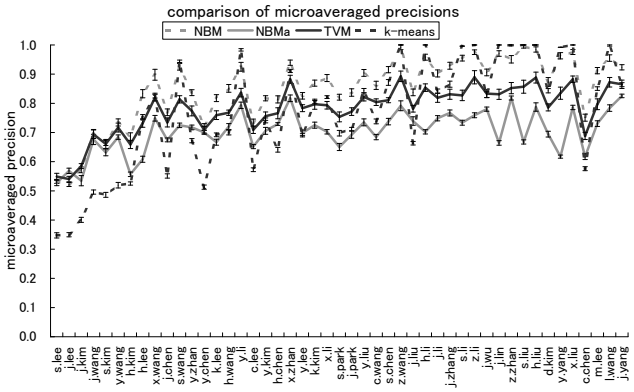

Figure 4: Comparision of microaveraged precisions.

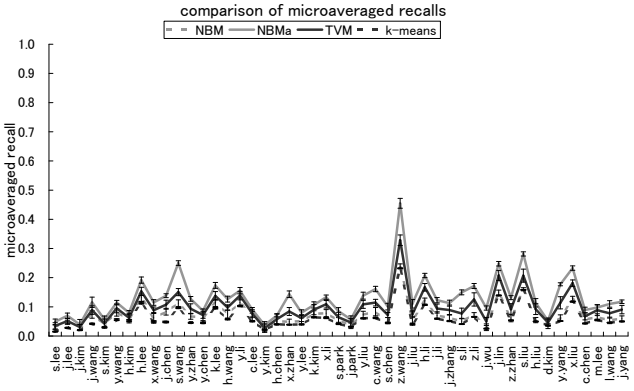

Figure 5: Comparision of microaveraged recalls.

ber of clusters in determining S, T or K. Table 3 agrees with Table 2 in the fact that TVM gives intermediate results between NBM and NBMa. However, as is depicted in Figure 8, the number of found full names when we assume that we know the true number of clusters is far smaller than that when we assume that we do not know the true number (cf. Figure 3). That is, as for the number of full names which can be found by name disambiguation, the fact that we know the true number of clusters is not a favorable factor. As for k-means method, Table 3 shows that it is inferior to all other methods with respect to all evaluation criteria.

4.4 Evaluation results for another set of abbreviate names

To increase the completeness of our evaluation experiment, we conducted the same evaluation with respect to another set of abbreviated names presented in Table 4. To any of these 53 names, from 30 to 40 full names correspond. That is, by using these abbreviated names as a query name, we can check if TVM can give a good balance between precision and recall for a query name to which only a moderate number of full names can be abbreviated. Table 5 summarizes the evaluation results for these query names when we assume that the true number of clusters is not known, where we set $K = 64$ (for NBM and NBMa) and $S = T = 8$ (for TVM) for all query names. Table 6 summarizes the results when we assume that the true number of clusters is known, where we set $S = T = \lceil \sqrt{\text{true number of clusters}} \rceil$

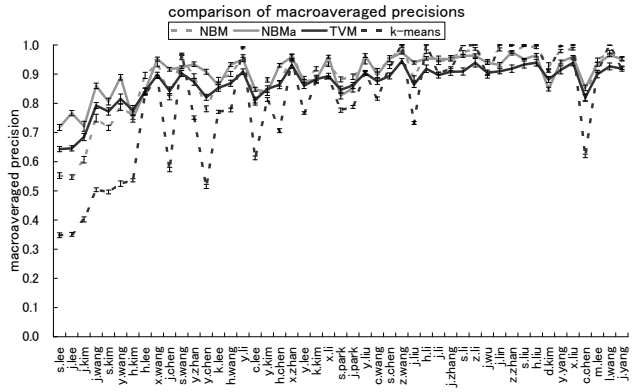

Figure 6: Comparision of macroaveraged precisions.

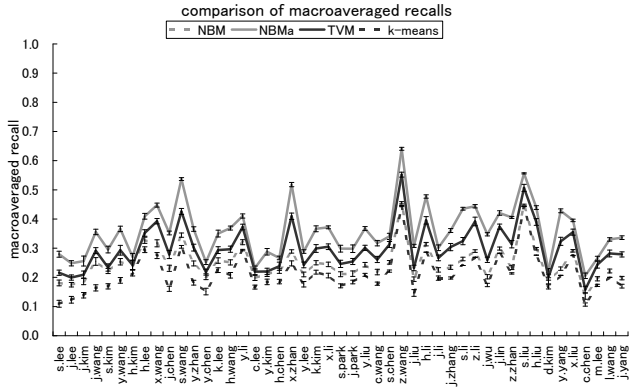

Figure 7: Comparision of macroaveraged recalls.

Table 3: Evaluation results under the assumption that the true number of clusters is known.

method	P_{mic}	P_{mac}	R_{mic}	R_{mac}	F_{mic}	F_{mac}
NBMa	0.6002	0.7729	0.1701	0.4051	0.2574	0.5282
NBM	0.5208	0.5277	0.1163	0.2804	0.1856	0.3632
TVM	0.5469	0.6358	0.1367	0.3278	0.2129	0.4294
k-means	0.3555	0.3517	0.0731	0.1916	0.1174	0.2424

and $K = ST$. Since any abbreviated name in Table 4 corresponds to only a moderate number of full names, both precision and recall increase in comparison with the abbreviated names in Table 1 (cf. Table 2 and Table 3). However, also for the abbreviated names in Table 4, TVM gives halfway results between NBM and NBMa. That is, our observation is confirmed again. k-means is inferior to other methods with respect to all evaluation measures for these query names.

5. CONCLUSION

In this paper, we provided a method for correctly assigning each citation data to its true author by disambiguating an abbreviated author name used as a query. First, we collected all citation data including an abbreviated author name used as a query. Then, we splitted the obtained set of citation data into disjoint clusters by the three methods: NBM, NBMa, and TVM. NBM is based on the naive Bayes mixture model and uses all three data fields of citation data, i.e., coauthor names, title words, and journal or conference name. NBMa is also based on the naive Bayes

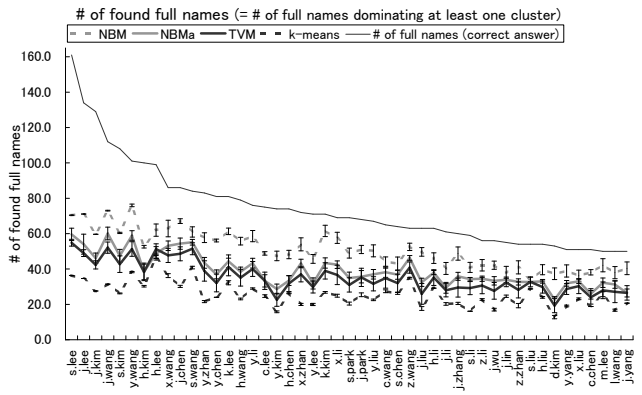

Figure 8: True number of clusters (uppermost graph) and number of found full names under the assumption that the correct number of clusters is known.

mixture model and only uses the coauthor name field. TVM is based on a new probabilistic model, called the two-variable mixture model, and uses all three data fields. We evaluated clustering results by four measures: microaveraged precision/recall and macroaveraged precision/recall. When we assume that the true number of clusters is not known, the recall was quite small due to oversegmentation. This problem is shared by all of NBM, NBMa, and TVM. While NBMa provided cluster numbers close to the true numbers, NBMa also failed to find many full names. In contrast, TVM finds as many full names as NBM and achieves a good balance between precision and recall in comparison with NBM and NBMa. Although we tested k-means as a baseline method, it was inferior to NBM, NBMa, and TVM with respect to every evaluation measure in almost all experiment settings.

However, the disambiguation accuracy was not satisfying as a whole. We should take into account some options to especially improve recall. As we discussed in Section 1, it seems expensive to use additional information sources about authors, journals, conferences, and relevant research fields, because we should constantly update such additional data and should keep them reliable and consistent. In contrast, we can easily know from which paper each citation data is taken. This kind of information can be obtained in the course of gathering citation data, and thus requires only moderate effort. Our main future work is to provide a probabilistic model incorporating dependencies between entities appearing in the articles which are a source of citation data.

Table 4: Another set of abbreviated names.

Abbr. name	# of full names	# of data	Abbr. name	# of full names	# of data
w.wang	40	234	j.choi	34	191
l.zhang	40	302	h.yang	34	174
l.chen	40	252	y.zhou	33	104
j.zhou	40	120	s.yu	33	130
g.wang	40	137	j.hu	33	115
c.kim	40	197	z.yang	32	67
b.lee	40	178	y.xu	32	179
l.li	39	141	y.choi	32	130
c.wu	39	306	w.zhang	32	165
z.liu	38	223	w.li	32	208
s.cho	38	171	s.kang	32	148
l.liu	38	216	d.liu	32	103
h.chang	38	106	c.huang	32	223
c.li	38	190	y.yu	31	121
x.huang	37	90	t.kim	31	185
j.chang	37	174	s.huang	31	164
y.zhao	36	94	c.park	31	140
t.watanabe	36	133	y.zhu	30	106
c.chang	36	374	y.lu	30	111
y.lin	35	250	t.tanaka	30	71
y.chang	35	198	t.nguyen	30	125
x.yang	35	92	q.li	30	169
t.wang	35	102	g.zhang	30	91
s.choi	35	155	d.wang	30	171
g.lee	35	114	c.zhang	30	160
k.chen	34	146	a.gupta	30	377
j.liu	34	102			

Table 5: Evaluation results for the name set in Table 4 under the assumption that the true number of clusters is unknown.

method	P_{mic}	P_{mac}	R_{mic}	R_{mac}	F_{mic}	F_{mac}
NBMa	0.7089	0.8921	0.1920	0.4127	0.2886	0.5574
NBM	0.7118	0.7448	0.1268	0.3033	0.2102	0.4274
TVM	0.7110	0.8047	0.1566	0.3499	0.2469	0.4813
k-means	0.5494	0.5684	0.0952	0.2349	0.1581	0.3265

Table 6: Evaluation results for the name set in Table 4 under the assumption that the true number of clusters is known.

method	P_{mic}	P_{mac}	R_{mic}	R_{mac}	F_{mic}	F_{mac}
NBMa	0.6553	0.8205	0.2222	0.4355	0.3173	0.5622
NBM	0.5834	0.5948	0.1582	0.3208	0.2394	0.4099
TVM	0.6092	0.7002	0.1850	0.3659	0.2715	0.4735
k-means	0.4317	0.4301	0.1071	0.2275	0.1632	0.2842

6. REFERENCES

[1] http://citeseer.ist.psu.edu/

[2] http://www.informatik.uni-trier.de/~ley/db/

[3] http://www.tartarus.org/~martin/PorterStemmer/

[4] X. Dong, A. Halevy, and J. Madhavan, Reference Reconciliation in Complex Information Spaces, in *Proc. of SIGMOD2005*, pp. 85-96, 2005.

[5] H. Han, C. L. Giles, H. Zha, C. Li, and K. Tsioutsiouliklis, Two Supervised Learning Approaches for Name Disambiguation in Author Citations, in *Proc. of JCDL2004*, pp. 296-305, 2004.

[6] H. Han, W. Xu, H. Zha, and C. Lee Files, A Hierarchical Naive Bayes Mixture Model for Name Disambiguation in Author Citations, in *Proc. of SAC'05*, pp. 1065-1069, 2005.

[7] H. Han, H. Zha, and L. Giles, Name disambiguation in author citations using a k-way spectral clusering method, in *Proc. of JCDL2005*, pp. 334-343, 2005.

[8] A. Hotho, S. Staab, and G. Stumme, WordNet improves text document clustering, in *Proc. of SIGIR 2003 Semantic Web Workshop*, 2003.

[9] D. V. Kalashnikov, S. Mehrotra, and Z. Chen, Exploiting Relationships for Domain-Independent Data Cleaning, in *Proc. of the SIAM International Conference on Data Mining*, 2005.

[10] K. Rose, E. Gurewitz, and G. Fox, A Deterministic Annealing Approach to Clustering, *Pattern Recognition Letters*, Vol. 11, pp. 589-594, 1990.

[11] K. Nigam, A. McCallum, S. Thrun, and T. M. Mitchell, Text Classification from Labeled and Unlabeled Documents using EM, *Machine Learning*, Vol. 39, No. 2/3, pp. 103-134, 2000.