Scientometrics (2014) 101:1955–1972
DOI 10.1007/s11192-014-1283-x

Unsupervised author disambiguation using Dempster–Shafer theory

Hao Wu · Bo Li · Yijian Pei · Jun He

Received: 12 December 2013 / Published online: 20 April 2014

Abstract The name ambiguity problem presents many challenges for scholar finding, citation analysis and other related research fields. To attack this issue, various disambiguation methods combined with separate disambiguation features have been put forward. In this paper, we offer an unsupervised Dempster–Shafer theory (DST) based hierarchical agglomerative clustering algorithm for author disambiguation tasks. Distinct from existing methods, we exploit the DST in combination with Shannon's entropy to fuse various disambiguation features and come up with a more reliable candidate pair of clusters for amalgamation in each iteration of clustering. Also, some solutions to determine the convergence condition of the clustering process are proposed. Depending on experiments, our method outperforms three unsupervised models, and achieves comparable performances to a supervised model, while does not prescribe any hand-labelled training data.

Keywords Author disambiguation · Dempster–Shafer theory of evidence · Hierarchical clustering · Unsupervised

Introduction

With increasing electronic publication on the web, researchers almost fully rely on digital libraries to retrieve relevant works. However, many digital libraries (e.g. Google Scholar, DBLP, Microsoft Academic Search) cannot correctly auto-index the publication records of authors because of the name ambiguity problem. An author may have multiple names and multiple authors may have the same name caused by name variations. This problem

H. Wu (✉) · B. Li · Y. Pei
School of Information Science and Engineering, Yunnan University, Kunming 650091, China
e-mail: haowu@ynu.edu.cn

J. He
Nanjing University of Information Science and Technology, Nanjing 210044, China

seriously affects the performance of scholar finding (Wu et al. 2009), citation analysis (Strotmann and Zhao 2012), database integration (Kalashnikov and Mehrotra 2006), social network analysis (Strotmann et al. 2009) and other related research fields. To attack the issue, many solutions based on different principles, such as supervised models and unsupervised models have been put forward (Ferreira et al. 2012; Smalheiser and Torvik 2009). Simultaneously, a variety of disambiguation features, such as coauthors, titles/content of papers, topics of articles, emails/affiliations, web correlations, etc, have been investigated deeply.

Although various effective disambiguation methods combined with rich disambiguation features have been proposed, there still remain many issues needed to be further investigated. One issue may be raised by end users is that they may want to complete disambiguation task by themselves as records of publications have been requested and obtained from the digital libraries. From this perspective, it is clear that we need to examine a simple yet effective method. Many of the most effective models for author disambiguation are mainly based on supervised machine learning techniques, since they have the advantage of being able to combine a larger pool of heterogeneous data sources in an optimized way. Supervised techniques require manually hand-labeled training data, and sophisticated technology makes them uneasy to be carried out and used by end users. Unsupervised heuristic is a more viable option for us, however, how to fuse heterogeneous information about various features becomes a new challenge for developing such models. For instance, in a bottom-up clustering model, a candidate pair of clusters most likely belongs to an individual author have to be chosen and merged by using all available features in each iteration. Although the features can be fused into a single evidence using a linear-weighted manner (for example, regression analysis), it is straightforward that when using fusion techniques there will be an increase in conflicting information from different sources of evidence, considering that available features usually contain an amount of noise in author disambiguation task. The increasing uncertainty may affect the selection of a candidate, thus degrade the clustering accuracy. Fortunately, the Dempster–Shafer theory (Shafer 1976) can help us solve this problem.

The Dempster–Shafer theory (DST) is a mathematical theory of evidence (Shafer 1976). It provides an opportunity to combine evidence from different sources and arrive at a degree of belief to support an individual or a set of hypotheses by considering all the available evidence. In a finite discrete space, the DST can be understood as a generalization of probability theory. However, dissimilar from the traditional probability theory where the probabilities have to be associated with individual atomic hypotheses, DST allows to associate measures of uncertainty with sets of hypotheses. This way enables the theory to distinguish between uncertainty and ignorance (Lucas and Van Der Gaag 1991). The DST is often used as a method of sensor fusion, however, its applications in other fields, such as information retrieval (Lalmas and Ruthven 1998; Moreira and Wichert 2013; Ruthven and Lalmas 2002), and e-business model (Yu et al. 2005), have illustrated that it can provide a better reasoning process and be extended to other domains as well.

Inspired by these, we propose a novel unsupervised method leveraged by the DST of evidence for the author disambiguation task. In our approach, we first treat each high-level disambiguation feature as a dimension of evidence, then utilize the DST to fuse these evidence and come up with a more reliable candidate couple of clusters for amalgamation in every iteration of a bottom-up clustering process. Specially, we use Shannon's entropy to automatically uncover the importance of discrete dimensional evidence in the fusion of evidence. We also present some solutions to determine the optimal convergence condition of the clustering process. Depending on experiments, our method outperforms three unsupervised models, and achieves comparable performances to a supervised model, while does not require any hand-labelled training data and can be easily extended to above-mentioned disambiguation scenario.

The rest of the paper is structured as follows. Section 2 reviews related works. Section 3 presents our methods, and Sect. 4 describes the experimental results. In Sect. 5, we present our conclusions and suggest the direction of future work.

Related works

Automatic name disambiguation methods proposed in the literatures are usually based on supervised or unsupervised techniques (Ferreira et al. 2012; Smalheiser and Torvik 2009). Supervised methods learn from a training set containing pre-labeled examples a "model" to either predict the author of a publication or to determine if two publications are owned by the same author. The derived model can be then applied to a set of unseen instances. In the former case, the records in the training and test sets represent publications, while in the latter, the records correspond to comparisons between two publications. Supervised methods usually exploit various machine learning techniques to combine a large pool of heterogeneous data sources in an optimized way (Culotta et al. 2007). The most representative approaches of this kind include generative models, such as Naive Bayes probability model (Han et al. 2004) and Hidden Markov Random Field (HMRF) (Tang et al. 2012); discriminative models, such as Support Vector Machine (SVM) (Han et al. 2004) and logistic regression (Gurney and Horlings 2012); hybrid models, such as random Forest (Treeratpituk and Giles 2009). However, supervised methods require manually hand-labeled training data and to achieve higher performance, more sophisticated models need to be developed.

In contrast, unsupervised methods usually exploit similarities between attributes of the publications (by means of predefined similarity functions), to group those publications that likely belong to the same author. And such works focus on mining various strong features and their combinations, such as the characteristics of author name (Milojevic 2013; Torvik et al. 2005), coauthors (Fan et al. 2011; Kang et al. 2009; Cota et al. 2010; Velden et al. 2011), topics of articles (Song et al. 2007), web correlations (Tan et al. 2006; Yang et al. 2008), citations (Levin et al. 2012; McRae-Spencer and Shadbolt 2006), emails/affiliations (Wu and Ding 2013), titles/content of papers and so on. To group publication records, a most common strategy is to use a clustering methodology, for example, hierarchical clustering (Song et al. 2007; Tan et al. 2006; Yang et al. 2008; Yin et al. 2007) and density-based clustering (Han et al. 2005; Huang and Seyda Ertekin 2006). However, a key issue facing the clustering process is how to synthesize heterogeneous features into a single dimensional evidence. For this, many works take learning-based approaches to fit diverse feature values in a linear-weighted manner. For instance, Huang and Seyda Ertekin (2006) experimented with SVMs to learn pairwise similarities of papers, then apply the DBSCAN-based clustering strategy (Ester et al. 1996) to generate clusters of papers. Yin et al. (2007) proposed a method called DISTINCT, which also utilizes SVMs to weigh different types of linkages during the clustering process. Different from the work (Huang and Seyda Ertekin 2006), DISTINCT takes a set of distinguishable objects in the database as the training set without seeking for manually labeled data. Nevertheless, such methods for fusing evidence derived from numerous features cannot handle uncertainty, thus may affect the clustering accuracy, considering that incoming information provides uncertain and conflicting evidence.

Beyond the two main types of solutions, some works are intention to develop a hybrid disambiguating model. Ferreira et al. (2010) proposed a self-taught approach for author disambiguation task. They first use a feature of co-authors to obtain some distinguishable seed clusters, then employ an association rule based method to assign the remaining clusters of papers to these seed clusters. A more contemporary direction is to integrate user feedback into existing models to fulfill interactive author disambiguation. Wang et al. (2011) proposed a pairwise factor graph model (PFGM) grounding on the HMRF for active author disambiguation. Authors develop several strategies to select the most uncertain disambiguation results, then utilize users' corrections on these results to retrain the PFGM model to improve disambiguation performance. Ferreira et al. (2012) propose improving their previous work (Ferreira et al. 2010) by exploiting user relevance feedback. Similar to the work (Wang et al. 2011), they also select a very small portion of the publication records mostly unsure about the correct authorship and ask the administrators for labeling them. This feedback is used to improve the effectiveness of the whole process. Experiments have proved interactive model is a promising way for author disambiguation tasks.

Be similar to existing works, our proposed method falls in the unsupervised category. But different from similar works (Huang and Seyda Ertekin 2006; Yin et al. 2007), we use the DST to fuse features with different types and handle the uncertainty during the clustering process. Also, we propose a quite distinct clustering algorithm to generate clusters of papers. In addition, our proposed framework enables utilizing users' feedbacks by adding only a dimension of evidence. This would be helpful to develop online disambiguation tools.

The framework for author disambiguation

Features and correlation measures

Affiliation is a strong feature to imply whether two papers belong to the same author or not. If two papers simultaneously share both coauthors and affiliations, then the answer is pretty certain. To estimate the pairwise similarity of affiliations, we use the *Jaccard Coefficent (JC)* and the *Levenshtein distance (LD)*. Given two affiliations' strings t_i and t_j, $JC(t_i, t_j)$ and $LD(t_i, t_j)$, respectively, measures the similarity between t_i and t_j in the word-level and the character-level. We use the combination of two metrics as $JC(t_i, t_j) * LD(t_i, t_j)$ to obtain the final score. Since affiliation information is not always available in the metadata of papers, we may exploit another feature-*CoAffOccur* [proposed in (Wang et al. 2011)] to supply the correlation estimation between the two papers. The information of CoAffOccur can be obtained by processing the first pages of documents and the feature takes a boolean value to express whether the contents of the two papers contain a same affiliation or not.

Venue Authors with the same name may work in different fields, thus publishing in different venues. If two disjoint sets of papers share many venues, it implies they belong to a same research field. The venue feature often contains a lot of noise due to name abbreviations and changes. For this, we take the approach same to the *Affiliation* to estimate the pairwise correlations of venues.

ContentSim This feature evaluates the similarity of content between two papers. It can be calculated based on the title and the abstract of papers. We can use the cosine measure and *JC*-based measure to obtain the word-level similarity of the two papers. Also, other advanced technologies, such as topic models (Song et al. 2007), can allow us to obtain the topic-level similarity of the two papers. All these information can be viewed as sub-features of ContentSim, and be combined together using a simple data fusion formula.

CoAuthor Similar to the Affiliation, if two papers have a same coauthor(except the target author a), both of them almost definitely belong to the target author. The value of the CoAuthor feature is quantified as the number of shared coauthors in the two papers except a.

Citation It is possible that an author cites his own paper, but seldom cites a paper authored by another author with the same name (McRae-Spencer and Shadbolt 2006; Wang et al. 2011). If there exists citation relation in the collection of papers with a same name label, it must be helpful to distinguish the papers. The value of citation feature is taken as 1, if paper p_i cites paper p_j. Note that, other citation information, such as co-citation and co-cited, can also be classified into this kind of feature if they are available.

WebCorrelation Currently, scholarly papers are usually listed on their authors' Web pages (e.g. Homepage and Blog). If two papers occur in the same Webpages owned by one author, it is highly likely that the two papers belong to this author. Web-based correlation has been proposed in author disambiguation tasks, and has proved a useful feature to enhance disambiguation performance (Tan et al. 2006; Yang et al. 2008; Wang et al. 2011). Also, many search utilities, such as Google and Yahoo, support to extract this information from the Web. The feature value can be quantified as the co-occurrence times of the two papers in author's Web pages.

In short, these high-level features (shown in Table 1) have been deeply investigated and widely applied in author disambiguation tasks, thus they can be seen as fundamental evidence for author disambiguation. Moreover, since coming from different sources, such features can be considered as relatively independent evidence, and are quite suitable for fusion using Dempster–Shafer theory.

Method for disambiguation features fusion

In this section, we present how to fuse various disambiguation features using the DST in combination with Shannon's entropy. The DST of evidence provides a way to associate measures of uncertainty to sets of hypothesis when the individual hypothesis is imprecise or unknown (Lucas and Van Der Gaag 1991). All possible mutually exclusive hypothesis are contained in a *frame of discernment*. For our application, the hypothesis will be if either a couple of clusters of papers in a collection Ω belongs to a same author or not. For instance, given three clusters of papers $s_1 = \{p_1\}$, $s_2 = \{p_2, p_3\}, s_3 = \{p_4, p_5, p_6\}$ and $\Omega = \{s_1, s_2, s_3\}$, further let $\theta = \{c_1 = \langle s_1, s_2\rangle, c_2 = \langle s_2, s_3\rangle, c_3 = \langle s_1, s_3\rangle\}$ (where $c_k = \langle s_i, s_j\rangle$ indicates c_k is a couple of two paper clusters s_i and s_j), the frame of discernment is then a *power-set* of θ, that is, $2^\theta = \{\{c_1, c_2, c_3\}, \{c_1, c_2\}, \{c_1, c_3\}, \{c_2, c_3\}, \{c_1\}, \{c_2\}, \{c_3\}, \emptyset\}$.

To set up the framework of the DST, the *basic probability assignment* or *mass* function (presented by m) are required. The mass function defines a mapping of 2^θ to the interval [0,1], where the mass of the null set is 0, namely, $m(\emptyset) = 0$, and the summation of masses of all the subsets of 2^θ is 1, namely, $\sum_{A \in 2^\theta} m(A) = 1$. The value of the mass function $m(A)$ for a given set A, expresses the proportion of all relevant and available evidence that supports the claim that a particular element of power set of θ belongs to the set A and not any subset of A. In the case of our approach, every evidence will provide a mass function to each element contained in the frame of discernment, by using features related to this evidence. When the evidence observes a couple of clusters and detects there exist some sub-features associated with it, the probability that the observed couple $A = \{c\}$ may

Table 1 Features for a pair of papers $\langle p_i, p_j \rangle$

Feature	Description
Affiliation	Correlation of p_i and p_j based on affiliation
Venue	Correlation of p_i and p_j based on venue
Contentsim	Similarity between contents of p_i and p_j
Coauthor	Correlation of p_i and p_j based on co-authorship
Citation	Correlation of p_i and p_j based on citations
Webcorrelation	Correlation of p_i and p_j based on Web pages

belong to a same author is given by the confidence interval *[Belief(A), Plausibility(A)]*. The lower bound, *Belief(A)*, is defined as the sum of all the masses of the proper subset of A. The upper bound, *Plausibility(A)*, is the sum of all masses of the sets intersecting A.

$$Belief(A) = \sum_{B|B \subseteq A} m(B), Plausibility(A) = \sum_{B|B \cap A \neq \emptyset} m(B) \tag{1}$$

The purpose of aggregation of features is to meaningfully summarize and simplify a collection of evidence whether the evidence is from a single source or multiple sources. These multiple evidence provides separate assessments for the same frame of discernment, and the DST is based on the assumption that these evidence is independent. To combine the evidence ψ_1 and ψ_2 detected by two features F_1 and F_2, DST provides a combination rule which is indicated by Eq. 2.

$$[m_{\psi_1} \oplus m_{\psi_2}](A) = (1 - K)^{-1} \sum_{B \cap C = A \neq \emptyset} m_{\psi_1}(B) \cdot m_{\psi_2}(C) \tag{2}$$

where, $[m_{\psi_1} \oplus m_{\psi_2}](\emptyset) = 0, K$ measures the amount of conflict between the two evidence and is given by Eq. 3.

$$K = \sum_{B \cap C = \emptyset} m_{\psi_1}(B) \cdot m_{\psi_2}(C) \tag{3}$$

In our approach, the mass functions are used to represent the correlation between two clusters of papers. However, the DST requires that we know how certain evidence is when determining whether two clusters of papers belong to the same author or not. In traditional applications, these values are given by domain experts and represent the importance of evidence, however, this does not pose a flexible strategy. To address this issue, we use the Shannon's entropy formula by following the works (Yu et al. 2005; Moreira and Wichert 2013). Let ψ(representing evidence) be a discrete random variable corresponding to a first-level feature $F = \{f_1, \ldots, f_{|F|}\}$ (for example, *Affiliation = {CoAff,CoAffOccur}* means that the first-level feature *Affiliation* contains two sub-features *CoAff* and *CoAffOccur*) and $c \in \theta$. Let $rel(f, c)$ be a function which determines if the score $score(f, c)$ provided by a sub-feature $f \in F$ for c is relevant ($rel(f, c)$ takes 1 if $score(f, c) > 0$, otherwise, it returns 0), and $|\theta|, |F|$ be respectively cardinalities of θ and F. Entropy of the evidence ψ (corresponding to F) is defined as Eq. 4,

$$E(\psi) = -\sum_{f \in F} \sum_{c \in \theta} \frac{rel(f, c)}{|F| \cdot |\theta|} log \frac{rel(f, c)}{|F| \cdot |\theta|} \tag{4}$$

where, if the entropy is 0, there are no levels of uncertainty associated with the evidence ψ, since feature F provides consistent information for all couples of clusters. However, the entropy value is not normalized, we need to get to the maximum entropy associated with ψ to normalize it. The maximum entropy can be estimated directly by $MaxE(\psi) = log|F| \cdot |\theta|$, as the sub-features of F are equally distributed over all couples of clusters.

In summary, for our application of DST, the mass function for evidence ψ will be given by Eq. 5, where the values of $score(f, c)$ for every sub-features are scaled using the min-max normalization technique, then summed directly to support a couple of clusters. Noted that, since we focus on applying the DST to fuse various disambiguation evidence and coming up with a more reliable candidate pair of clusters for amalgamation during the clustering process. We only consider pairs of clusters($\{c\}$) and the θ when defining the mass functions. And under our context, it is meaningless and hard to be explained given a mass function for other elements in the framework of discernment. Although the full power of DST may be not used, this simplified method provides another handy way to combine multiple sources of evidence in author disambiguation tasks, compared with additional feature combination schemes previously proposed.

$$m_{\psi}(A) = \begin{cases} \sum_{f \in F} score(f, c) & \text{if} \quad A = \{c\} \\ E(\psi)/MaxE(\psi) & \text{if} \quad A = \theta \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

Finally, when finishing the calculation of mass function for every evidence in an evidence space Ψ, we need to rescale all mass values to fulfill the conditions of the mass function. The mass of θ in distinct evidence is normalized depending on Eq. 6, after that, the masses of the remaining elements of 2^{θ} within the same evidence are rescaled using Eq. 7.

$$m_{\psi}(\theta) = \frac{m_{\psi}(\theta)}{\sum_{\psi_i \in \Psi} m_{\psi_i}(\theta)} \tag{6}$$

$$m_{\psi}(A \neq \theta) = (1 - m_{\psi}(\theta)) \frac{m_{\psi}(A)}{\sum_{A_j} m_{\psi}(A_j)} \tag{7}$$

Author disambiguation algorithms

After fusing various disambiguation features using above-mentioned DST framework, we can get a matrix to represent pairwise correlations of papers. It seems resemble the traditional similarity matrix, however, different from the similarity matrix, each entry in this matrix is linked to a belief function and a plausibility function. Although we can perform clustering on this matrix using a density-based method(e.g. DBSCAN), our tests confirmed that this is not the right choice. Therefore, we present another DST-based hierarchical agglomerative clustering (DSHAC) algorithm for author disambiguation. Details of the DSHAC are shown as Algorithm 1.

Algorithm 1 *DSHAC:DST based hierarchical agglomerative clustering*

```
Input: An ambiguous author group of papers $\Omega = \{\{p_1\}, ..., \{p_{N_p}\}\}$, an user-specified number of authors $N(optional)$, a feature space $\Phi = \{F_1, ..., F_{|\Phi|}\}$, a set of evidence $\Psi$, a convergence $flag$;
Output: Disambiguated $\Omega$;
  Partitioning $\Omega$ by $CoAuthor$ feature, and remove it from $\Phi$;% first step
  repeat
    if $|\Omega| = N$ and $N > 0$ then
      flag← true; break;
    end if
    for each feature $F_k \in \Phi$ do
      create an evidence $\psi_k$ corresponds to $F_k$;
      for each unique pair $c = \langle s_i, s_j \rangle, s_i, s_j \in \Omega$ do
        estimate correlation of $c$ based on feature $F_k$(by Eq. 8)and add $c$ to $\psi_k$;
      end for
      estimate $E(\psi)$ for $\psi_k$ by Eq.4;
      if $E(\psi) = 0$ then
        remove $F_k$ from $\Phi$ and abandon $\psi_k$;
      else
        add $\psi_k$ to $\Psi$;
      end if
    end for
    for all $\psi_k$ in $\Psi$ do
      estimate every $m_\psi(c)$ and $m_\psi(\theta)$ by Eq.5, Eq.6 and Eq.7;
    end for
    for each $\psi_k \in \Psi$ do
      $\psi_{comb} = \psi_{comb} \oplus \psi_k$;% combine all available evidence
    end for
    select the best candidate $c$ according to $\psi_{comb}$;
    merge $c.s_j$ into $c.s_i$ and remove $c.s_j$ from $\Omega$;
    if flag ← $cvgtest()$ then
      break;
    end if
  until flag=false
  return $\Omega$ obtained previously;
```

In the first step, we partition the original set of papers into some seed clusters utilizing some discriminative features. The basic feature considered is the coauthors. We set up a co-authorship graph for an ambiguous group of papers, then exploit a recurring pattern to generate seed clusters over the co-authorship graph, that is, two papers are placed together in a same cluster if both papers have at least two coauthors in common. We have based this strategy on a general observation that only very rarely two ambiguous authors share a coauthor (Kang et al. 2009; Cota et al. 2010; Ferreira et al. 2010; Velden et al. 2011; Milojevic 2013). However, this strategy can still bring forth transitivity errors and mis-clustering of papers, thus we may exploit more discriminative features, e.g. affiliations of papers, for partitioning the original set of papers. For this case, we could assign two papers into a seed cluster when both the similarity values are greater than a threshold. It is a pity that using multiple features could generate more fragmented seed clusters. [Noticed that our proposed strategy can still work well even confronted with more complex cases, for example, all of the author names are as an abbreviation format. We have proposed a method to deal with such complex situations in Wu et al. (2012)].

The disambiguation of the first step can bring the following benefits: (a) we can obtain pure seed clusters of papers, it can decrease the uncertainty brought by noise evidence in subsequent clustering process. Our tests confirmed that the purity of seed clusters generated in the first step can reach nearly 100 % in pairwise precision. It means that each seed cluster definitely belongs to one person. (b) it reduces the computational overhead of subsequent iteration process. According to our tests, the iterations in subsequent clustering process can be reduced by more than 60 %.

Algorithm 2 *CvgTest:Convergence Test for DSHAC*

```
Input: $\Omega$, $\Phi$, $\xi$, $\zeta$, candidate pair of clusters $c =< s_1, s_2 >$, and the number of available evidence $M$(fixed
  usually as 2);
Output: A convergence flag;
  if $|\Phi| = 0$ then
    return true;
  end if
  flag1 ← false; flag2 ← false;
  if $|\Phi| \leq M$ then
    calculate objective function $O_k = \frac{2}{|\Omega||\Omega-1|} \sum_{s_i, s_j \in \Omega} D(s_i, s_j)$ by Eq.9;
    if $O_k > O_{k-1}$ and $|O_k - O_{k-1}| > \xi$ then
      flag1 ← true;
    end if
  end if
  if $|\Phi| = M - 1$ then
    if $corr_{AL}(s_1, s_2) < \zeta$ then
      flag2 ← true;
    end if
  end if
  return flag← flag1 or flag2;
```

In the second step, disambiguation is performed over the seed clusters using iteration strategy of hierarchical agglomerative clustering (HAC). In each iteration, we first calculate the pairwise correlations of clusters for each high-level feature using a linkage criteria. We respectively examined three well-known linkage criteria (namely, single-linkage, complete-linkage, and average-linkage), and found the best performance is achieved by the average-linkage (shown as Eq. 8).

$$corr_{AL}(s_1, s_2) = \frac{\sum_{p_i \in s_1} \sum_{p_j \in s_2} corr(p_i, p_j)}{|s_1| \cdot |s_2|} \tag{8}$$

Then, we create an evidence for each high-level feature, and estimate the mass for each couple of clusters contained in the discernment frame of the evidence depending upon Eqs. (5–7). Finally, we combine all available evidence together (by Eq. 3) and select a candidate couple of clusters with the largest belief (if several candidates with the same belief are found, the one with the highest plausibility is returned) in the combined evidence to merge. Such iteration is continued until the convergence test is satisfied.

During the process of clustering, one fundamental problem is how to determine the convergence condition. To deal with this problem, we exploit three kinds of convergence settings: (a) A presetting number of clusters: The clustering process will terminate as the number of clusters changes to equal out a user specified number. (b) The number of available evidence: During the clustering process, strong evidence would be first used. When the clustering process converges, there is usually only one or two weak evidence. According to our experiments, nearly 95 % of cases are the ContentSim if only one evidence is left. In this case, we need further merge those clusters with high content similarities. Thus it is necessary to preset a correlation threshold for each weak evidence. (c) Distance between clusters: a most common approach to determine whether a clustering process converges or not is to investigate inter-clusters or inner-clusters distance. For example, the family of k-means algorithms terminates the clustering process by minimizing the inner-clusters distance (Han et al. 2005). In contrast with the k-means algorithms, we carry on the inter-cluster distance to detect the best convergence point. Given the clusters of papers Ω and a feature $F \in \Phi$, we first calculate the average feature vector V^F to represent every clusters, then we use cosine metric to determine the pairwise distances of these feature vectors, the computation is shown as Eq. 9.

$$V_{\Phi}(s) = corr^{\Phi}_{AL}(s, s), D(s_1, s_2) = cosine(V_{\Phi}(s_1), V_{\Phi}(s_2)) \tag{9}$$

To use the third setting, we need to set the second convergence condition, for example considering the situation when the number of available evidence changes to two. Under this condition, we test the overall distance of clusters O_k for k-th iteration, if a significant growth ξ is observed on the overall distance, the clustering process will be terminated. A practical combination of convergence settings is summarized as Algorithm 2.

Experiments and evaluation

Datasets

We perform our experiments on a real-world dataset came from scholarly search system-Arnetminer. The dataset consists of 100 ambiguous author groups, and each group contains several distinct authors sharing a unique name spelling and their published papers. It sums up 6,730 papers associated with 1,382 distinct authors, which mean an average of approximately five papers per author. The dataset was originally created by Tang et al. (2012); Wang et al. (2011) and derived from DBLP[1] which is the longest running and most important computer science bibliography and database on the Web today. Since some sub-collections of this dataset had been used in several previous works (Cota et al. 2010; Ferreira et al. 2010, 2012; Han et al. 2004, 2005; Huang and Seyda Ertekin 2006; Tan et al. 2006, 2012; Wu et al. 2012; Yang et al. 2008), Tang et al. expanded the collection to include more ambiguous author groups. Each paper is manually labelled with a number indicating the cluster that it has been assigned to. The annotation was performed based on the publication list on the authors' homepages, affiliation and email addresses in the PDF files. There are a few extreme cases (less than 1 %) that even human cannot judge which person (cluster) a paper belongs to. For such cases, the paper is posted to an "other" cluster. Beyond the basic disambiguation features such as coauthors, titles and venues of papers, more rich disambiguation features of papers such as the *Citation*, the *Affiliation* and the *WebCorrelation* are added by Tang et al. Specifically, the *CoAffOccur* information is extracted based on the first page of PDF files of papers. For the *WebCorrelation*, Tang and co-workers (2011) used Google to find relevant Web pages for a given author name, and then applied a pre-trained classification model to identify whether a returned web page is a homepage or not. Based on the homepage accepted, the *WebCorrelation* of two papers is assgined as a boolean value to indicate whether this two papers co-occur in one homepage or not. Note that, in this dataset, there are only 47 % papers having the citations, 45 % papers having the authors affiliations, and 67 % author names having corresponding homepages. Also, feature values may contain noise. The statistics of the dataset are listed in Table 2 and more details are available online[2].

In our experiments, all correlation values of features are taken as a boolean value(namely, 1 or 0) except the *ContentSim*. For the *CoAuthor*, two authors are considered to refer to the same person if the two full names are identical. For the *Affiliation*, we consider two affiliations referring to the same one, if the correlation value of them are larger than a threshold. For *ContentSim*, we only employ the pairwise similarities of paper titles, and the correlation values are calculated using the cosine measure in combination with TF-IDF

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

[2] http://www.arnetminer.org/disambiguation

(Term Frequency-Inverse Document Frequency) based word-level weight. In addition, we fuse *CoAff* and *CoAffOccur* provided by the dataset into a single evidence considering that they can be treated as sub-features of the *Affiliation*.

Evaluation measures and baseline methods

To evaluate the effectiveness of proposed disambiguation method, we used two evaluation metrics: K metric and pairwise F1.

The K metric (Lapidot 2002; Cota et al. 2010) consists of the geometric mean between the average cluster purity (ACP) and the average author purity (AAP), namely $K = \sqrt{ACP \cdot AAP}$. It analyses the purity and fragmentation of the empirical clusters extracted by each method. Given an ambiguous group of papers Ω, ACP (Eq. 10) evaluates the purity, and AAP (Eq. 11) evaluates the fragmentation of the ground-truth clusters to the empirical clusters for this group, respectively:

$$ACP = \frac{1}{N_p} \sum_{j=1}^{N} \sum_{i=1}^{M} \frac{n_{ij}^2}{n_j} \tag{10}$$

$$AAP = \frac{1}{N_p} \sum_{i=1}^{M} \sum_{j=1}^{N} \frac{n_{ij}^2}{n_i} \tag{11}$$

where N_p is the total number of papers in Ω, N is the number of empirical clusters for Ω, M is the number of ground-truth clusters for Ω, n_i and n_j are respectively the number of papers in the ground-truth cluster s_i and the empirical cluster s_j, n_{ij} is the number of papers occurred in both clusters s_i and s_j.

Pairwise F1 (*pF1*) (Rijsbergen 1979) is the F1 metric calculated using pairwise precision (*pP*) and pairwise recall (*pR*). To define pairwise measures, we define a function $Pairs(s)$ that takes in a cluster s and returns the set of distinct pairs of papers in it. Then we estimate pairwise measures by estimating the number of pairs of papers as followings Eqs. 12 and 13,

$$pP = \frac{|\{Pairs(s_{j=1:N})\} \cap \{Pairs(s_{i=1:M})\}|}{|\{Pairs(s_{j=1:N})\}|} \tag{12}$$

$$pR = \frac{|\{Pairs(s_{j=1:N})\} \cap \{Pairs(s_{i=1:M})\}|}{|\{Pairs(s_{i=1:M})\}|} \tag{13}$$

The pF1 metric is defined as the harmonic mean of pairwise precision and pairwise recall: $pF1 = 2pP \cdot pR / pP + pR$

In addition, to learn about the ability of an approach on detecting true number of distinct authors, we define a simple measure, shown as followings (Eq. 14),

$$R = \frac{2 \times min(M, N)}{M + N} \tag{14}$$

where M and N are respectively the number of empirical clusters and ground-truth clusters for an ambiguous paper group. Also, we define the Root Mean Square Deviation of R as

Table 2 The statistics of the dataset

Parameters	Number
Number of ambiguous groups	100
Number of distinct authors	1,382
Number of papers	6,730
Number of papers per author	4.87

$RMSD = \sqrt{\frac{1}{N_t}\sum_{i=1}^{N_t}(R_i - 1)^2}$ (N_t is the total number of test cases) to measure the differences between predicted author number and observed author number. Higher R means a higher performance on estimation of the number of distinct authors. Note that, during results analysis, all the evaluation metrics are averaged over all $N_t = 100$ ambiguous groups.

As far as the baseline model is concerned, we select three unsupervised methods: spectral clustering (SC) (Han et al. 2005), DBSCAN (Huang and Seyda Ertekin 2006), HHC (Cota et al. 2010) and a supervised method-pairwise factor graph model (PFGM) (Tang et al. 2012; Wang et al. 2011).

HHC (Cota et al. 2010) is a heuristic-based hierarchical clustering method to deal with the name disambiguation problem. The method successively fuses clusters of papers of compatible authors based on several heuristic and similarity measures on the components of the papers. Experiments with a dataset taken from the DBLP collection show that HHC outperforms a SVM-based supervised method and a K-Means-based unsupervised method. HHC has a strategy similar to our DSHAC, where both of them partition the original set of papers into some seed clusters using the co-authorship information. In the second phase, two seed clusters are fused if the similarity estimated on either of the field of title and venue is more critical than a given threshold. This process continues until it cannot fuse clusters anymore. For our experiments, we use HHC based on the seed clusters produced by the first phase of the DSHAC, and set correlation measure thresholds corresponding to each feature. Noticed that, different from the DSHAC which fuses various features simultaneously, during the hierarchical clustering stage, we use all the features in a flat manner as proposed in the work (Cota et al. 2010).

For both the SC and the DBSCAN, we first use SVMs to learn the pairwise similarity of papers given features above-mentioned, and then apply respectively the clustering strategies of the SC and the DBSCAN to generate consequences. For experiments, we create the training and test sets built on the ground-truth of the dataset. The records in both the training and test sets consist of comparisons between two publications. Given a record $<p_i, p_j, label, V_F>$ where the feature vector V_F consists of all pairwise correlation values estimated on all presented features, if p_i and p_j belong to a same cluster, we assign $label = 1$ to represent positive correlation, otherwise we assign $label = 0$. We enumerate all records according to the ground-truth of the dataset, and hold out a small proportion of the records (about 10 %) to train the SVMs. Then for each ambiguous author group, the remaining records are invoked as the test set. We apply the learned "model" to estimate the confidence values which determine the distances between the record pairs (i.e. pairs of papers). The distance matrix obtained by this method is taken as the input both for the SC and the DBSCAN algorithms.

The PFGM is built on the HMRF (Tang et al. 2012) and outperforms several other disambiguation models (we ignore the comparisons with them for simplicity) on the same dataset presented here (Wang et al. 2011). To evaluate the PFGM method, we utilize the disambiguation results of the PFGM (provided in the dataset) to calculate the K metrics and the pF1 metrics.

Results and analysis

We first analyze the disambiguation performance of the DSHAC based on different options and convergence settings. The results are presented in Table 3. For *FirstStep*, the clusters are generated using the co-authorship among papers. $Evid \leq (1, 2)$ indicates the case that the clustering process converges as the number of available evidence decreases to equal out two or one (some times this number directly changes from three to one, thus we use the symbol $\leq$). *Fixed-N* and *Auto-N* represent the cases that the clustering process terminates as the number of clusters(*N*) reaches a user-specified value and an auto-detected value, respectively. According to Table 3, exploiting co-authorship among papers can generate pure seed clusters, thus achieve high precision (nearly 100 %). Also, it can intensively reduce the number of clusters (by more than 60 % compared to the total number of papers) being merged in the following steps, thus save the calculation cost for the whole process. In examining the different settings of convergence condition, we found that the strategy built on the number of available evidence is simple, yet obtains good results. When the number is configured as two, the clustering results achieve higher precision, while, as it is set as one, the disambiguation results obtain a higher recall. This outcome allows the disambiguation results to achieve a better tradeoff between the precision and the recall using further optimization. When the true number of distinct authors is given (i.e. *Fixed-N*), the best results are obtained in both *K* and *pF1* metrics. Concerning the disambiguation performance obtained by the CvgTest algorithm (i.e. *Auto-N*), it significantly improves disambiguation effects in all indicators compared with the strategy of considering the number of available evidence. In particular, *R* metric reaches 0.897. *Auto-N* also achieves substantially the same results in both K metric and pF1 measure in comparison with the case of *fixed-N*.

We next show results concerning the comparison of the disambiguation performance obtained by DSHAC and the performance obtained by four baseline methods. The results are presented in Figs.1, 2, and 3. According to Figs.1 and 2, the SC method performs worst, even given the exact number of distinct authors. The main reason for this situation is that the SC method generates clusters based on the principle of k-means clustering (Han et al. 2005), and isolated noise data points(papers) are forced to be assigned to a bigger cluster. In the publication dataset, however, a common observation is that many authors have only a small number of papers, and even only one paper. Such a sharp noise situation often makes k-means-like algorithms a higher mis-clustering rate in author disambiguation tasks. In contrast to the SC method, the DBSCAN method can handle this noise situation (Ester et al. 1996), and therefore significantly outperforms it (in our implementation of the DBSCAN method, isolated papers are treated as separated clusters). Compared to both the SC method and the DBSCAN method, our method has more advantages. On one hand, our method outperforms both of them on all indicators; on the other hand, our method does not need any supervised strategies to learn feature weight (Both the SC method and the DBSCAN method need a feature-weight learning stage for fusion of various features). With respect to the HHC, it has similar performance to the DBSCAN, specially, it outperforms the DBSCAN in the ACP metric and the pP metric since the heuristic strategy used in the HHC ensures more pure clusters. However, the HHC has an obligation to set a threshold value for each feature to ensure the quality of clustering, which is a hard job if

Table 3 Experimental results of DSHAC based different options and settings

DSHAC	ACP	AAP	K	pP	pR	pF1	R
FirstStep	**0.939**	0.628	0.753	**0.9998**	0.567	0.683	0.650
Evid ≤ 2	0.905	0.743	0.812	0.929	0.734	0.794	0.763
Evid ≤ 1	0.831	0.831	0.822	0.810	0.867	0.799	0.836
Fixed-N	0.844	**0.860**	**0.851**	0.865	**0.884**	**0.869**	–
Auto-N	0.874	0.829	0.849	0.899	0.849	0.863	**0.897**

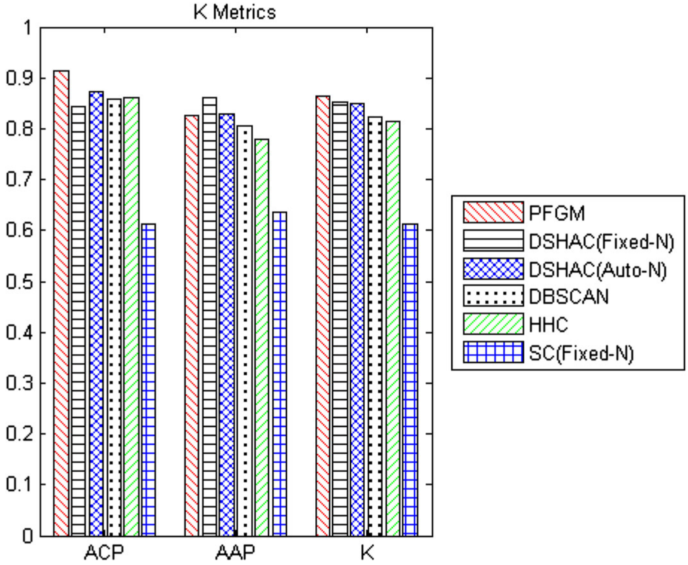

Fig. 1 Author disambiguation performance by K metrics

more disambiguation features are added. The HHC may be further improved on disambiguation performance by adjusting the thresholds. Nevertheless, the importance of every feature is not considered. In contrast to the HHC, the DSHAC becomes handy when fusing seed clusters since it can automatically identify the importance of each feature using Shannon’s entropy formula, consequently some gains on disambiguation performance can be made.

Although the PFGM method performs best in K metric and $pF1$ measure, our method achieves comparable results to it. In the K metric and the $pF1$ metric, the DSHAC model has a loss of 1.5 and 2.2 %, respectively. On the R indicators, it improves the PFGM model by 5 %. One point worths emphasizing now is that three baseline methods employ the ground-truth to optimize the combination of features to reduce the conflicting among distinctive features, our method handles this conflicting automatically. These results indicate that the DSHAC is attractive in practice. First, it does not require any training data, thus simplifies the design and development of softwares. Second, the framework of the DSHAC is flexible, since it is easy to coordinate with numerous strong evidence to improve author disambiguation. For example, new evidence can be added to capture users’

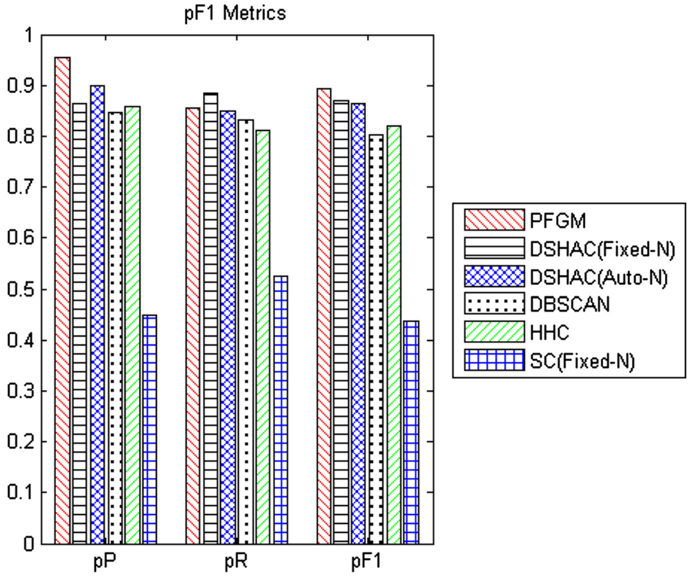

Fig. 2 Author disambiguation performance by pF1 metrics

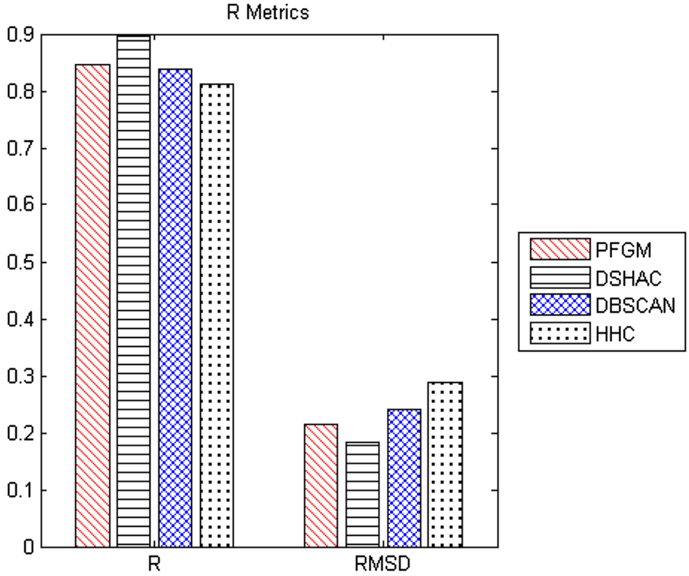

Fig. 3 Author disambiguation performance by R metrics

feedbacks during the clustering process, then this additional evidence can be entered into our framework to achieve more complete results. By the way, an active model for author disambiguation can be fulfilled (such a situation is, however, beyond the scope of this article, we will study it in the future).

Conclusion

We have proposed an unsupervised approach (DSHAC) for author disambiguation task on the basis of the Dempster–Shafer theory of evidence. According to experiments, our method is superior to three unsupervised baseline methods, and also achieves comparable results to a supervised method even without any training mechanism. The main contribution of this paper lies in the DSHAC algorithm for author disambiguation. The DST framework enables flexible and effective fusion of numerous heterogeneous features in author disambiguation tasks. The use of Shannon's entropy formula provides a heuristic manner to uncover the importance of each feature. Thus allows us to build a completely unsupervised disambiguation method.

However, there still exist some the limitations with respect to our method. Firstly, author disambiguation results have not been much better than other unsupervised methods in terms of accuracy. Secondly, the experimental dataset is relatively small and seems specialised so the method may not work well on other datasets thus need to be tailored to different situations. Also, the best parameters have been taken in experiments may lead to overfitting and optimistic accuracy estimates. In addition, we currently only consider the pairs of clusters and the θ when defining the mass functions (refer to Eq. 5) to provide one simple solution for author disambiguation tasks based on the DST, thus the full power of the DST may be not used. To make the DST fully play to its strengths, more sophisticated models need to be developed.

As future work, we would like to continually improve our proposed model and experiment the model with large-scale datasets. Also, we plan to develop an active and online model for author disambiguation based on the DSHAC. In addition, we would want to examine more features, such as full-text of papers, emails of authors or other implicit evidence to improve the disambiguation performance.

Acknowledgments This work is supported by the Scientific Research Project of Yunnan University (2010YB024) and the Applied Basic Research Project of Yunnan Province (2013FB009). This work of Jun He is supported by the National Natural Science Foundation of China (61203273). We are grateful to anonymous reviewers for their useful comments and suggestions which contribute to substantially improving this paper.

References

Cota, R. G., Ferreira, A. A., Nascimento, C., Goncalves, M. A., & Laender, A. H. (2010). An unsupervised heuristic-based hierarchical method for name disambiguation in bibliographic citations. *Journal of the American Society for Information Science and Technology*, *61*(9), 1853–1870.

Culotta, A., Kanani, P., Hall, R., Wick, M., & McCallum, A. (2007). Author disambiguation using error-driven machine learning with a ranking loss function. In *Proceedings of the 6th international workshop on information integration on the web*.

Ester, M., Kriegel, H. P., Sander, J., & Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the 2nd international conference on knowledge discovery and data mining* (pp. 226–231).

Fan, X., Wang, J., Pu, X., Zhou, L., & Lv, B. (2011). On graph-based name disambiguation. *Journal of Data and Information Quality*, *2*(2), 10.

Ferreira, A. A., Goncalves, M. A., & Laender, A. H. (2012). A brief survey of automatic methods for author name disambiguation. *ACM SIGMOD Record*, *41*(2), 15–26.

Ferreira, A. A., Machado, T. M., & Goncalves, M. A. (2012). Improving author name disambiguation with user relevance feedback. *Journal of Information and Data Management*, *3*(3), 332–347.

Ferreira, A. A., Veloso, A., Goncalves, M. A., & Laender, A. H. (2010). Effective self-training author name disambiguation in scholarly digital libraries. In *Proceedings of the 10th ACM/IEEE-CS joint conference on digital libraries* (pp. 39–48).

Gurney, T., Horlings, E., & Van Den Besselaar, P. (2012). Author disambiguation using multi-aspect similarity indicators. *Scientometrics*, *91*(2), 435–449.

Han, H., Giles, C. L., & Hong, Y. Z. (2004). Two supervised learning approaches for name disambiguation in author citations. In *Proceedings of the 4th ACM/IEEE-CS joint conference on digital librarie* (pp. 296–305).

Han, H., Zhang, H., & Giles, C. L. (2005). Name disambiguation in author citations using a k-way spectral clustering method. In *Proceedings of the 5th ACM/IEEE-CS joint conference on digital libraries* (pp. 334–343).

Huang, J., & Seyda Ertekin, C. L. G. (2006). Efficient name disambiguation for large scale databases. In *Proceedings of the 10th European conference on principles and practice of knowledge discovery in databases* (pp. 536–544).

Kalashnikov, D. V., & Mehrotra, S. (2006). Domain-independent data cleaning via analysis of entity relationship graph. *ACM Transactions on Database System*, *31*(2), 716–767.

Kang, I. S., Na, S. H., Lee, S., Jung, H., Kim, P., Sung, W. K., et al. (2009). On co-authorship for author disambiguation. *Information Processing & Management*, *45*(1), 84–97.

Lalmas, M., & Ruthven, I. (1998). Representing and retrieving structured documents using the Dempster–Shafer theory of evidence: Modelling and evaluation. *Journal of Documentation*, *54*(5), 529–565.

Lapidot, I. (2002). *Self-organizing-maps with BIC for speaker clustering*. Martigny, IDIAP Research Institute, Switzerland: Technical report.

Levin, M., Krawczyk, S., Bethard, S., & Jurafsky, D. (2012). Citation-based bootstrapping for large-scale author disambiguation. *Journal of the Association for Information Science and Technology*, *63*(5), 1030–1047.

Lucas, P., & Van Der Gaag, L. (1991). *Principles of expert systems*. Chicago: Addison-Wesley Longman Publishing Co., Inc.

McRae-Spencer, D. M., & Shadbolt, N. R. (2006). Also by the same author: AKTiveAuthor, a citation graph approach to name disambiguation. In *Proceedings of the 6th ACM/IEEE-CS joint conference on digital libraries* (pp. 53–54).

Milojevic, S. (2013). Accuracy of simple, initials-based methods for author name disambiguation. *Journal of Informetrics*, *7*(4), 767–773.

Moreira, C., & Wichert, A. (2013). Finding academic experts on a multisensor approach using Shannon's entropy. *Expert Systems Applications*, *40*(14), 5740–5754.

Pereira, D. A., Ribeiro, B. N., Ziviani, N., Alberto, H. F., Goncalves, A. M., & Ferreira, A. A. (2009). Using web information for author name disambiguation. In *Proceedings of the 9th ACM/IEEE joint conference on digital libraries* (pp. 49–58).

Rijsbergen, C. J. V. (1979). *Information retrieval* (2nd ed.). London: Butterworths.

Ruthven, I., & Lalmas, M. (2002). Using Dempster–Shafer's theory of evidence to combine aspects of information use. *Journal of Intelligent Information Systems*, *19*(3), 267–301.

Shafer, G. (1976). *A mathematical theory of evidence* (Vol. 1). Princeton: Princeton University Press.

Smalheiser, N. R., & Torvik, V. I. (2009). Author name disambiguation. *Annual Review of Information Science and Technology*, *43*(1), 1–43.

Song, Y., Huang, J., Councill, I. G., Li, J., & Giles, C. L. (2007). Efficient topic-based unsupervised name disambiguation. In *Proceedings of the 7th ACM/IEEE joint conference on digital libraries* (pp. 342–352).

Strotmann, A., & Zhao, D. (2012). Author name disambiguation: What difference does it make in author-based citation analysis? *Journal of the Association for Information Science and Technology* , *63*(9), 1820–1833.

Strotmann, A., Zhao, D., & Bubela, T. (2009). Author name disambiguation for collaboration network analysis and visualization. *Journal of American Society for Information Science technology*, *46*, 1–20.

Tan, Y. F., Kan, M. Y., & Lee, D. W. (2006). Search engine driven author disambiguation. In *Proceedings of the 6th ACM/IEEE joint conference on digital libraries* (pp. 314–315).

Tang, J., Fong, A. C. M., Wang, B., & Zhang, J. (2012). A unified probabilistic framework for name disambiguation in digital library. *IEEE Transactions on Knowledge and Data Engineering* , *24*(6), 975–987.

Torvik, V. I., Weeber, M., Swanson, D. R., & Smalheiser, N. R. (2005). A probabilistic similarity metric for Medline records: A model for author name disambiguation. *Journal of the American Society for Information Science and Technology*, *56* (2), 140–158.

Treeratpituk, P., & Giles, C. L. (2009). Disambiguating authors in academic publications using random forests. In *Proceedings of the 9th ACM/IEEE-CS joint conference on digital libraries* (pp. 39–48).

Velden, T. A., Haque, A. U., & Lagoze, C. (2011). Resolving author name homonymy to improve resolution of structures in co-author networks. In *Proceedings of the 11th ACM/IEEE-CS joint conference on digital libraries* (pp. 241–250).

Wang, X., Tang, J., Cheng, H., & Yu, P. S. (2011). ADANA: Active name disambiguation. In *Proceedings of the IEEE 11th international conference on data mining* (pp. 794–803).

Wu, J., & Ding, X. (2013). Author name disambiguation in scientific collaboration and mobility cases. *Scientometrics*, *96*(3), 683–697.

Wu, H., Pei, Y. J., & Li, B. (2012). Scholar search-oriented author disambiguation. In *Proceedings of the 9th international conference on fuzzy systems and knowledge discovery* (pp. 1166–1170).

Wu, H., Pei, Y. J., & Yu, J. (2009). Detecting academic experts by topic-sensitive link analysis. *Frontiers of Computer Science in China*, *3*(4), 445–456.

Yang, K. H., Peng, H. T., Jiang, J. Y., Lee, H. M., & Ho, J. H. (2008). Author name disambiguation for citations using topic and web correlation. In *Proceedings of the 12th European conference on research and advanced technology for digital libraries* (pp. 185–196).

Yin, X., Han, J., & Yu, P. S. (2007). Object distinction: Distinguishing objects with identical names. In *Proceedings of IEEE the 23rd international conference on data engineering* (pp. 1242–1246).

Yu, Z., Tian, Y., & Xi, B. (2005). Dempster–Shafer evidence theory of information fusion based on info-evolutionary value for e-business with continuous improvement. In *Proceedings of IEEE international conference on e-Business engineering* (pp. 586–590).