Adaptively Entropy-Based Weighting Classifiers in Combination Using Dempster-Shafer Theory for Word Sense Disambiguation

Van-Nam Huynh [a,*], Tri Thanh Nguyen [b], Cuong Anh Le [b]

[a] *Japan Advanced Institute of Science and Technology*
1-1 Asahidai, Nomi, Ishikawa 923-1292, Japan

[b] *College of Technology*
Vietnam National University, Hanoi
144 Xuan Thuy, Cau Giay District, Hanoi, Vietnam

Abstract

In this paper we introduce an evidential reasoning based framework for weighted combination of classifiers for word sense disambiguation (WSD). Within this framework, we propose a new way of defining adaptively weights of individual classifiers based on ambiguity measures associated with their decisions with respect to each particular pattern under classification, where the ambiguity measure is defined by Shannon's entropy. We then apply the discounting-and-combination scheme in Dempster-Shafer theory of evidence to derive a consensus decision for the classification task at hand. Experimentally, we conduct two scenarios of combining classifiers with the discussed method of weighting. In the first scenario, each individual classifier corresponds to a well-known learning algorithm and all of them use the same representation of context regarding the target word to be disambiguated, while in the second scenario the same learning algorithm applied to individual classifiers but each of them uses a distinct representation of the target word. These experimental scenarios are tested on English lexical samples of Senseval-2 and Senseval-3 resulting in an improvement in overall accuracy.

Key words: Computational linguistics, Classifier combination, Word sense disambiguation, Dempster's rule of combination, Entropy.

* Corresponding author. Email address: huynh@jaist.ac.jp (V.-N. Huynh)

1 Introduction

Polysemous words that have multiple senses or meanings appear pervasively in many natural languages. While it seems not much difficult for human beings to recognize the correct meaning of a polysemous word among its possible senses in a particular language given the context or discourse where the word occurs, the issue of automatic disambiguation of word senses is still one of the most challenging tasks in natural language processing (NLP) [29], though it has received much interest and concern from the research community since the 1950s (see [15] for an overview of WSD from then to the late 1990s). Roughly speaking, WSD is the task of associating a given word in a text or discourse with an appropriate sense among numerous possible senses of that word. This is only an "intermediate task" which necessarily accomplishes most NLP tasks such as grammatical analysis and lexicography in linguistic studies, or machine translation, man-machine communication, message understanding in language understanding applications [15]. Besides these directly language oriented applications, WSD also have potential uses in other applications involving knowledge engineering such as information retrieval, information extraction and text mining, and particularly is recently beginning to be applied in the topics of named-entity classification, co-reference determination, and acronym expansion (cf. [1,4,6,10,38]).

So far, many approaches have been proposed for WSD in the literature. From a machine learning point of view, WSD is basically a classification problem and therefore it can directly benefit by the recent achievements from the machine learning community. As we have witnessed during the last two decades, many machine learning techniques and algorithms have been applied for WSD, including Naive Bayesian (NB) model, decision trees, exemplar-based model, support vector machines (SVM), maximum entropy models (MEM), etc. [1,24]. On the other hand, as observed in studies of classification systems, the set of patterns misclassified by different learning algorithms or techniques would not necessarily overlap [18]. This means that different classifiers may potentially offer complementary information about patterns to be classified. In other words, features and classifiers of different types complement one another in classification performance. This observation highly motivated the interest in combining classifiers to build an ensemble classifier which would improve the performance of the individual classifiers. Particularly, classifier combination for WSD has been received considerable attention recently from the community as well, e.g. [11,12,14,16,19–21,32,39].

Typically, there are two scenarios of combining classifiers mainly used in the literature [18]. The first approach is to use different learning algorithms for different classifiers operating on the same representation of the input pattern or on the same single data set, while the second approach aims to have all

classifiers using a single learning algorithm but operating on different representations of the input pattern or different subsets of instances of the training data. In the context of WSD, the work by Klein et al. [19], Florian and Yarowsky [12], and Escudero et al. [11] can be grouped into the first scenario. Whilst the studies given in [20,21,32] can be considered as belonging to the second scenario. Also, Wang and Matsumoto [39] used similar sets of features as in [32] and proposed a new voting strategy based on kNN method.

In addition, an important research issue in combining classifiers is what combination strategy should be used to derive an ensemble classifier. In [18], the authors proposed a common theoretical framework for combining classifiers which leads to many commonly used decision rules used in practice. Their framework is essentially based on the Bayesian theory and well-known mathematical approximations which are appropriately used to obtain other decision rules from the two basic combination schemes. On the other hand, when the classifier outputs are interpreted as evidence or belief values for making the classification decision, Dempster's combination rule in the Dempster-Shafer theory of evidence (D-S theory, for short) offers a powerful tool for combining evidence from multiple sources of information for decision making [2,3,9,8,21,34,40]. Despite the differences in approach and interpretation, almost D-S theory based methods of classifier combination assume the involved individual classifiers providing fully reliable sources of information for identifying the label of a particular input pattern. In other words, the issue of weighting individual classifiers in D-S theory based classifier combination has been ignored in previous studies. However, by observing that it is not always the case that all individual classifiers involved in a combination scenario completely agree on the classification decision, each of these classifiers does not by itself provide 100% certainty as the whole piece of evidence for identifying the label of the input pattern, therefore it should be weighted somehow before building a consensus decision. Fortunately, this weighting process can be modeled in D-S theory by the so-called discounting operator.

In this paper, we present a new method of weighting individual classifiers in which the weight associated with each classifier is defined adaptively depending on the input pattern under classification, making use of the measure of Shannon entropy. Intuitively, the higher ambiguity the output of a classifier is, the lower weight it is assigned and then the lesser important role it plays in the combination. Then by considering the problem of classifier combination as that of weighted combination of evidence for decision making, we develop a combination algorithm based on the discounting-and-combination scheme in D-S theory of evidence to derive a consensus decision for WSD. As for experimental results, we also conduct two typical scenarios of combination as briefly mentioned above: In the first scenario, different learning methods are used for different classifiers operating on the same representation of the context corresponding to a given polysemous word; in the second scenario all classifiers use

the same learning algorithm, namely NB, but operating on different representations of the context as considered in [21]. These combination scenarios are experimentally tested on English lexical samples of Senseval-2 and Senseval-3, resulting in an improvement in overall correctness.

The rest of this paper is organized as follows. Section 2 will begin with a brief introduction to basic notions from D-S theory of evidence and then follows by a short review of the related studies of classifier combination using D-S theory. Section 3 devotes to the D-S theory based framework for weighted combination of classifiers in WSD. The experimental results are presented and analyzed in Section 4. Finally, Section 5 presents some concluding remarks.

2 Background and Related Work

In this section we briefly review basic notions of D-S theory of evidence and its applications in ensemble learning studied previously.

2.1 Basic of Dempster-Shafer Theory of Evidence

The Dempster-Shafer (D-S) theory of evidence, originated from the work by Dempster [7] and then developed by Shafer [36], has appeared as one of the most popular theories for modeling and reasoning with uncertainty and imprecision. In D-S theory, a problem domain is represented by a finite set Θ of mutually exclusive and exhaustive hypotheses, called *frame of discernment* [36]. In the standard probability framework, all elements in Θ are assigned a probability, and when the degree of support for an event is known, the remainder of the support is automatically assigned to the negation of the event. On the other hand, in D-S theory the mass assignment representing evidence is carried out for events as it knows, and committing support for an event does not necessarily imply that the remaining support is committed to its negation. Formally, a *basic probability assignment* [1] (BPA, for short) is a function $m : 2^{\Theta} \rightarrow [0, 1]$ satisfying

$$m(\emptyset) = 0, \text{ and } \sum_{A \in 2^{\Theta}} m(A) = 1$$

The quantity $m(A)$ can be interpreted as a measure of the belief that is committed exactly to A, given the available evidence. A subset $A \in 2^{\Theta}$ with $m(A) > 0$ is called a *focal element* of m. A BPA m is called to be *vacuous* if $m(\Theta) = 1$ and $m(A) = 0$ for all $A \neq \Theta$.

[1] Also called a *mass function*

A belief function on Θ is defined as a mapping $\text{Bel} : 2^\Theta \to [0,1]$ which satisfies $\text{Bel}(\emptyset) = 0$, $\text{Bel}(\Theta) = 1$ and for any finite family $\{A_i\}_{i=1}^n$ in 2^Θ, we have

$$\text{Bel}(\bigcup_{i=1}^{n} A_i) \geq \sum_{\emptyset \neq I \subseteq \{1,\ldots,n\}} (-1)^{|I|+1} \text{Bel}(\bigcap_{i \in I} A_i)$$

Given a belief function Bel, a plausibility function Pl is then defined by $\text{Pl}(A) = 1 - \text{Bel}(\neg A)$. In D-S theory, belief and plausibility functions are often derived from a given BPA m, denoted by Bel_m and Pl_m respectively, which are defined as follows

$$\text{Bel}_m(A) = \sum_{\emptyset \neq B \subseteq A} m(B), \text{ and } \text{Pl}_m(A) = \sum_{A \cap B \neq \emptyset} m(B)$$

The difference between $m(A)$ and $Bel_m(A)$ is that while $m(A)$ is our belief committed to the subset A excluding any of its proper subsets, $Bel_m(A)$ is our degree of belief in A as well as all of its subsets. Consequently, $Pl_m(A)$ represents the degree to which the evidence fails to refute A. Note that all the three functions are in an one-to-one correspondence with each other. In other words, any one of these conveys the same information as any of the other two.

Two useful operations that especially play an important role in the evidential reasoning are *discounting* and *Dempster's rule of combination* [36]. The discounting operation is used when a source of information provides a BPA m, but knowing that this source has probability α of reliability. Then one may adopt $(1 - \alpha)$ as one's *discount rate*, resulting in a new BPA m^α defined by

$$m^\alpha(A) = \alpha \times m(A), \quad \text{for any } A \subset \Theta \tag{1}$$
$$m^\alpha(\Theta) = (1 - \alpha) + \alpha \times m(\Theta) \tag{2}$$

Consider now two pieces of evidence on the same frame Θ represented by two BPAs m_1 and m_2. Dempster's rule of combination is then used to generate a new BPA, denoted by $(m_1 \oplus m_2)$ (also called the orthogonal sum of m_1 and m_2), defined as follows

$$\begin{aligned} (m_1 \oplus m_2)(\emptyset) &= 0, \\ (m_1 \oplus m_2)(A) &= \tfrac{1}{1-\kappa} \sum_{B \cap C = A} m_1(B) m_2(C) \end{aligned} \tag{3}$$

where

$$\kappa = \sum_{B \cap C = \emptyset} m_1(B) m_2(C) \tag{4}$$

Note that the orthogonal sum combination is only applicable to such two BPAs that verify the condition $\kappa < 1$.

Since its inception, the D-S theory has been widely used in reasoning with uncertainty and information fusion in intelligent systems. Particularly, its applications to classifier combination has received attention since early 1990s, e.g., [2,3,21,34,40].

In the context of single-class classification problem, the frame of discernment is often modeled by the set of all possible classes or labels used to assign to an input pattern, where each pattern is assumed belonging to one and only one class. Formally, let $\mathcal{C} = \{c_1, c_2, \ldots, c_M\}$ be the set of classes, which is called the frame of discernment of the problem. Assume that we have R classifiers, denoted by $\{\psi_1, \ldots, \psi_R\}$, participating in the combination process. Given an input pattern $\mathbf{x}$, each classifier ψ_i produces an output $\psi_i(\mathbf{x})$ defined as

$$\psi_i(\mathbf{x}) = [s_{i1}, \ldots, s_{iM}] \tag{5}$$

where s_{ij} indicates the degree of confidence or support in saying that "*the pattern* $\mathbf{x}$ *is assigned to class* c_j *according to classifier* ψ_i." Note that s_{ij} can be a binary value or a continuous numeric value and its semantic interpretation depends on what type of learning algorithm used to build ψ_i. In the following we present briefly an overview of related works in classifier combination using D-S theory.

In [40], Xu et al. actually explored three different schemes for combining classifiers based on voting principle, Bayesian formalism and D-S theory, respectively. In particularly, their method of combination using D-S formalism assumes that each individual classifier produces a crisp decision on classifying an input $\mathbf{x}$, which is used as the evidence come from the corresponding classifier. Then this evidence is associated with prior knowledge defined in terms of performance indexes of the classifier to define its corresponding PBA, where performance indexes of a classifier are defined by recognition, substitution and rejection rates obtained by testing the classifier on a test sample set. Formally, assume that the recognition rate and the substitution rate of ψ_i are ϵ_r^i and ϵ_s^i (usually $\epsilon_r^i + \epsilon_s^i < 1$, due to the rejection action), respectively, Xu et al. defined a BPA m_i from $\psi_i(\mathbf{x})$ as following:

(1) If ψ_i rejected $\mathbf{x}$, i.e. $\psi_i(\mathbf{x}) = [0, \ldots, 0]$, m_i has only a focal element $\mathcal{C}$ with $m_i(\mathcal{C}) = 1$.
(2) If $\psi_i(\mathbf{x}) = [0, \ldots, 0, s_{ij} = 1, 0, \ldots, 0]$, then $m_i(\{c_j\}) = \epsilon_r^i$, $m_i(\neg\{c_j\}) = \epsilon_s^i$, where $\neg\{c_j\} = \mathcal{C} \setminus \{c_j\}$, and $m_i(\mathcal{C}) = 1 - \epsilon_r^i - \epsilon_s^i$.

In a similar way one can obtain all BPAs m_i $(i = 1, \ldots, R)$ from R classifiers ψ_i $(i = 1, \ldots, R)$. Then Dempster's rule (3) is applied to combine these BPAs to obtain a combined BPA $m = m_1 \oplus \ldots \oplus m_R$, which is used to make the

final decision on the classification of $\mathbf{x}$.

Rogova in [34] developed a D-S theory based model for combining the results of neural network classifiers. In general, the author used a proximity measure between a reference vector of each class and a classifier's output vector, where the reference vector is the mean vector μ_j^i of the output set of each classifier ψ_i for each class c_j. Then, for any input pattern $\mathbf{x}$, the proximity measures $d_j^i = \phi(\mu_j^i, \psi_i(\mathbf{x}))$ are transformed into the following PBAs:

$$m_i(\{c_j\}) = d_j^i, \qquad m_i(\mathcal{C}) = 1 - d_j^i \tag{6}$$

$$m_{\neg i}(\neg\{c_j\}) = 1 - \prod_{k \neq j}(1 - d_k^i), \qquad m_{\neg i}(\mathcal{C}) = \prod_{k \neq j}(1 - d_k^i) \tag{7}$$

which together constitute the knowledge about c_j and hence are combined to define the evidence from classifier ψ_i on classifying $\mathbf{x}$ as $m_i \oplus m_{\neg i}$. Finally, all evidences from all classifiers are combined using Dempster's rule to obtain an overall BPA for making the final decision on the classification.

Somewhat similar to Rogova's method, Al-Ani and Deriche [2] recently proposed a new technique for combining classifiers using D-S theory, in which different classifiers correspond to different feature sets. In their approach, the distance between the output classification vector provided by each single classifier and a reference vector is used to estimate BPAs. These BPAs are then combined making use of Dempster's rule of combination to obtain a new output vector that represents the combined confidence in each class label. However, instead of defining a reference vector as the mean vector of the output set of a classifier for a class as in Rogova's work, it is measured such that the mean square error (MSE) between the new output vector obtained after combination and the target vector of a training data set is minimized. This interestingly makes their combination algorithm trainable. Formally, given an input $\mathbf{x}$ the BPA m_i derived from classifier ψ_i is defined as follows:

$$m_i(\{c_j\}) = \frac{d_i^j}{\sum_{k=1}^{M} d_i^k + g_i} \tag{8}$$

$$m_i(\mathcal{C}) = \frac{g_i}{\sum_{k=1}^{M} d_i^k + g_i} \tag{9}$$

where $d_i^j = \exp(-\|\mathbf{v}_j^i - \psi_i(\mathbf{x})\|^2)$, $\mathbf{v}_j^i$ is a reference vector and g_i is a coefficient. Both of $\mathbf{v}_j^i$ and g_i will be estimated via the minimized MSE learning process, see [2] for more details.

More recently, Bell et al. [3] have developed a new method and technique for representing and combining outputs from different classifiers for text categorization based on D-S theory. Different from all the above mentioned methods,

the authors directly used outputs of individual classifiers to define the so-called 2-points focused mass functions which are then combined using Dempster's rule of combination to obtain an overall mass function for making the final classification decision. Particularly, given an input $\mathbf{x}$ the output $\psi_i(\mathbf{x})$ from classifier ψ_i is normalized first to obtain a probability distribution p_i over $\mathcal{C}$ as follows

$$p_i(c_j) = \frac{s_{ij}}{\sum_{k=1}^{M} s_{ik}}, \text{ for } j = 1, \ldots, M \tag{10}$$

Then the collection $\{p_i(c_j)\}_{j=1}^{M}$ is arranged so that

$$p_i(c_{i_1}) \geq p_i(c_{i_2}) \geq \ldots \geq p_i(c_{i_M}) \tag{11}$$

Finally, a BPA m_i represented the evidence from ψ_i on the classification of $\mathbf{x}$ is defined by

$$m_i(\{c_{i_1}\}) = p_i(\{c_{i_1}\}) \tag{12}$$
$$m_i(\{c_{i_2}\}) = p_i(\{c_{i_2}\}) \tag{13}$$
$$m_i(\mathcal{C}) = 1 - m_i(\{c_{i_1}\}) - m_i(\{c_{i_2}\}) \tag{14}$$

This mass function is called the 2-points focused mass function and the set $\{\{c_{i_1}\}, \{c_{i_2}\}, \mathcal{C}\}$ is referred to as a triplet. Basically, Bell et al. discarded classes appearing in the list (11) from the third and the sum of their degrees of support considered as noise are treated as ignorance, i.e. it is assigned to the frame of discernment $\mathcal{C}$.

Another recent attempt has been made in [21] to develop a method for weighted combination of classifiers for WSD based on D-S theory. Considering various ways of using context in WSD as distinct representations of a polysemous word under consideration, Le et al. [21] built NB classifiers corresponding to these distinct representations of the input and then weighted them by their accuracies obtained by testing with a test sample set, where weighting is modeled by the *discounting operator* in D-S theory. Finally, discounted BPAs are combined to obtain the final BPA which is used for making the classification decision. Formally, let $\mathbf{f}_i$ be the i-th representation of an input $\mathbf{x}$ and classifier ψ_i building on $\mathbf{f}_i$ produces a posterior probability distribution $P(\cdot|\mathbf{f}_i)$ on $\mathcal{C}$. Assume that α_i is the weight of ψ_i defined by its accuracy. Then the piece of evidence represented by $P(\cdot|\mathbf{f}_i)$ should be discounted at a discount rate of $(1 - \alpha_i)$, resulting in a BPA m_i defined by

$$m_i(\{c_j\}) = \alpha_i \times P(c_j|\mathbf{f}_i), \text{ for } j = 1, \ldots, M \tag{15}$$
$$m_i(\mathcal{C}) = 1 - \alpha_i \tag{16}$$

This method of weighting clearly focuses on only the strength of individual classifiers, which is defined by testing them on the designed sample data set

and therefore does not be influenced by an input pattern under classification. However, the information quality of soft decisions or outputs provided by individual classifiers might vary from pattern to pattern. In the following section, we propose a new method of adaptively weighting individual classifiers based on ambiguity measures associated with their outputs corresponding to a particular pattern under consideration. Roughly speaking, the higher ambiguity the output of a classifier is, the lower weight it is assigned. It is worth emphasizing again that both weighting and combining processes could be modeled within the developed framework of classifier combination using evidential operations.

3 Weighted Combination of Classifiers in D-S Formalism

Let us return to the classification problem with M classes $\mathcal{C} = \{c_1, \ldots, c_M\}$. Also assume that we have R classifiers ψ_i $(i = 1, \ldots, R)$, built using different R learning algorithms or different R representations of patterns. For each input pattern $\mathbf{x}$, let us denote by

$$\psi_i(\mathbf{x}) = [s_{i1}(\mathbf{x}), \ldots, s_{iM}(\mathbf{x})]$$

the soft decision or output given by ψ_i for the task of assigning $\mathbf{x}$ into one of M classes c_j. If the output $\psi_i(\mathbf{x})$ is not a posterior probability distribution on $\mathcal{C}$, it can be normalized to obtain an associated probability distribution defined by (10) above as done in [3]. Thus, in the following we always assume that $\psi_i(\mathbf{x})$ is a probability distribution on $\mathcal{C}$.

Each probability distribution $\psi_i(\mathbf{x})$ is now considered as the belief quantified from the information source provided by classifier ψ_i for classifying $\mathbf{x}$. However, this information does not by itself provide 100% certainty as a complete evidence sufficiently for making the classification decision. Therefore, it may be helpful to quantify somehow the quality of information offering from ψ_i regarding the classification of $\mathbf{x}$ and to take this measure into account when combining classifiers. Intuitively, if the uncertainty associated with $\psi_i(\mathbf{x})$ is high, it would make us more ambiguous in the decision made solely using $\psi_i(\mathbf{x})$ and then, the role it plays in the combination should be less important. This intuition suggests us a way of defining weights associated with classifiers using the measure of Shannon entropy as following.

For the sake of clarity, let us denote $m_i(\cdot|\mathbf{x})$ the probability distribution $\psi_i(\mathbf{x})$ on C, i.e. $m_i(c_j|\mathbf{x}) = s_{ij}(\mathbf{x})$. Then the weight associated with ψ_i regarding the classification of $\mathbf{x}$ is defined by

$$w_i(\mathbf{x}) = 1 - \frac{H(m_i(\cdot|\mathbf{x}))}{\log(M)} \tag{17}$$

where H is Shannon entropy expression of the probability distribution $m_i(\cdot|\mathbf{x})$, i.e.,

$$H(m_i(\cdot|\mathbf{x})) = -\sum_{j=1}^{M} m_i(c_j|\mathbf{x}) \log(m_i(c_j|\mathbf{x}))$$

Note that the definition of a classifier weight by (17) essentially depends on the input $\mathbf{x}$ under consideration, then the weight of an individual classifier can vary differently from pattern to pattern depending on how ambiguity associated with its decision on the classification of a particular pattern.

Now our aim is to combine all pieces of evidence $m_i(\cdot|\mathbf{x})$'s from individual classifiers ψ_i's on the classification of input $\mathbf{x}$, taking into account their weights $w_i(\mathbf{x})$'s respectively, to obtain an overall mass function $m(\cdot|\mathbf{x})$ on $\mathcal{C}$ for making the final classification decision. Formally, such an overall mass function $m(\cdot|\mathbf{x})$ can be formulated in the general form of the following:

$$m(\cdot|\mathbf{x}) = \bigoplus_{i=1}^{R} (w_i(\mathbf{x}) \otimes m_i(\cdot|\mathbf{x})) \tag{18}$$

where $\otimes$ is the discounting operator and $\oplus$ is a combination operator in general. Under such a general formulation, using two different combination operators in D-S theory we can obtain the following two decision rules for the classification of $\mathbf{x}$.

As mentioned in [36], an obvious way to use discounting with Dempster's rule of combination is to discount all mass functions $m_i(\cdot|\mathbf{x})$ $(i = 1, \ldots, R)$ at corresponding rates $(1 - w_i(\mathbf{x}))$ $(i = 1, \ldots, R)$ before combining them. This discounting-and-orthogonal sum combination strategy is carried out as follows.

First, from each mass function $m_i(\cdot|\mathbf{x}))$ and its associated weight $w_i(\mathbf{x})$, we obtain the corresponding discounted mass function, denoted by $m_i^w(\cdot|\mathbf{x})$, as follows:

$$m_i^w(\{c_j\}|\mathbf{x}) = w_i(\mathbf{x}) \times m_i(c_j|\mathbf{x})), \text{ for } j = 1, \ldots, M \tag{19}$$

$$m_i^w(\mathcal{C}|\mathbf{x}) = (1 - w_i(\mathbf{x})) \tag{20}$$

Then, Dempster's rule of combination allows us to combine all $m_i^w(\cdot|\mathbf{x})$ $(i = 1, \ldots, R)$ under the independent assumption of information sources for generating the overall mass function $m(\cdot|\mathbf{x})$. Note that, by definition, focal elements of each $m_i^w(\cdot|\mathbf{x})$ are either singleton sets [2] or the whole frame of discernment $\mathcal{C}$. It is easy to see that $m(\cdot|\mathbf{x})$ also verifies this property if applicable. Interestingly, the commutative and associative properties of the orthogonal sum operation with respect to a combinable collection of $m_i^w(\cdot|\mathbf{x})$'s $(i = 1, \ldots, R)$

[2] So, we write $m_i^w(c_j|\mathbf{x})$ instead of $m_i^w(\{c_j\}|\mathbf{x})$, without any danger of confusion.

and the mentioned property essentially form the basis for developing an efficient algorithm for calculation of the $m(\cdot|\mathbf{x})$ as described in the following algorithm.

Algorithm 1 The Combination Algorithm Using Dempster's Rule

Input: $m_i(\cdot|\mathbf{x})$ $(i = 1, \ldots, R)$

Output: $m(\cdot|\mathbf{x})$ – the combined mass function

1: Initialize $m(\cdot|\mathbf{x})$ by $m(\mathcal{C}|\mathbf{x}) = 1$, $m(c_j|\mathbf{x}) = 0$ for any $j = 1, \ldots, M$
2: **for** $i = 1$ to R **do**
3: Calculate $w_i(\mathbf{x})$ via (17)
4: Calculate $m_i^w(\cdot|\mathbf{x})$ via (19) and (20)
5: Compute the combination $m \oplus m_i^w(\cdot|\mathbf{x})$ via (21) and (22)
6: Put $m(\cdot|\mathbf{x}) := m \oplus m_i^w(\cdot|\mathbf{x})$
7: **end for**
8: **return** $m(\cdot|\mathbf{x})$

$$
\begin{aligned}
m \oplus m_i^w(c_j|\mathbf{x}) = & \frac{1}{\kappa_i}[m(c_j|\mathbf{x}) \times m_i^w(c_j|\mathbf{x}) + m(c_j|\mathbf{x}) \times m_i^w(\mathcal{C}|\mathbf{x}) \\
& + m(\mathcal{C}|\mathbf{x}) \times m_i^w(c_j|\mathbf{x})], \text{ for } j = 1, \ldots, M \qquad (21) \\
m \oplus m_i^w(\mathcal{C}|\mathbf{x}) = & \frac{1}{\kappa_i}(m(\mathcal{C}|\mathbf{x}) \times m_i^w(\mathcal{C}|\mathbf{x})) \qquad (22)
\end{aligned}
$$

where κ_i is a normalizing factor defined by

$$
\kappa_i = \left[1 - \sum_{j=1}^{M} \sum_{\substack{k=1 \\ k \neq j}}^{M} m(c_j|\mathbf{x}) \times m_i^w(c_k|\mathbf{x}) \right] \qquad (23)
$$

Finally, the mass function $m(\cdot|\mathbf{x})$ is used to make the final classification decision according to the following decision rule:

$$
\mathbf{x} \text{ is assigned to the class } c_{k^*}, \text{ where } k^* = \arg\max_j m(c_j|\mathbf{x}) \qquad (24)
$$

It would be interesting to note that an issue may arise with the orthogonal sum operation is in using the total probability mass κ associated with conflict as defined in the normalization factor. Consequently, applying it in an aggregation process may yield counterintuitive results in the face of significant conflict in certain situations as pointed out in [42]. Fortunately, in the context of the weighted combination of classifiers, by discounting all $m_i(\cdot|\mathbf{x})$ $(i = 1, \ldots, R)$ at corresponding rates $(1 - w_i(\mathbf{x}))$ $(i = 1, \ldots, R)$, we actually reduce conflict between the individual classifiers before combining them.

Now, instead of using Dempster's rule of combination after discounting $m_i(\cdot|\mathbf{x})$ as above, we apply the averaging operation [3] over discounted mass functions $m_i^w(\cdot|\mathbf{x})$ $(i = 1, \ldots, R)$ to obtain the mass function $m(\cdot|\mathbf{x})$ defined by

$$m(c_j|\mathbf{x}) = \frac{1}{R}\sum_{i=1}^{R} w_i(\mathbf{x}) \times m_i(c_j|\mathbf{x}), \text{ for } j = 1, \ldots, M \tag{25}$$

$$m(\mathcal{C}|\mathbf{x}) = 1 - \frac{\sum_{i=1}^{R} w_i(\mathbf{x})}{R} \triangleq 1 - \overline{w}(\mathbf{x}) \tag{26}$$

Note that the probability mass unassigned to individual classes but the whole frame of discernment $\mathcal{C}$, $m(\mathcal{C}|\mathbf{x})$, is the average of discount rates. Therefore, if instead of allocating the average discount rate $(1 - \overline{w}(\mathbf{x}))$ to $m(\mathcal{C}|\mathbf{x})$ as above, we use $1 - m(\mathcal{C}|\mathbf{x}) = \overline{w}(\mathbf{x})$ as a normalization factor and then easily obtain

$$m(c_j|\mathbf{x}) = \frac{1}{\sum_{i=1}^{R} w_i(\mathbf{x})}\sum_{i=1}^{R} w_i(\mathbf{x}) \times m_i(c_j|\mathbf{x}), \text{ for } j = 1, \ldots, M \tag{27}$$

which interestingly turns out to be the weighted mixture of individual classifiers corresponding to the weighted sum decision rule.

In the following section we will conduct several experiments for WSD to test the proposed method of weighting classifiers with two typical scenarios of combination as mentioned previously.

4 An Experimental Study for WSD

4.1 Individual Classifiers in Combination

In the first scenario of combination, we used three well-known statistical learning methods including the Naive Bayes (NB), Maximum Entropy Model (MEM), and Support Vector Machines (SVM). The selection of individual classifiers in this scenario is basically guided by the direct use of output results for defining mass functions in the present work. Clearly, the first two classifiers produce classified outputs which are probabilistic in nature. Although a standard SVM classifier does not provide such probabilistic outputs, the issue of mapping SVM outputs into probabilities has been studied [33] and recently become popular for applications requiring posterior class probabilities [3,26].

[3] Note that this averaging operation was also mentioned briefly by Shafer [36] for combining belief functions.

We have used the library implemented for maximum entropy classification available at [37] for building the MEM classifier. Whilst the SVM classifier is built based upon LIBSVM implemented by Chang and Lin [5], which has the ability to deal with the multiclass classification problem and output classified results as posterior class probabilities.

In the second scenario of combination, we used the same NB learning algorithm for individual classifiers, however, each of which has been built using a distinct set of features corresponding to a distinct representation of a polysemous word to be disambiguated. It is of interest noting that NB is commonly accepted as one of learning methods represents state-of-the-art accuracy on supervised WSD [11]. In particularly, given a polysemous word $\mathbf{w}$, which may have M possible senses (classes): c_1, c_2,..., c_M, in a context C, the task is to determine the most appropriate sense of $\mathbf{w}$. Generally, context C can be used in two ways [15]: in the *bag-of-words approach*, the context is considered as words in some window surrounding the target word $\mathbf{w}$; in the *relational information based approach*, the context is considered in terms of some relation to the target such as distance from the target, syntactic relations, selectional preferences, phrasal collocation, semantic categories, etc. As such, different views of context may provide different ways of representing context C. Assume we have such R representations of C, say $\mathbf{f}_1, \ldots, \mathbf{f}_R$, serving for the aim of identifying the right sense of the target $\mathbf{w}$. Then we can build R individual classifiers, where each representation $\mathbf{f}_i$ is used by the corresponding i-th classifier. In our experiments, six different representations of context explored in [21] are used for this purpose.

4.2 Representations of Context for WSD

The context representation plays an essentially important role in WSD. For predicting senses of a word, information usually used in previous studies is the topic context which is represented as bag of words. In [31], Ng and Lee proposed to use more linguistic knowledge resources that then became popular for determining word sense in many studies later on. The knowledge resources used in their paper included topic context, collocation of words, and a syntactic relationship verb-object. In [23], the authors use another information type, which is words or part-of-speech and each is assigned with its position in relation with the target word. In classifier combination for WSD, topical context with different sizes of context windows is usually used for creating different representations of a polysemous word, such as in Pedersen [32] and Wang and Matsumoto [39].

As observed in [21], two of the most important information sources for determining the sense of a polysemous word are the topic of context and relational

information representing the structural relations between the target word and the surrounding words in a local context. Under such an observation, the authors have experimentally designed four kinds of representation with six feature sets defined as follows: $\mathbf{f}_1$ is a set of collocations of words; $\mathbf{f}_2$ is a set of words assigned with their positions in the local context; $\mathbf{f}_3$ is a set of part-of-speech tags assigned with their positions in the local context; $\mathbf{f}_4, \mathbf{f}_5$ and $\mathbf{f}_6$ are sets of unordered words in the large context with different windows: small, median and large respectively. Symbolically, we have

$$\begin{aligned}
\mathbf{f}_1 &= \{\mathbf{w}_{-l} \cdots \mathbf{w}_{-1}\mathbf{w}\mathbf{w}_1 \cdots \mathbf{w}_r | l + r \leq n_1\} \\
\mathbf{f}_2 &= \{(\mathbf{w}_{-n_2}, -n_2), \ldots, (\mathbf{w}_{-1}, -1), (\mathbf{w}_1, 1), \ldots, (\mathbf{w}_{n_2}, n_2)\} \\
\mathbf{f}_3 &= \{(p_{-n_3}, -n_3), \ldots, (p_{-1}, -1), (p_1, 1), \ldots, (p_{n_3}, n_3)\} \\
\mathbf{f}_i &= \{\mathbf{w}_{-n_i}, \ldots, \mathbf{w}_{-2}, \mathbf{w}_{-1}, \mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_{n_i}\} \text{ for } i = 4, 5, 6
\end{aligned}$$

where $\mathbf{w}_i$ is the word at position i in the context of the ambiguous word $\mathbf{w}$ and p_i be the part-of-speech tag of $\mathbf{w}_i$, with the convention that the target word $\mathbf{w}$ appears precisely at position 0 and i will be negative (positive) if $\mathbf{w}_i$ appears on the left (right) of $\mathbf{w}$. Here, we set $n_1 = 3$ (maximum of collocations), $n_2 = 5$, $n_3 = 5$ (windows size for local context), and for topic context, three different window sizes are used: $n_4 = 5$ (small), $n_5 = 10$ (median), and $n_6 = 100$ (large). Topical context is represented by a set of content words that includes nouns, verbs and adjectives in a certain window size. Note that after these words being extracted, they will be converted into their root morphology forms for use. It has been shown that these representations for the individual classifiers are richer than the representation that just used the words in context because the feature containing richer information about structural relations is also utilized. Even the unordered words in a local context may contain structure information as well, collocations and words as well as part-of-speech tags assigned with their positions may bring richer information.

4.3 Test Data

Concerning evaluation exercises in automatic WSD, three corpora so-called Senseval-1, Senseval-2 and Senseval-3 have been built on the occasion of three corresponding workshops held in 1998, 2001, and 2004 respectively. There are different tasks in these workshops with respect to different languages and/or the objectives of disambiguating single-word or all-words in the input. In this paper, the investigated combination rules will be tested on English lexical samples of Senseval-2 and Senseval-3. These two datasets are more precise than the one in Senseval-1 and widely used in current WSD studies.

A total of 73 nouns, adjectives, and verbs are chosen in Senseval-2 with the sense inventory is taken from WordNet 1.7. The data came primarily from

the Penn Treebank II corpus, but was supplemented with data from the British National Corpus whenever there was an insufficient number of Treebank instances (see [17] for more detail). Examples in English lexical sample of Senseval-3 are extracted from the British National Corpus. The sense inventory used for nouns and adjectives is taken from WordNet 1.7.1, which is consistent with the annotations done for the same task during Senseval-2. Verbs are instead annotated with senses from Wordsmyth [4]. There are 57 nouns, adjectives, and verbs in this data (see [28] for more detail).

In these datasets, each polysemous word is associated with its corresponding training dataset and test dataset. The training dataset contains sense-tagged examples, i.e. in each example the polysemous word is assigned with the right sense. The test dataset contains sense-untagged examples, and the evaluation is based on a key-file, i.e. the right senses of these test examples are listed in this file. The evaluation used here follows the proposal in [27], which provides a scoring method for exact matches to fine-grained senses as well as one for partial matches at a more coarse-grained level. Note that, like most related studies, the fine-grained score is computed in the following experiments.

4.4 Experimental Results

Firstly, Table 1 and Table 2 provide the experimental results obtained by using the entropy-based method of weighting classifiers and two strategies of weighted combination as discussed in Section 3 for two scenarios of combination. In these tables, WDS_1 and WDS_2 stand for two combination methods which apply the discounting-and-orthogonal sum combination strategy and the discounting-and-averaging combination strategy, respectively. In Table 2, C_i $(i = 1, \ldots, 6)$ respectively represent six individual classifiers corresponding to the six feature sets $\mathbf{f}_i$ $(i = 1, \ldots, 6)$. The obtained results show that in both cases combined classifiers always outperform individual classifiers participating in the corresponding combination. Especially, in the second scenario of combination both combined classifiers WDS_1 and WDS_2 strongly dominate all individual classifiers. Note that all representations of context used to build individual classifiers in the second scenario have been utilized jointly for defining a unique representation of context commonly used for individual classifiers in the first scenario. This would interpret why individual classifiers in the first scenario also provide results much better than individual classifiers in the second scenario and slightly inferior to corresponding WDS_1 and WDS_2.

It is also interesting to see that in both scenarios of combination, the results yielded by the discounting-and-averaging combination strategy (i.e., WDS_2) are comparable or even better than that given by the discounting-and-orthogonal

[4] http://www.wordsmyth.net/

sum combination strategy (i.e., WDS_1), while the former is computational more simple than the latter. Although the averaging operation was actually mentioned briefly by Shafer [36] for combining belief functions, it has been almost completely ignored in the studies of information fusion and particularly classifier combination with D-S theory. Interestingly also, Shafer [36] did show that discounting in fact turns combination into averaging when all the information sources being combined are highly conflicting and have been sufficiently discounted. This might, intuitively, provide an interpretation for a good performance of WDS_2.

Table 1
Experimental results for the first scenario of combination

%	Individual classifiers			Combined classifiers	
	NB	MEM	SVM	WDS_1	WDS_2
Senseval-2	65.6	65.5	63.5	66.3	**66.5**
Senseval-3	72.9	72.0	72.5	**73.3**	**73.3**

Table 2
Experimental results for the second scenario of combination

%	Individual classifiers						Combined classifiers	
	C_1	C_2	C_3	C_4	C_5	C_6	WDS_1	WDS_2
Senseval-2	56.7	54.6	54.7	56.8	56.8	52.5	64.4	**65.0**
Senseval-3	62.4	62.3	64.1	61.9	63.9	59.5	71.0	**72.3**

Secondly, to have a comparative view of obtained results, Table 3 provides an experimental comparison of overall performances of the developed framework of weighted combination of classifier for WSD with the best systems in the contests for the English lexical sample tasks of Senseval-2 [17] and Senseval-3 [28], respectively. Here, DS_1 is the method of weighted combination using Dempster's rule in which weights of individual classifiers are defined using their accuracies obtained by testing on a test sample set as proposed in [21]. The best system of Senseval-2 contest also used a combination technique: the output of subsystems (classifiers) which were built based on different machine learning algorithms were merged by using weighted and threshold-based voting and score combination (see [41] for the detail). The best system of Senseval-3 contest used the Regularized Least Square Classification (RLSC) algorithm with a correction of the a priori frequencies (refer to [13] for more details). Note that the methods using in these systems are also corpus-based methods.

Table 3
A comparison with the best systems in the contests of Senseval-2 and Senseval-3

%	The best system	Accuracy-based weighting DS_1	Adaptively weighting WDS_2
Senseval-2	64.2	64.7	**66.3**
Senseval-3	72.9	72.4	**73.3**

5 Conclusions

In this paper the Dempster-Shafer theory based framework for weighted combination of classifiers for WSD has been introduced. Within this framework, we have proposed a new method for defining adaptively weights of individual classifiers using entropy measures considered as ambiguity associated with their classified outputs. We have also discussed two combination strategies using evidential operations in Dempster-Shafer theory, which consequently resulted in two corresponding rules for deriving a consensus classification decision.

Experimentally, we have conducted two typical scenarios of classifier combination with the proposed weighting method and two developed combination methods, which were tested on English lexical samples of Senseval-2 and Senseval-3. The experimental result has shown that the discussed framework of weighted combination of classifiers using Dempster-Shafer theory have provided several decision combination methods for WSD that outperform the best systems in the contests of Senseval-2 and Senseval-3.

It seems that the entropy-based weighting method proposed in this paper along with the discussed framework of weighted combination of classifiers would be best appropriate to apply for integrating semi-supervised learning with classifier combination for WSD as studied recently in [22]. In the context of semi-supervised learning, the insufficiency of labeled data may influence the output quality of individual classifiers and then discounting them by their weights defined by the entropy-based weighting method would effectively contribute in improving the quality of combined classifiers. This, however, is left for the future work.

Acknowledgements

This work was partially supported by a Grant-in-Aid for Scientific Research (No. 20500202) from the Japan Society of the Promotion of Science (JSPS) and FY-2008 JAIST International Joint Research Grant. The authors would like to appreciate constructive comments and helpful suggestions from anonymous

referees, which have helped improving the presentation of the paper.

References

[1] E. Agirre, P. Edmonds (Eds.), *Word Sense Disambiguation: Algorithms and Applications* (Springer, Dordrecht, the Netherlands 2006).

[2] A. Al-Ani, M. Deriche, A new technique for combining multiple classifiers using the Dempster–Shafer theory of evidence, *Journal of Artificial Intelligence Research* **17** (2002) 333–361.

[3] D. Bell, J. W. Guan, Y. Bi, On combining classifiers mass functions for text categorization, *IEEE Transactions on Knowledge and Data Engineering* **17** (10) (2005) 1307–1319.

[4] S. Bloehdorn, H. Andreas, Text classification by boosting weak learners based on terms and concepts, *Proceedings of the fourth IEEE International Conference on Data Mining*, 2004, pp. 331–334.

[5] C. C. Chang, C. J. Lin, LIBSVM: A library for support vector machines, http://www.csie.ntu.edu.tw/cjlin/libsvm, 2001.

[6] P. Clough, M. Stevenson, Cross-language information retrieval using Euro WordNet and word sense disambiguation, *Proceedings of Advances in Information Retrieval, 26th European Conference on IR Research (ECIR)*, 2004, Sunderland, UK, pp. 327–337.

[7] A. P. Dempster, Upper and lower probabilities induced by a multi-valued mapping, *Annals of Mathematics and Statistics* **38** (1967) 325–339.

[8] T. Denoeux, A neural network classifier based on DempsterShafer theory, *IEEE Transactions on Systems, Man and Cybernetics A* **30** (2) (2000) 131–150.

[9] T. Denoeux, A k-nearest neighbor classification rule based on DempsterShafer theory, *IEEE Transactions on Systems, Man and Cybernetics* **25** (5) (1995) 804–813.

[10] S. Dill, N. Eiron, D. Gibson, D. Gruhl, R. Guha, A. Jhingran, T. Kanungo, S. Rajagopalan, A. Tomkins, J. A. Tomlin, J. Y. Zien., Semtag and seeker: bootstrapping the semantic web via automated semantic annotation. *In Proceedings of the Twelfth International Conference on World Wide Web*, 2003, pp. 178–186.

[11] G. Escudero, L. Màrquez, G. Rigau, Boosting applied to word sense disambiguation, *Proceedings of the 11th European Conference on Machine Learning*, 2000, pp. 129–141.

[12] R. Florian, D. Yarowsky, Modeling consensus: Classifier combination for word sense disambiguation, *Proceedings of EMNLP 2002*, pp. 25–32.

[13] C. Grozea. Finding optimal parameter settings for high performance word sense disambiguation, *Proceedings of ACL/SIGLEX Senseval-3*, Barcelona, Spain, July 2004, pp. 125–128.

[14] V. Hoste, I. Hendrickx, W. Daelemans, A. van den Bosch, Parameter optimization for machine-learning of word sense disambiguation, *Natural Language Engineering* **8** (3) (2002) 311–325.

[15] N. Ide, J. Véronis, Introduction to the Special Issue on Word Sense Disambiguation: The State of the Art, *Computational Linguistics* **24** (1998) 1–40.

[16] A. Kilgarriff, J. Rosenzweig, Framework and results for English SENSEVAL, *Computers and the Humanities* **36** (2000) 15–48.

[17] A. Kilgarriff, English lexical sample task description, *Proceedings of senseval-2: Second International Workshop on Evaluating Word Sense Disambiguation Systems*, 2001, Toulouse, France, pp. 17–20.

[18] J. Kittler, M. Hatef, R. P. W. Duin, J. Matas, On combining classifiers, *IEEE Transactions on Pattern Anal. and Machine Intell.* **20** (3) (1998) 226–239.

[19] D. Klein, K. Toutanova, H. Tolga Ilhan, S. D. Kamvar, C. D. Manning, Combining heterogeneous classifiers for word-sense disambiguation, *ACL WSD Workshop*, 2002, pp. 74–80.

[20] C. A. Le, V.-N. Huynh, A. Shimazu, An evidential reasoning approach to weighted combination of classifiers for word sense disambiguation, *MLDM 2005*, P. Perner, A. Imiya (Eds.), Springer-Verlag, LNCS **3587**, pp. 516–525.

[21] C. A. Le, V.-N. Huynh, A. Shimazu, Y. Nakamori, Combining classifiers for word sense disambiguation based on Dempster-Shafer theory and OWA operators, *Data & Knowledge Engineering* **63** (2) (2007) 381–396.

[22] A. C. Le, A. Shimazu, V.-N. Huynh, L. M. Nguyen, Semi-supervised learning integrated with classifier combination for word sense disambiguation, *Computer Speech and Language* **22** (4) (2008) 330–345.

[23] C. Leacock, M. Chodorow, G. Miller, Using corpus statistics and WordNet relations for sense identification, *Computational Linguistics* **24** (1) (1998) 147–165.

[24] Y. K. Lee and H. T. Ng, 2002. An Empirical Evaluation of Knowledge Sources and Learning Algorithms for Word Sense Disambiguation. *Proceedings of EMNLP*, pages 41–48.

[25] G. Leroy, T. C. Rindflesch, Effects of information and machine learning algorithms on word sense disambiguation with small datasets, *International Journal of Medical Informatics* **74** (7-8) (2005) 573–585.

[26] H.-T. Lin, C.-J. Lin, R. C. Weng, A note on Platts probabilistic outputs for support vector machines, *Machine Learning* **68** (2007) 267–276.

[27] I. D. Melamed, P. Resnik, Tagger evaluation given hierarchical tag sets, *Computers and the Humanities* **34** (1-2) (2000) 79–84.

[28] R. Mihalcea , T. Chklovski, A. Killgariff , The Senseval-3 English lexical sample task, *Proceedings of ACL/SIGLEX Senseval-3*, Barcelona, Spain, July 2004, pp. 25–28.

[29] A. Montoyo, A. Suarez, G. Rigau and M. Palomar, Combining knowledge- and corpus-based Word-Sense-Disambiguation methods, *Journal of Artificial Intelligence Research* **23** (2005) 299–330.

[30] R. J. Mooney, Comparative experiments on disambiguating word senses: An illustration of the role of bias in machine learning, *Proceedings of the EMNLP 1996*, pp. 82–91.

[31] H. T. Ng, H. B. Lee, Integrating multiple knowledge sources to disambiguate word sense: An exemplar-based approach, *Proceedings of the 34th Annual Meeting of the ACL*, 1996, pp. 40–47.

[32] T. Pedersen, A simple approach to building ensembles of Naive Bayesian classifiers for word sense disambiguation, *Proceedings of the North American Chapter of the ACL*, 2000, pp. 63–69.

[33] J. Platt, Probabilistic outputs for support vector machines and comparison to regularized likelihood methods. In A. Smola, P. Bartlett, B. Schölkopf, D. Schuurmans (Eds.), *Advances in Large Margin Classifiers* (Cambridge: MIT Press, 2000).

[34] G. Rogova, Combining the results of several neural network classifiers, *Neural Networks* **7** (5) (1994) 777–781.

[35] M. Sanderson, Word sense disambiguation and information retrieval, *Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1994, Dublin, Ireland, pp. 142–151.

[36] G. Shafer, *A Mathematical Theory of Evidence* (Princeton University Press, Princeton, 1976).

[37] Y. Tsuruoka, A simple C++ library for maximum entropy classification, http://www-tsujii.is.s.u-tokyo.ac.jp/∼tsuruoka/maxent/, 2006.

[38] P. Vossen, G. Rigau, I. Alegria, E. Agirre, D. Farwell, M. Fuentes, Meaningful results for Information Retrieval in the MEANING project, *Proceedings of Third International WordNet Conference*, Jeju Island, Korea, 2006.

[39] X. J. Wang, Y. Matsumoto, Trajectory based word sense disambiguation, *Proceedings of the 20th International Conference on Computational Linguistics*, Geneva, August 2004, pp. 903–909.

[40] L. Xu, A. Krzyzak, C. Y. Suen, Several methods for combining multiple classifiers and their applications in handwritten character recognition, *IEEE Transactions on Systems, Man and Cybernetics* **22** (1992) 418–435.

[41] D. Yarowsky, S. Cucerzan, R. Florian, C. Schafer, R. Wicentowski, The Johns Hopkins SENSEVAL2 System Descriptions, *Proceedings of SENSEVAL2*, 2001, pp. 163–166.

[42] L. A. Zadeh, Reviews of Books: A Mathematical Theory of Evidence, *The AI Magazine* **5** (1984) 81–83.