Artificial Intelligence 172 (2008) 1731–1751

Contents lists available at ScienceDirect

Artificial Intelligence

www.elsevier.com/locate/artint

The combination of multiple classifiers using an evidential reasoning approach

Yaxin Bi [a,*], Jiwen Guan [b], David Bell [b]

[a] School of Computing and Mathematics, University of Ulster at Jordanstown, Co Antrim, BT37 0QB, UK
[b] School of Computer Science, The Queen's University of Belfast, Belfast, BT7 1NN, UK

ARTICLE INFO

Article history:
Received 4 June 2007
Received in revised form 12 June 2008
Accepted 12 June 2008
Available online 19 June 2008

Keywords:
Ensemble methods
Dempster's rule of combination
Evidential reasoning
Evidential structures
Combination functions

ABSTRACT

In many domains when we have several competing classifiers available we want to synthesize them or some of them to get a more accurate classifier by a combination function. In this paper we propose a 'class-indifferent' method for combining classifier decisions represented by evidential structures called *triplet* and *quartet*, using Dempster's rule of combination. This method is unique in that it distinguishes important elements from the trivial ones in representing classifier decisions, makes use of more information than others in calculating the support for class labels and provides a practical way to apply the theoretically appealing Dempster–Shafer theory of evidence to the problem of ensemble learning. We present a formalism for modelling classifier decisions as triplet mass functions and we establish a range of formulae for combining these mass functions in order to arrive at a consensus decision. In addition we carry out a comparative study with the alternatives of *simplet* and *dichotomous structure* and also compare two combination methods, Dempster's rule and majority voting, over the UCI benchmark data, to demonstrate the advantage our approach offers.

1. Introduction

The idea characterizing ensemble learning is to learn and retain multiple classifiers and combine their decisions in some way in order to classify new instances [36]. The attraction of this approach in supervised machine learning is based on the premise that a combination of classifiers is often more accurate than an individual classifier. A theoretical explanation of its success is that different classifiers offer complementary information about instances to be classified which could be harnessed to improve the performance of the individual classifiers [27].

Generally speaking a successful ensemble method depends on two components: a set of appropriate classifiers and a combination method, function or scheme [30]. Classifiers assign single classes or sets of classes to a new instance along with respective numeric values as decisions, and a combination function merges these decisions in some way to determine a final decision—usually by voting among the decisions.

Ensemble classifiers can be generated in different ways. A typical approach is to use a single learning algorithm to operate on different subsets of attributes or instances of the training data, as done in bagging [9] and boosting [11,21], and in derivatives such as random forests [10] or the random subspace method for constructing decision tree forests [26]. Another approach is to use different learning algorithms to operate on a single data set [4,8,32]. Among any set of individual classifiers, some are more accurate for a given task and others are less accurate. However there is often not a dominant

* Corresponding author.
E-mail addresses: y.bi@ulster.ac.uk (Y. Bi), j.guan@qub.ac.uk (J. Guan), da.bell@qub.ac.uk (D. Bell).

0004-3702/$ – see front matter
doi:10.1016/j.artint.2008.06.002

one for the complete data distribution. By taking account of the strengths of classifiers through combination functions, the performance of the best individual classifier can be improved [12].

Kuncheva [28] roughly characterizes combination methods, based on the forms of classifier outputs, into two categories. In the first category, the combination of decisions is performed on single class labels, including majority voting and Bayesian probability, which have been extensively examined in the ensemble literature [18,27,40,49]. The second category is concerned with the utilization of continuous values corresponding to class labels. One set of methods, often called *class-aligned* methods, is based on using the same class labels from different classifiers in calculating the support for class labels, regardless of what the support for the other classes is. This method includes linear sum and order statistics, to which considerable effort has been devoted [25,27,47,48,51]. Another method, called stacked generalization or meta-learning, is to use continuous values of class labels as a set of features to learn a combination function in addition to a set of classifiers [19,46,54]. An alternative group of methods, which are called *class-indifferent* methods, is to make use of as much information as possible obtained from both single classes and sets of classes in calculating the support for each class [28].

Class-aligned methods and class-indifferent methods are both based on continuous values of class labels in calculating the support for class labels. A distinction between them is, however, that the latter takes impacts from different classes into account in determining the support for a class that permits the presence of uncertainty information—as happens when an instance is classified into different classes by different classifiers. Several related studies are presented in the literature, where class-indifferent methods utilize single classes and sets of classes [16,39,49]. Class-indifferent methods for combining decisions in the form of a list of ordered decisions have not been intensively studied and are poorly understood. In particular, little is known about the value of evidential reasoning methods for combining truncated lists of ordered decisions [5,8].

In this study we consider a class-indifferent approach to combining classifiers using Dempster's rule of combination. Our focus is on generating classifiers using different learning methods to manipulate a single data set, and the combination of classifiers is modeled as a process of reasoning under uncertainty. We model each output given by classifiers on new instances as a list of contender decisions and reduce it to subsets of 2 and 3 decisions, respectively, which are then represented by the evidential structures of *triplet* and *quartet* [5–8]. We first establish a formalism for combining triplets and quartets using Dempster's rule of combination to constrain the final decision, and then we empirically and analytically examine the effect of different sizes of decision lists on the combination of classifiers. Furthermore we justify the assumption we make that modelling classifier results as independent bodies of evidence is sensible.

The advantages of our approach are summarized as follows. The first advantage is that our method makes use of a wide range of evidence items in classification to make the final decision. The idea is inspired by the observation that if only the 'best' single class labels are selected on the basis of their corresponding values, valuable information contained in the discarded labels may be lost. Arguably, the potential loss of support from the other classes should be avoided by utilizing this support information in the decision making process. The evidence structures, such as the triplet, are able to distinguish the important classes from the trivial ones and incorporate the best-supported class, the second best-supported class, and the rest of the classes which are treated in terms of *ignorance* within the process of decision making. The second advantage is that these evidence structures provide an efficient way for combining many pieces of evidence since they break down a large list of contender decisions into smaller, more tractable subsets. Like the simplet and dichotomous structures [2,44], our method deals well with a long-standing criticism saying that the evidence theory does not translate easily into practical applications due to the computational complexity of combining multiple pieces of evidence.

To validate our method and illustrate its power, we have carried out numerous comparative experiments over the UCI data sets [3]. We experimentally compare the triplet and quartet with the alternatives of simplet, dichotomous structure and the full list of decisions. We also make a comparison between Dempster's rule and majority voting in combining classifiers. During the course of classifier combination, another important issue, namely the extent of agreement reached on classification decisions among classifiers, is assessed by means of κ statistics, and the associative property of combining triplets is also experimentally examined. Finally to explain our empirical findings, we present an investigation into the calculation process of Dempster's rule which provides an insight into the reason for superiority of our method.

The rest of the paper is organized as follows. Section 2 presents the representation of classifier outputs and the idea of class-independent methods. Section 3 reviews the Dempster–Shafer theory of evidence. Section 4 presents a review of several related studies with a focus on the alternative structures of simplet and dichotomous structure previously used in combining classifiers. The rationale of the evidential structure of the triplet, and the associated formulae, are presented in Section 5. The combination functions of Dempster's rule with different evidential structures and majority voting for combining classifiers are evaluated and the experimental settings and results are detailed in Section 6. Section 7 presents a discussion about the advantage of the triplet and the quartet over the alternatives. Section 8 gives a justification for the independence of evidence derived from classifier outputs. The concluding summary is given in Section 9.

2. Representation of classifier outputs and combination methods

In supervised machine learning, a learning algorithm is provided with training instances of the form $\{\langle d_1, c_1\rangle, \ldots, \langle d_{|D|}, c_q\rangle\}$ $(d_i \in D, c_i \in C, 1 \leqslant q \leqslant |C|)$ for inducing some unknown function f such that $f(d) = c$. D is the space of attribute vectors and each vector d_i is in the form $(w_{i_1}, \ldots, w_{i_n})$ whose components are symbolic or numeric values; C is a set of categorical classes and each class c_i is in the form of class label. Given a set of training data, a learning algorithm

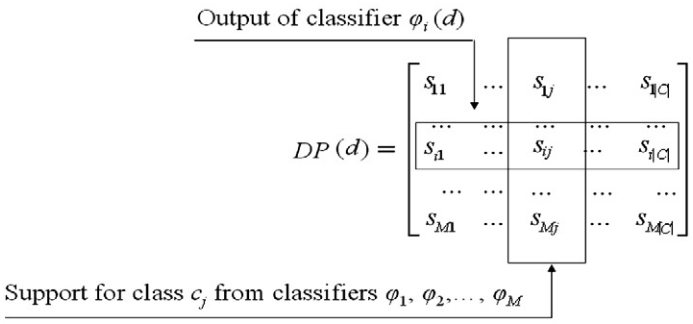

Fig. 1. A decision profile for instance d generated by $\varphi_1(d), \varphi_2(d), \ldots, \varphi_M(d)$.

is aimed at learning a function φ—a classifier from the training data. The classifier φ is an approximation to the unknown function f.

Given a new instance d, a classification task is to decide, using φ, whether instance d belongs to class c_i. In a multi-class assignment, we denote such a process as a mapping:

$$\varphi : D \to C \times [0, 1], \tag{1}$$

where $C \times [0, 1] = \{(c_i, s_i) \mid c_i \in C, 0 \leqslant s_i \leqslant 1\}$, s_i is a numeric value that can be in different forms, such as a similarity score, a class-conditional probability (prior posterior probability) or other measure, depending on the types of learning algorithms used. It represents the degree of support or confidence in the proposition that *instance d is assigned to class* c_i. The greater the value for class c_i, the greater the confidence we have that the instance belongs to that class. Without loss of generality, we denote the classifier output by $\varphi(d) = \{s_1, \ldots, s_{|C|}\}$—a general representation of classifier outputs.

From the representation $\varphi(d)$, alternative forms of outputs can be derived. For example, we could rank all class labels according to their continuous values in descending order. By choosing a single label at the top of the ranked list—the single label with maximal value in the classifier output $\varphi(d)$, we can assign a unique label or a label subset to instance d as a classification decision. Alternatively, we rearrange the process of assigning a unique label to instance d as a mapping of each pair $\langle d, c_i \rangle$ to a Boolean value true (T) or false (F) in terms of the oracle output [29]. If value T is assigned to $\langle d, c_i \rangle$, that means a decision is made that the proposition of instance d belonging to class c_i is true, whereas value F indicates the proposition is false. These alternatives can be seen as the output information at the final stage of classification.

Given an ensemble of classifiers, $\varphi_1, \varphi_2, \ldots, \varphi_M$, if each classifier outputs only a single class label for instance d, the results of classifiers can be combined using majority voting (weighted) [31] or Naive Bayes [49], for example. More generally, if each classifier produces multiple classes as output for instance d—a numeric score vector (list) that is represented as $\varphi_i(d)$ below, all these vectors can then be organized into a matrix called a decision profile (DP) as depicted in Fig. 1 [28]. There are several different ways that the combination of classifier outputs can be carried out.

$$\varphi_i(d) = \{s_{ij} \mid 1 \leqslant j \leqslant |C|\}, \quad 1 \leqslant i \leqslant M. \tag{2}$$

One of the most commonly used combination methods is to calculate the support for class c_j using only the DP's jth column, i.e. $s_{1j}, s_{2j}, \ldots, s_{Mj}$, regardless of what the support for the other classes is. We call such a method a class-aligned method. Some examples of this are linear sum [51], order statistics: *minimum*, *maximum* and *median* [48], and probabilistic *product* and *sum* rules [27]. Alternatively, the combination of classifier outputs can be performed on an entire decision profile, or on the selected information in order to constrain a class decision. We refer to this alternative group of methods as class-indifferent methods.

There exist several related contributions to this subject, including the combination of neural network classifiers using Dempster's rule [41] and the combination of neural network classifiers derived from different feature sets using Dempster's rule [1]. In a broad sense, these studies take a similar approach to calculating the support for classes. That is, for each classifier against each class, a mean vector (called a *reference* vector) is generated and organized into a matrix called a *decision template*, denoted by DT_i. For M classifiers and $|C|$ classes, $|C|$ decision templates with $M \times |C|$ dimensions are formed. Given a decision profile $DP(d)$, the closeness between $DP(d)$ and DT_i, $1 \leqslant i \leqslant |C|$, is computed using different proximity measures such as Euclidean distance and cosine function, the class with the largest support is assigned to instance d.

In our work, the concept of class-indifferent methods is slightly different from the one aforementioned. We neither generate decision templates nor use an entire decision profile to compute the degrees of support for every class. Instead we select 2 and 3 classes from each $\varphi(d)$ according to their numeric values, and restructure these into a new list composed of three and four subsets of classes of C respectively, which are represented by the evidential structures of triplet and quartet. With a triplet, for example, the first subset contains the class with the largest value of confidence, and the second contains the second largest valued class, and the third one is the whole set of classes C. In this way, the decision profile as illustrated in Fig. 1 will be restructured into a triplet or quartet decision profile and each column no longer corresponds to the same class. The degree of support for each class is computed through combining all triplets or quartets in a decision profile. We will detail our method in later sections.

$$|^{0}_{\longleftarrow bel(A)\longrightarrow}|^{bel(A)}_{\longleftarrow ign(A)\longrightarrow}|^{pls(A)}_{\longleftarrow bel(\bar{A})\longrightarrow}|^{1}$$

Fig. 2. Representation of belief interval.

3. Dempster–Shafer (DS) theory of evidence

The DS theory of evidence has been recognized as an effective method for coping with uncertainty or imprecision embedded in evidence used in a reasoning process. It is often viewed as a generalization of Bayesian probability theory, by providing a coherent representation for *ignorance* (lack of evidence) and also by discarding the *insufficient reasoning principle*. The DS theory is well suited to a range of decision making activities. It formulates a reasoning process as pieces of evidence and hypotheses, and subjects these to a strict formal process in order to infer conclusions from the given uncertain evidence, avoiding human subjective intervention to some extent.

In the DS theory, which is also referred to as evidence theory, evidence is represented in terms of *evidential functions* and *ignorance*. These functions include *mass functions, belief functions, and plausibility functions* [44]. Any one of these conveys the same information as any of the others.

Definition 1. Let Θ be a finite nonempty set, called the *frame of discernment*. Let $[0, 1]$ be an interval of numeric values. A mapping function $m : 2^{\Theta} \to [0, 1]$ is called a *mass function* if it satisfies:

$$1)\quad m(\emptyset) = 0, \qquad 2)\quad \sum_{X \subseteq \Theta} m(X) = 1.$$

A mass function is a *basic probability assignment* (bpa) to all subsets X of Θ. A subset A of a frame Θ is called a *focal element* or *focus* of a mass function m over Θ if $m(A) > 0$ and A is called a singleton if it is a one-element subset.

Definition 2. A function $bel : 2^{\Theta} \to [0, 1]$ is called a *belief function* if it satisfies:

$$1)\quad bel(\emptyset) = 0, \qquad 2)\quad bel(\Theta) = 1$$

and for any collection $A_1, A_2, \ldots, A_n$ $(n \geqslant 1)$ of subsets of Θ,

$$bel(A_1 \cup A_2 \cup \cdots \cup A_n) \geqslant \sum_{I \subseteq \{1,2,\ldots,n\}, I \neq \emptyset} (-1)^{|I|+1} bel\Big(\bigcap_{i \in I} A_i\Big).$$

One of the attractive features of the DS theory is that the belief function is in contrast to conventional probability, where the inequality is replaced by an equality. With this function, a plausibility function can be defined as $pls(A) = 1 - bel(\overline{A})$. Notice that a belief function gathers all of the support that a subset A gets from all of the mass functions of its subsets, whereas a plausibility function is the difference between 1 and all of the support of A's complement subsets. The difference $pls(A) - bel(A)$ represents the residual *ignorance*, denoted by *ignorance*(A) (or *ign*(A)). This gives another important feature of DS—the representation of what is precisely known and what remains unknown. Fig. 2 presents an intuitive representation of supporting a subset A by its belief function and plausibility function along with ignorance.

Definition 3. Let m_1 and m_2 be two mass functions on the frame of discernment Θ, and for any subset $A \subseteq \Theta$, the *orthogonal sum* $\oplus$ of two mass functions on A is defined as

$$m(A) = (1/N) \sum_{X,Y \subseteq \Theta, X \cap Y = A} m_1(X) m_2(Y), \tag{3}$$

where $N = 1 - \sum_{X \cap Y = \emptyset} m_1(X) m_2(Y)$ and $K = 1/N$ is called the normalization constant of the orthogonal sum $m_1 \oplus m_2$. The orthogonal sum is a fundamental operation of evidential reasoning and it is often called Dempster's rule of combination (Dempster's rule, for short). There are two conditions governing when it is used to combine mass functions.

The first condition is that $N \neq 0$, i.e. if $N = 0$, the orthogonal sum does not exist.

The second is that two mass functions must be independent of each other; that is they must represent independent opinions or evidence sources relative to the same frame of discernment. Notice that Dempster's rule simply involves two mass functions, it does not tell us whether one mass function is independent of another, nor is it able to rule out the dependence between two mass functions. Therefore, the context of applying Dempster's rule and the effect of combining two mass functions depend very much on how evidence sources are treated or modelled as mass functions, rather than on accounting for their relation alone [24,43].

3.1. Simple support function

In some situations, when the structure of evidence is taken into account mass functions can be significantly simplified. The simplest form of a mass function is called a *simple support function* which is limited to providing a degree of support for a single proposition $A \subseteq \Theta$, which is referred to as a *simplet* structure or *simplet*. This structure provides no support at all for any other propositions discerned by a frame of discernment Θ [44].

Formally, a mass function is said to be *simple* if there is a non-empty subset $A \subseteq \Theta$ such that

$$m(A) = s,$$
$$m(\Theta) = 1 - s,$$
$$m(\Theta - A) = 0,$$

where $0 < s \leqslant 1$, the subset A is a *focus* of m and s is called the *degree of support* and the mass function in this form is called a *simple support function* focused on A.

A simple support function is the belief function *bel* of a simple mass function m. Thus,

$$bel(A) = \sum_{X \subseteq A} m(X) = m(A)$$

for all $A \subseteq \Theta$.

Let $m_1, m_2, \ldots, m_n$ be simple mass functions with the common focus A and respective degrees of support $s_1, s_2, \ldots, s_n$. Then $m_1 \oplus m_2 \oplus \cdots \oplus m_n$ is still a simple mass function with the same focus A, and support degrees for A and others are as follows:

$$(m_1 \oplus m_2 \oplus \cdots \oplus m_n)(A) = 1 - \prod_{i=1,2,\ldots,n} (1 - s_i),$$
$$(m_1 \oplus m_2 \oplus \cdots \oplus m_n)(\Theta) = \prod_{i=1,2,\ldots,n} (1 - s_i),$$
$$(m_1 \oplus m_2 \oplus \cdots \oplus m_n)(\Theta - A) = 0.$$

Separable mass functions include both simple mass functions and orthogonal sums of simple mass functions. They are based on both homogeneous (with the same focuses) and heterogeneous evidence, where different subsets of the frame of discernment can be referenced from different evidence sources.

3.2. Dichotomous function

A mass function m is said to be a dichotomous function if the only possible focal elements of m are A, $\Theta - A$, Θ for some $A \subseteq \Theta$. A special case occurs when A is a singleton $\{x\} \subseteq \Theta$. In such a situation, a dichotomous mass function m has no focuses other than $\{x\}$, $\Theta - \{x\}$, Θ for some x which is referred to as a dichotomous structure [2].

Let $\Theta = \{x_1, x_2, \ldots, x_{|\Theta|}\}$. Suppose that for every $i = 1, 2, \ldots, |\Theta|$, there is a dichotomous mass function m_i: $m_i(\{x_i\}), m_i(\Theta - \{x_i\}), m_i(\Theta)$ and $m_i(\{x_i\}) + m_i(\Theta - \{x_i\}) + m_i(\Theta) = 1$. We view these quantities as follows:

- $m_i(\{x_i\})$ is the degree of support for $\{x_i\}$;
- $m_i(\Theta - \{x_i\})$ is the degree of support for the refutation of $\{x_i\}$; and
- $m_i(\Theta)$ is the degree of the support not assigned for or against the proposition $\{x_i\}$.

Barnett has proposed a technique based on dichotomous mass functions instead of general mass functions. It means that instead of potentially exponential time complexity function in deriving evidence combination, the computation of dichotomous mass functions involves only the 3 particular subsets $\{x\}$, $\Theta - \{x\}$, Θ for each $x \in \Theta$; the general mass functions have to enumerate all $2^{|\Theta|}$ subsets of Θ.

Barnett's approach is to consider the *entire* orthogonal sum for evidence bodies which have the structure $m_1 \oplus m_2 \oplus \cdots \oplus m_{|\Theta|}$. These are precisely those evidence spaces which are separable into exactly $|\Theta|$ dichotomous mass functions $m_1, m_2, \ldots, m_{|\Theta|}$, and the time complexity of combining these mass functions is linear. Guan and Bell [22,23] generalized this to consider the general orthogonal sum as explained below.

Let $\Theta = \{x_1, \ldots, x_{|\Theta|}\}$. Suppose that for each $x_i \in \Theta$, there are l_i dichotomous mass functions of repeated focus: $m_i^j(\{x_i\}), m_i^j(\Theta - \{x_i\}), m_i^j(\Theta)$; $m_i^j(\{x_i\}) + m_i^j(\Theta - \{x_i\}) + m_i^j(\Theta) = 1$; where $i = 1, 2, \ldots, |\Theta|$; $j = 1, 2, \ldots, l_s$; $s = l_1, \ldots, l_k$; $1 \leqslant k \leqslant |\Theta|$; $0 \leqslant l_1, \ldots, l_k$. The task now is to calculate quantities associated with

$$m = \underbrace{m_1^1 \oplus \cdots \oplus m_1^{l_1}}_{l_1 \text{ items}} \cdots \oplus \underbrace{m_k^1 \oplus \cdots \oplus m_k^{l_k}}_{l_k \text{ items}}, \tag{4}$$

where $l_1 + \cdots + l_k = n$, and n is the number of masses to be summed which may be greater than $|\Theta|$. In Eq. (4), the calculation of combining n dichotomous mass functions is divided into two parts. The first part is to combine the mass functions with repeated focal elements. The second part is to combine the mass functions for all focuses. A method of combining these mass functions was studied in [22,23], and it has been shown that the computational complexity of combining such mass functions is linear.

4. Existing methods

In this section we first develop a model to unify the tasks of combining the outputs of multiple classifiers in the conceptual framework of Dempster–Shafer theory and then look at several DS-based studies.

In the supervised learning domain, classifiers can assign one or more classes to each instance. The key assumption underlying our approach here is a single class assignment where each instance belongs to one and only one class. This assumption suggests that only the one-element subsets in 2^C will be of semantic interest and they will be used to represent propositions. By using the DS terminology, given a frame of discernment $C = \{c_1, \ldots, c_{|C|}\}$, evidence derived from classifiers concerns specific individual classes, tending to support singleton classes of C, and the other subsets of 2^C are not particularly meaningful. Therefore the powerset 2^C for all the propositions could be reduced to the subsets that contain only individual classes making up a frame of discernment C itself.

Formally let M be the number of classifiers $\varphi_1, \ldots, \varphi_M$ and let $C = \{c_1, \ldots, c_{|C|}\}$ be a set of classes. For any instance $d \in D$, each classifier produces an output vector $\varphi_i(d)$. Classifying d means assigning it into one class in C, i.e., deciding among a set of $|C|$ hypotheses: d belongs to c_k, $k = 1, \ldots, |C|$, according to $\varphi_i(d)$. In DS terms, C is referred to as a frame of discernment, and the classifying process is regarded as one which decides the true value the proposition of that instance d belongs to c_k according to the knowledge $\varphi(d)$. $\varphi(d)$ can be regarded as a piece of evidence that represents the degrees of our support or belief for the proposition. Instead of 100% certainty, it only expresses some part of our belief committed to $\{c_k\} \in 2^C$ and the rest of our belief which cannot be directly derived from $\varphi(d)$ and the negation of the proposition remains unknown or indiscernible. In the DS formalism, such a situation is regarded as *ignorance*, and belief (mass) functions provide us with an effective way to express it. This is one of the attractive features of DS-based methods.

The merit of assuming the single class assignment is that it significantly eases the application of DS theory to practical problems since the computations of $2^{|C|}$ evidential functions in a general context are reduced to $|C|$. The following sections review four DS-based methods for representing evidence and defining mass (belief) functions based on $\varphi(d)$.

4.1. Xu's method

Xu et al. [49] discussed several approaches for combining multiple classifiers, including majority voting (weighted) and the Bayesian formalism, and proposed a DS model for combining the results of multiple classifiers. Their method treated classifier outputs as single class labels and defined the sources of evidence for the propositions of interest on the basis of the performance of classifiers in terms of *recognition*, *substitution* and *rejection* rates. The items of evidence were represented by dichotomous mass functions.

Let D be a training data set, and suppose that the recognition and substitution rates of classifiers are denoted by $\epsilon_r^i(\varphi_i(D))$ and $\epsilon_s^i(\varphi_i(D))$, and the rejection rates are given by $1 - \epsilon_r^i - \epsilon_s^i$. For a new instance d, a piece of evidence $\varphi_i(d)$ is represented by the following mass function:

$$m_k^i\big(\{c_k\}\big) = \epsilon_r^i\big(\varphi_i(d)\big), \quad 1 \leqslant i \leqslant M, \ \exists k \in \big\{1, \ldots, |C|\big\}, \tag{5}$$

$$m_k^i\big(\{\bar{c}_k\}\big) = \epsilon_s^i\big(\varphi_i(d)\big), \quad 1 \leqslant i \leqslant M, \ \exists k \in \big\{1, \ldots, |C|\big\}, \tag{6}$$

$$m_k^i(C) = 1 - m_k^i\big(\{c_k\}\big) - m_k^i\big(\{\bar{c}_k\}\big), \tag{7}$$

where $\{\bar{c}_k\} = C - \{c_k\}$. With M pieces of evidence existing, represented by M dichotomous mass functions, the degrees of support for classes can be calculated through combining these mass functions by formula (3). A final class decision for a given instance is made on selecting the class with the largest degree of support. Notice that such a combination is an class-indifferent method since the calculation of support for a class c_k is not only based on the mass functions which have the same focus c_k, but the support is also impacted by the mass values of the second subset $\{\bar{c}_k\} = C - \{c_k\}$ and the whole set C, i.e. the *ignorance*.

In the above definition, however, the dichotomous mass functions are not directly defined on the basis of the numeric values of $\varphi_i(d)$ (see formula (2)). Instead they are defined based on the overall performance of classifiers. Therefore for different instances, the basic probability assignments derived from the classifier outputs—dichotomous mass functions—are the same. This method ignores the fact that normally a classifier does not have the same performance on different classes, this might consequently degrade the combined performance of classifiers. Nevertheless the proposed method lays groundwork for formalizing the problem of combining classifier outputs using the dichotomous evidence structure and it achieved a considerable performance improvement when applied to handwriting recognition.

4.2. Rogova's method

Rogova [41] proposed a model for combining the results of neural network classifiers using the DS theory. This work accounted for the relation between classifier outputs and reference vectors as pieces of evidence and developed a general way to measure the relation using proximity measures. The pieces of evidence are represented by simple support functions. Formally, let D_k be a subset of a training data set, in which all instances belong to class $c_k \in C$ and let $\{\varphi_i(x)\}$ $(x \in D_k)$ be a set of classifier outputs. A mean vector of $\{\varphi_i(x)\}$ is denoted by E_k^i called a reference vector for class k. For any instance d, a general proximity measure used for E_k^i and $\varphi_i(d)$ is defined as follows:

$$\mu_k^i = \phi\big(E_k^i,\ \varphi_i(d)\big), \quad i \leqslant M, k \leqslant |C|. \tag{8}$$

The measure ϕ can be in different forms—cosine function, Euclidean distance, and so forth. It provides us with evidence about how likely it is that instance d belongs to class c_k. If the output $\varphi_i(d)$ is far from E_k^i, $\varphi_i(d)$ is considered as providing very little information regarding the proposition of d being under c_k; in that case, ϕ must therefore take on a 'small' value. On the contrary, if $\varphi_i(d)$ is close to E_k^i, one will be much more inclined to believe that d and E_k^i belong to the same class. Simple support functions are given below:

$$m_k^i\big(\{c_k\}\big) = \mu_k^i, \qquad m_k^i(C) = 1 - \mu_k^i, \tag{9}$$

$$m_{j(j\neq k)}^i\big(\{\bar{c}_k\}\big) = 1 - \prod_{r=1, r\neq k} \big(1 - \mu_r^i\big), \qquad m_{j(j\neq k)}^i(C) = 1 - m_j^i\big(\{\bar{c}_k\}\big) = \prod_{r=1, r\neq k} \big(1 - \mu_r^i\big). \tag{10}$$

The interpretation of the above formulation is that formulae (9) and (10) quantify the relation of $\varphi_i(d)$ with each class of reference vector E_k^i—pieces of evidence. However, each piece of evidence represents only some part of our belief which is committed to $\{c_k\}$ and it does not point to any other particular hypotheses. Thus the rest of our belief cannot be distributed to anything else other than C—the whole frame of discernment. Based on this formulation, for a unseen instance, a decision profile is remodeled by formula (11) and the kth $(1 \leqslant k \leqslant |C|)$ column of DP corresponds to M simple support functions with two focal elements, viz. $\{c_k\}$ and the whole set C.

$$DP(d) = \big\{m_j^i \mid 1 \leqslant j \leqslant |C|\big\}, \quad 1 \leqslant i \leqslant M. \tag{11}$$

For such a decision profile, the combination will be performed on each column k using Dempster's rule, resulting in a degree of support for class c_k as follows:

$$m\big(\{c_k\}\big) = \big(m_k^1 \oplus \cdots \oplus m_k^M\big)\big(\{c_k\}\big). \tag{12}$$

The final class $c = \operatorname{argmax}\{m(\{c_k\}) \mid 1 \leqslant k \leqslant |C|\}$ is assigned to instance d. The combination performed in this way using Dempster's rule can be seen as a class-aligned method.

Rogova's work is based on an original idea proposed by Mandler and Schurmann [35] and it extended that work in two aspects regardless of how reference vectors are generated. The first aspect was to introduce a generic form of proximity measure ϕ, allowing different distance measures to be applied to compute class-conditional probabilities—basic probability assignments. The second was to obtain more support for c_k by combining two simple mass functions given by formulae (9) and (10). However, the issue with this extension is the use of the opponent of c_k, i.e., $\bar{c}_k$, which was not explicitly specified. If $\{\bar{c}_k\} = C - \{c_k\}$, then formula (10) seems not to make sense and the combination $m_k^i \oplus m_j^i$ (Rogova used the notation of $m_k \oplus m_{\bar{k}}$) is questionable.

4.3. Al-Ani's method

A similar attempt to apply the DS theory of evidence to combining neural network outputs was carried out by Al-Ani et al. [1]. This method also treated the distance between a reference vector and a classifier output as a piece of evidence. But the difference from the previous work is in the way it obtains reference vectors. It first initializes reference vectors for each class, and then iteratively uses training instances to optimize reference vectors through minimizing the mean square errors between combined classifier outputs and the target outputs, ensuring the optimized reference vectors can be achieved. Finally the distance between the optimized reference vectors and classifier outputs is defined as a piece of evidence and is represented by a simple support function. Let E_k^i be an optimized reference vector. For any instance d, each classifier produces an output vector $\varphi_i(d)$ $(1 \leqslant i \leqslant M)$, a simple support function is defined below:

$$m_k^i\big(\{c_k\}\big) = \frac{\mu_k^i}{\sum_{j=1}^{|C|} \mu_j^i + g^i}, \tag{13}$$

$$m_k^i(C) = \frac{g^i}{\sum_{j=1}^{|C|} \mu_j^i + g^i}, \tag{14}$$

where $\mu_k^i = \exp(-\|E_k^i - \varphi_i(d)\|^2)$ and g^i is a coefficient to be tuned. The combination method is the same as Rogova's, which is static. However, the way of obtaining reference vectors through minimizing the overall mean square error makes

the process of combining classifiers trainable, which may lead to better performance than a static combination scheme, but with the additional cost for training as well as additional training data.

4.4. Denoeux's method

Denoeux [17] proposed an evidence theoretic k-nearest neighbors (kNN) method for classification problems based on the DS theory. Unlike the methods above which were designed for combining classifiers in ensemble learning, this method focuses on a single classifier φ in classifying new instances, by accounting for distances from their neighbors to determine class labels. Formally, let D be a training data set, for instance, $d \in D$ and let Φ be a set of the k-nearest neighbors of d according to some distance measures (e.g., Euclidean distance). Classifying d means assigning it to one of the classes $c_k \in C$ based on the weights of representative classes of its neighbors. Thus the distance between d and neighbor $d_i \in \Phi$ is considered as a piece of evidence to support a proposition about the class membership of d. The evidence is represented by a simple support function as follows:

$$m^i\big(\{c_k\}\big) = \phi(d, d_i), \quad d_i \in \Phi, \tag{15}$$

$$m^i(C) = 1 - \phi(d, d_i), \tag{16}$$

$$m^i(A) = 0, \quad \forall A \in 2^C \setminus \big\{\{c_k\}, C\big\}, \tag{17}$$

where ϕ was suggested to be $\exp(-\gamma(\mu^i)^2)$ and $\mu^i = \|d - d_i\|$. This formulation appears to be similar to Rogova's method without considering what specific distance measures are used. However, Denoeux's method is quite different from Rogova's in the sense that the former computes distances in feature space, whereas the latter works in the decision (classifier output) space. Therefore the decision profile of this method is slightly different from the one given in Fig. 1. For the decision matrix each neighbor is treated as a row and the column corresponds to classes, and the combination using Dempster's rule is carried out on the column. In order to improve the classification accuracy, an effective decision procedure was proposed to determine the optimal or near-optimal parameter values from the data by minimizing an error function [50].

Denoeux's method shows the advantage of permitting a clear distinction between the presence of conflicting information—as happens when an instance is close to several neighbors from different classes—and incomplete information—when an instance is far away from any instances in its neighborhood. It proves to be very competitive with the standard kNN methods. A similar idea has been adapted to a neural network classifier by Denoeux in [16].

5. Triplet mass function

In this section we describe the development of a key evidence structure—the triplet and its formulation.

Starting by analyzing the computational complexity of combining multiple pieces of evidence, we consider how a more efficient method for combining evidence can be established. Given M pieces of evidence represented by formula (2), the computational complexity of combining these pieces of evidence using Eq. (3) is dominated by the number of elements in C and the number of classifiers M. In the worst case, the time complexity of combining M pieces of evidence is $O(|C|^{M-1})$. One way of reducing the computational complexity is to reduce the number of pieces of evidence being combined, so that the combination of evidence is carried by a partition of the frame of discernment C, with fewer focal elements than C, but including possible answers to the question of interest. The partition can thus be used in place of C when the computations of the orthogonal sum are carried out [42]. For example, a dichotomous structure can be used to partition the frame of discernment C into two subsets ϑ_1 and ϑ_2, where there are a number of mass functions that represent evidence in favor of ϑ_1 and against ϑ_2, along with the lack of evidence— *ignorance*. It has been shown that Dempster's rule can be implemented in such a way that the number of computations increases only linearly with the number of elements in C if the mass functions being combined are focused on the subsets where ϑ_1 is singleton and ϑ_2 is the complement of ϑ_1, i.e., $O(|C|)$ [22]. Another approach to reducing the computational complexity of Dempster's rule is to approximate the calculation of belief functions in a coarsened frame of discernment, which is detailed in [15].

The partitioning technique enables a large problem to be broken up into several smaller and more tractable problems. However, a fundamental issue in applying this technique is how to select elements that contain the possibly correct answers to the propositions corresponding to C.

An intuitive way is to select the element with the highest degree of confidence. Indeed, since the classifier outputs approximate class posteriori probabilities, selecting the maximum probability reduces to selecting the output that is the most 'certain' of the decisions. This could be justified from two perspectives. First, the probability assignments given in formula (2) give quantitative representation of judgments made by classifiers on the propositions; the greater their values, the more likely these decisions are correct. Thus selecting the maximum distinguishes the trivial decisions from the important ones. Second, the combination of decisions with the lower degrees of confidence may not contribute to the performance increase of combined classifiers, but only make the combination of classifier's decisions more complicated [48]. The drawback of selecting the maximum, however, is that the combined performance can be reduced by a single dominant classifier that repeatedly provides high confidence values. Contenders with the higher values are always chosen as the final classification decisions, but some of these may not be correct.

To cope with the deficiency resulting from the maximal selection, we propose to take the second maximum decision into account in combining classifiers. Its inclusion not only provides valuable information contained in the discarded class labels by the maximal selection for combining classifiers, but this also to some extent avoids the deterioration of the combined performance caused by the errors resulting from a single dominant classifier that repeatedly produces high confidence values. We propose a novel structure—a triplet—partitioning a list of decisions $\varphi(d)$ into three subsets.

Definition 4. Let C be a frame of discernment, where each choice $c_i \in C$ is a proposition that instance d is classified in category c_i. Let $\varphi(d) = \{s_1, s_2, \ldots, s_{|C|}\}$ be a list of scores, a localized mass function is defined by a mapping function, $m : 2^C \to [0,1]$, i.e. a bpa to $c_i \in C$ for $1 \leqslant i \leqslant |C|$ as follows:

$$m\big(\{c_i\}\big) = \frac{s_i}{\sum_{j=1}^{|C|} s_j}, \tag{18}$$

where $1 \leqslant i \leqslant |C|$.

This mass function expresses the degrees of belief with regard to the choices of classes to which a given instance could belong. With this definition, we rewrite formula (2) as $\varphi(d) = m(\{c_1\}), m(\{c_2\}), \ldots, m(\{c_{|C|}\})$.

Definition 5. Let C be a frame of discernment, and let $\{u\}, \{v\}$ be focal singletons and m be a respective mass function. An expression of the form $Y = \langle \{u\}, \{v\}, C \rangle$ is defined as a *triplet*, it satisfies

$$m\big(\{u\}\big) + m\big(\{v\}\big) + m(C) = 1$$

and mass function m is called a *triplet mass function*.

To obtain triplet mass functions, we define an *outstanding rule* below.

Definition 6. Let C be a frame of discernment and $\varphi(d) = \{m(\{x_1\}), m(\{x_2\}), \ldots, m(\{x_n\})\}$, where $|n| \geqslant 2$, an outstanding rule is a focusing operator, denoted by m^σ, which breaks up $\varphi(d)$ and makes

$$\{u\} = \operatorname{argmax} m\big(\big\{\big(\{x_1\}\big), m\big(\{x_2\}\big), \ldots, m\big(\{x_n\}\big)\big\}\big), \tag{19}$$

$$\{v\} = \operatorname{argmax} m\big(\big\{\{x\} \;\big|\; x \in \{x_1, \ldots, x_n\} - \{u\}\big\}\big), \tag{20}$$

$$m^\sigma\big(\{u\}\big) + m^\sigma\big(\{v\}\big) + m^\sigma(C) = 1. \tag{21}$$

Clearly m^σ is a triplet mass function and it is also referred as a *two-point mass function*. Based on this notation, formula (2) is simply rewritten as formula (22)

$$\varphi_i(d) = \big\{m^\sigma\big(\{u\}\big), m^\sigma\big(\{v\}\big), m^\sigma(C)\big\} \quad 1 \leqslant i \leqslant M. \tag{22}$$

For simplicity, we write $\varphi_i(d) = \{m(\{u\}), m(\{v\}), m(C)\}$.

With the above definition of a triplet, it is easy to illustrate that it meets the two conditions given in Definition 1. We now show that the mass $m(C)$ indeed represents ignorance.

Given a triplet $\langle \{u\}, \{v\}, C \rangle$, and $m(\{u\}) + m(\{v\}) + m(C) = 1$, we let $A = \{u\}$ and $\overline{A} = \{v\}$, then we have

$$bel(A) = \sum_{X \subseteq \{u\}} m(X) = m\big(\{u\}\big),$$

$$pls(A) = \sum_{X \cap \{u\} \neq \emptyset} m(X) = m\big(\{u\}\big) + m(C),$$

$$ignorance(A) = pls(A) - bel(A) = m\big(\{u\}\big) + m(C) - m\big(\{u\}\big) = m(C).$$

From the above formulation it can be seen that u gives the maximum of our quantitative judgments, representing the class with the highest degree of confidence (support) in a list of decisions. It implies that the decision has a strong possibility of being correct. v represents the class with the second highest degree of confidence in the decision list. This decision is less likely to be correct than u. However, its support is important in combining decisions since making a maximal selection may lose valuable information contained in the discarded class labels. Moreover, including v could avoid deterioration of the combined performance caused by a single error of a classifier. Apart from the support for u and v, a certain amount of confidence still remains unassigned, which is assigned to the entire set of classes—it is committed to the frame of discernment C. The triplet evidence structure can be intuitively interpreted as follows. If a classifier cannot successfully assign an instance to the correct class in two occasions, then it is not likely for the classifier to classify this instance correctly at all. We use the ignorance concept to capture important information in such a situation.

To develop formulae for combining two triplet mass functions, we need to consider the relation between two pairs of singletons in any two triplets.

Suppose we are given two triplets $\langle\{x_1\},\{y_1\},C\rangle$ and $\langle\{x_2\},\{y_2\},C\rangle$ where $x_i, y_i \in C$ $(i=1,2)$, and the associated triplet mass functions m_1 and m_2. The enumerative relations between any two pairs of focal points $\{x1\},\{y_1\}$ and $\{x_2\},\{y_2\}$ are illustrated below:

1. Two focal points equal: 1) if $\{x_1\}=\{x_2\}$ and $\{y_1\}=\{y_2\}$, then $\{x_1\}\cap\{y_2\}=\emptyset$ and $\{y_1\}\cap\{x_2\}=\emptyset$; 2) $\{x_1\}=\{y_2\}$ and $\{y_1\}=\{x_2\}$, then $\{x_1\}\cap\{x_2\}=\emptyset$ and $\{y_1\}\cap\{y_2\}=\emptyset$. In this case, the combination of two triplet functions involves three different focal elements.
2. One focal point equal: 1) if $\{x_1\}=\{x_2\}$ and $\{y_1\}\neq\{y_2\}$ then $\{x_1\}\cap\{y_2\}=\emptyset$, $\{y_1\}\cap\{x_2\}=\emptyset$ and $\{y_1\}\cap\{y_2\}=\emptyset$; 2) if $\{x_1\}\neq\{x_2\}$ and $\{y_1\}=\{y_2\}$, then $\{x_1\}\cap\{y_2\}=\emptyset$, $\{x_2\}\cap\{y_1\}=\emptyset$ and $\{x_1\}\cap\{x_2\}=\emptyset$; 3) if $\{x_1\}=\{y_2\}$ and $\{y_1\}\neq\{x_2\}$ then $\{x_1\}\cap\{x_2\}=\emptyset$, $\{y_1\}\cap\{x_2\}=\emptyset$ and $\{y_1\}\cap\{y_2\}=\emptyset$; 4) if $\{y_1\}=\{x_2\}$ and $\{x_1\}\neq\{y_2\}$ then $\{x_1\}\cap\{x_2\}=\emptyset$, $\{x_1\}\cap\{y_2\}=\emptyset$ and $\{y_1\}\cap\{y_2\}=\emptyset$. In this case, the combination of two triplet functions involves four different focal elements.
3. Totally different focal points: if $\{x_1\}\neq\{x_2\}$, $\{y_1\}\neq\{y_2\}$, $\{x_1\}\neq\{y_2\}$ and $\{y_1\}\neq\{x_2\}$, then $\{x_1\}\cap\{x_2\}=\emptyset$, $\{y_1\}\cap\{y_2\}=\emptyset$, $\{x_1\}\cap\{y_2\}=\emptyset$ and $\{y_1\}\cap\{x_2\}=\emptyset$, so the combination involves five different focal elements.

The above three different cases require different formulae for combination. In cases 2) and 3), the combinations of multiple triplet functions cannot be iteratively performed since we are interested only in combining two-point focuses. However, by applying the outstanding rule (see Definition 6), the combined results of any two triplets can be transformed to a triplet mass function. In the following sections, we seek general formulae for combining triplet mass functions based on the three different cases and present our combination algorithm.

5.1. Two focal points equal

Considering the case where two focal singletons $\{x_1\},\{y_1\}$ in one triplet are equal to $\{x_2\},\{y_2\}$ in another triplet, i.e., $x_1=x_2$, $y_1=y_2$ $(x_1\neq y_1)$ and $x_i, y_i \in C$ $(i=1,2)$, we have two triplet mass function m_1 and m_2 along with

$$m_1\big(\{x\}\big)+m_1\big(\{y\}\big)+m_1(C)=1,$$
$$m_2\big(\{x\}\big)+m_2\big(\{y\}\big)+m_2(C)=1.$$

First, we need to show the combination of $m_1\oplus m_2$ does exist and then establish formulae to compute their combination. To ensure the combinability of triplet mass functions, we need only show under what conditions two triplet mass functions m_1 and m_2 are not in conflict, i.e., the normalization factor $K^{-1}\neq 0$. By using formula (3) to combine m_1 and m_2, we can obtain $K^{-1}=1-m_1(\{x\})m_2(\{y\})-m_1(\{y\})m_2(\{x\})$, to make $K^{-1}\neq 0$ to be true, K^{-1} must be greater than zero, i.e., $0<1-m_1(\{x\})m_2(\{y\})-m_1(\{y\})m_2(\{x\})$, thus $m_1(\{x\})m_2(\{y\})+m_1(\{y\})m_2(\{x\})<1$. Under this condition, we establish the formulae for computing the combination of two triplet mass functions below:

$$(m_1\oplus m_2)\big(\{x\}\big)=K\big[m_1\big(\{x\}\big)m_2\big(\{x\}\big)+m_1\big(\{x\}\big)m_2(C)+m_1(C)m_2\big(\{x\}\big)\big], \tag{23}$$
$$(m_1\oplus m_2)\big(\{y\}\big)=K\big[m_1\big(\{y\}\big)m_2\big(\{y\}\big)+m_1\big(\{y\}\big)m_2(C)+m_1(C)m_2\big(\{y\}\big)\big], \tag{24}$$
$$(m_1\oplus m_2)(C)=Km_1(C)m_2(C), \tag{25}$$

where

$$K^{-1}=1-\sum_{X\cap Y=\emptyset} m_1(X)m_2(Y)=1-m_1\big(\{x\}\big)m_2\big(\{y\}\big)-m_1\big(\{y\}\big)m_2\big(\{x\}\big). \tag{26}$$

5.2. One focal point equal

We now consider the case where given two triplet mass functions m_1 and m_2 and two pairs of singletons $\{x\},\{y\}$ and $\{x\},\{z\}$ $(y\neq z$ and $x,y,z\in C)$, a focal element in one triplet is equal to one in another triplet. Following the procedure given in Section 5.1, we can show that m_1, m_2 are combinable if and only if the following condition is held:

$$m_1\big(\{x\}\big)m_2\big(\{z\}\big)+m_1\big(\{y\}\big)m_2\big(\{z\}\big)+m_1\big(\{y\}\big)m_2\big(\{x\}\big)<1.$$

Thus by the orthogonal sum operation, the general formulae for computing the combination of any two triplet mass functions are given below:

$$(m_1\oplus m_2)\big(\{x\}\big)=K\big[m_1\big(\{x\}\big)m_2\big(\{x\}\big)+m_1\big(\{x\}\big)m_2(C)+m_1(C)m_2\big(\{x\}\big)\big], \tag{27}$$
$$(m_1\oplus m_2)\big(\{y\}\big)=Km_1\big(\{y\}\big)m_2(C), \tag{28}$$
$$(m_1\oplus m_2)\big(\{z\}\big)=Km_1(C)m_2\big(\{z\}\big), \tag{29}$$
$$(m_1\oplus m_2)(C)=Km_1(C)m_2(C), \tag{30}$$

where

$$K^{-1} = 1 - m_1(\{x\})m_2(\{z\}) - m_1(\{y\})m_2(\{z\}) - m_1(\{y\})m_2(\{x\}). \tag{31}$$

It is noted that the new mass function $m_1 \oplus m_2$ is no longer a triplet mass function. It now involves four different focal elements $\{x\}, \{y\}, \{z\}$ and C. For more than two triplet functions, the combining process cannot proceed iteratively since we are interested only in two-point focuses. However, by applying the outstanding rule, the combined result can be transformed to a new triplet mass function. We detail the computational steps below.

By Definition 6, we have a new function $(m_1 \oplus m_2)^\sigma$ as follows:

$$(m_1 \oplus m_2)^\sigma(\{x'\}) + (m_1 \oplus m_2)^\sigma(\{y'\}) + (m_1 \oplus m_2)^\sigma(C) = 1.$$

To obtain $(m_1 \oplus m_2)^\sigma$, we assume

$$m_1 \oplus m_2(\{x\}) = f(x); \quad m_1 \oplus m_2(\{y\}) = f(y); \quad m_1 \oplus m_2(\{z\}) = f(x).$$

Then for focal element $\{x'\}$ we have

$$m_1 \oplus m_2(\{x'\}) = f(x'), \tag{32}$$

where $\{x'\} = \operatorname{argmax}(f(x), f(y), f(z))$.

For focal element $\{y'\}$ we have

$$m_1 \oplus m_2(\{y'\}) = f(y'), \tag{33}$$

where $y' = \operatorname{argmax}(f(t) \mid t \in (\{x, y, z\} - \{x'\}))$.

For focal element C we have

$$(m_1 \oplus m_2)^\sigma(C) = 1 - f(x') - f(y'). \tag{34}$$

5.3. Completely different focal points

Finally, let us examine the case where no focal element is common to two triplets. As indicated previously, the combination of two such triplet mass functions will involve five different focal elements. Let m_1, m_2 be two triplet functions, and $\{x\}, \{y\}$ and $\{u\}, \{v\}$ ($x \neq y, x \neq u$ and $y \neq v$, and $x, y, u, v, y \in C$) be two pairs of focal elements along with the following conditions:

$$m_1(\{x\}) + m_1(\{y\}) + m_1(C) = 1,$$
$$m_2(\{u\}) + m_2(\{v\}) + m_2(C) = 1.$$

Following the patten of the previous sections, it can be shown that m_1, m_2 are combinable if and only if the following constraint holds:

$$m_1(\{x\})m_2(\{u\}) + m_1(\{x\})m_2(\{v\}) + m_1(\{y\})m_2(\{u\}) + m_1(\{y\})m_2(\{v\}) < 1.$$

Given the above condition we derive the formulae for computing each focal element below:

$$(m_1 \oplus m_2)(\{x\}) = Km_1(\{y\})m_2(C) = f(x), \tag{35}$$
$$(m_1 \oplus m_2)(\{y\}) = Km_1(\{y\})m_2(C) = f(y), \tag{36}$$
$$(m_1 \oplus m_2)(\{u\}) = Km_1(C)m_2(\{z\}) = f(u), \tag{37}$$
$$(m_1 \oplus m_2)(\{v\}) = Km_1(C)m_2(\{z\}) = f(v), \tag{38}$$

where

$$\begin{aligned} K^{-1} &= 1 - \sum_{X \cap Y = \emptyset} m_1(X)m_2(Y) \\ &= 1 - m_1(\{x\})m_2(\{u\}) - m_1(\{x\})m_2(\{v\}) - m_1(\{y\})m_2(\{u\}) - m_1(\{y\})m_2(\{v\}), \end{aligned} \tag{39}$$

However, the same situation occurs as in Section 5.2, the combination of m_1, m_2 is no longer a triplet mass function, it now involves five focal elements $\{x\}, \{y\}, \{u\}, \{v\}, C$, therefore further combinations with more triplet functions are invalid in this context. To obtain a new triplet mass function, there is a need to apply the outstanding rule to the combined results.

More specifically, by Definition 6, we can obtain a new function $(m_1 \oplus m_2)^\sigma$ as follows:

$$(m_1 \oplus m_2)^\sigma(\{x'\}) + (m_1 \oplus m_2)^\sigma(\{y'\}) + (m_1 \oplus m_2)^\sigma(C) = 1.$$

Then for focal element $\{x'\}$ we have

$$m_1 \oplus m_2(\{x'\}) = f(x'), \tag{40}$$

where $\{x'\} = \operatorname{argmax}(f(x), f(y), f(u), f(v))$.

```
1  set T a set of triplet mass functions
2  ct: holds the combined results of triplet mass functions
3  set ct ← t' ∈ T
4  for each t ∈ T \ {t'} do
5    if (two focuses equal in t and ct) then {
6        ct ← ct ⊕ t          //combine them by formulae (23)–(26)
7    endif
8    else if (one focus equal in t and ct) then
9        ct ← ct ⊕ t          //combine them by formulae (27)–(31)
10       ct ← ct^σ            //transform it a new triplet by formulae (32)–(34)
11   endelseif
12   else
13       ct ← ct ⊕ t          //combine them by formulae (35)–(39)
14       ct ← ct^σ            //transform the combined results to a new triplet by formulae (40)–(42)
15   endelse
16 endfor
17 return ct
```

Algorithm 1. Combine multiple triplet mass functions.

For focal element $\{y'\}$ we have

$$m_1 \oplus m_2\big(\{y'\}\big) = f(y'), \tag{41}$$

where $y' = \text{argmax}(\{f(t) \mid t \in (\{x, y, u, v\} - \{x'\})\})$.

Finally, for focal element C we have

$$(m_1 \oplus m_2)^{\sigma}(C) = 1 - f(x') - f(y'). \tag{42}$$

We have shown that any two triplet mass functions are combinable if the conditions hold and we have established the formulae for computing the combination of two triplet mass functions. These formulae provide a way to not only compute the combinations of two triplet mass functions efficiently but also help us develop a general algorithm for combining multiple triplet mass functions in a complex situation where the evidence sources of triplets are independent of each other.

Suppose we are given M triplet mass functions $m_1, m_2, \ldots, m_M$, by using the algorithm below they can be combined in any order due to Dempster's rule being both commutative and associative. That means we can arrange these triplet mass functions into a certain order based on three cases mentioned above. By repeatedly applying the outstanding rule at each computational step of combining two triplet mass functions, the results can be transformed to a new triplet mass function. Formula (43) is a pairwise orthogonal sum calculation for combining any number of triplets which can be performed by the above algorithm. Its time complexity is approximately $O(2 \times |C|)$ and the final decision is a class with the largest degree of support among all the classes.

$$m = m_1 \oplus m_2 \oplus \cdots \oplus m_M = \big[\cdots\big[\,[m_1 \oplus m_2] \oplus \cdots \oplus m_M\big]\big]. \tag{43}$$

5.4. Focusing on more points—quartet

In the previous sections we have presented the formulation of triplet functions and developed formulae for combining them. Similarly, we can consider three-point focuses, four-point focuses, or more focuses. In general, when a mass function has more than four focal singletons, we can use the outstanding rule σ to focus on 3 points in terms of *quartet*. Let m be a mass function with focal singletons $\{x_1\}, \{x_2\}, \ldots, \{x_n\}, n \geqslant 4$, and $n \leqslant |C|$. Then the focusing operator σ is used to obtain m as follows:

$$m^{\sigma}\big(\{u\}\big) + m^{\sigma}\big(\{v\}\big) + m^{\sigma}\big(\{z\}\big) + m^{\sigma}(C) = 1,$$

where

$$\{u\} = \text{argmax}\, m\Big(\big\{\big(\{x_1\}\big), m\big(\{x_2\}\big), \ldots, m\big(\{x_n\}\big)\big\}\Big), \tag{44}$$

$$\{v\} = \text{argmax}\, m\Big(\big\{\{x\} \mid x \in \{x_1, \ldots, x_n\} - \{u\}\big\}\Big), \tag{45}$$

$$\{z\} = \text{argmax}\, m\Big(\big\{\{x\} \mid x \in \{x_1, \ldots, x_n\} - \{u, v\}\big\}\Big), \tag{46}$$

$$m^{\sigma}(C) = 1 - m^{\sigma}\big(\{u\}\big) - m^{\sigma}\big(\{v\}\big) - m^{\sigma}\big(\{z\}\big) \tag{47}$$

and $\{u\}, \{v\}, \{z\}$ are three-point focuses. The structure of the quartet is conceptually simple, but it has the added advantage that it can be used to handle the case where the classifier results are diverse. For this case, the theoretical analysis is clearly more complicated than that of the triplet structure. In addition to the properties of mass function and ignorance, the corresponding analysis also needs to address, in each case, the situation where the focuses are ordered by their masses for each of the two pieces of evidence being combined. For example, we need to consider the cases where all three focuses are the same, two focuses are identical, where just one focus is shared, and where there are no focuses in common. More details about the extension from triplet to quartet can be found in [6].

Table 1
General description of the datasets

Dataset	Instance	Class	Attribute	
			Categorical	Numerical
anneal	798	6	32	6
audiology	200	23	69	0
balance	625	3	4	0
car	1728	4	6	0
glass	214	7	0	9
autos	205	6	10	15
iris	150	3	0	4
letter	20 000	26	0	16
heart	303	5	8	5
segment	1500	7	0	19
soybean	683	19	35	0
wine	178	3	0	13
Zoo	101	7	15	2

Table 2
General description of the thirteen learning algorithms

No	Classifier	Description
0	AOD	Perform classification by averaging over all of a small space of alternative naive-Bayes-like models that have weaker (and hence less detrimental) independence assumptions than naive Bayes
1	NaiveBayes	The Naive Bayes classifier using kernel density estimation over multiple values for continuous attributes, instead of assuming a simple normal distribution
2	SMO	Sequential minimal optimization algorithm for training a support vector classifier using polynomial or RBF kernels
3	IBk	A instance-based learning algorithm. It uses a simple distance measure to find the training instance closest to the given test instance, and predicts the same class as this training instance
4	IB1	The IBk instance-based learner with $K = 1$ nearest neighbors, in order to offset KStar with a maximally local learner
5	KStar	The K instance-based learner using all nearest neighbors and an entropy-based distance
6	DecisionStump	Building and using a decision stump, but it is not used in conjunction with a boosting algorithm
7	J48	Decision tree induction, a Java implementation of C4.5
8	RandomForest	Constructing random forests for classification
9	DecisionTable	A decision table learner
10	JRip	A propositional rule learner—a Java implementation of Ripper. It repeats incremental pruning to produce error reduction
11	NNge	Nearest neighbor-like algorithm using non-nested generalized exemplars
12	PART	Generating a PART decision list for classification

6. Experimental evaluation

6.1. Experimental settings

In our experiments, we used thirteen data sets downloaded from the UCI machine learning repository [3]. All the selected data sets have at least three or more classes as required by the evidential structures. The details about these data sets can be found in Table 1.

For generating individual (base) classifiers, we used thirteen learning algorithms which are taken from the Waikato Environment for Knowledge Analysis (Weka) version 3.4 (see Table 2). These algorithms were simply chosen on the basis of the performance over three data sets which were randomly picked. They can make up various ensembles of classifiers. Parameters used for each algorithm were set at the default settings. Detailed description of these algorithms can be found in [55].

To reflect the ensemble performance faithfully and to avoid overfitting to some extent, the experiments were performed on a three partition scheme using a ten-fold cross validation. We therefore divided each of the data sets into 10 mutually exclusive sets. For each fold, after excluding the test set, the training set was further subdivided into 70% for a new training set and 30% for a validation set. Apart from evaluating the performance of individual classifiers, the validation set was used to select the best combination of classifiers. The performance of combining selected classifiers using the DS and MV (majority voting) combining schemes was evaluated on the testing set.

Seven groups of experiments are reported here, which were carried out individually and in combination across all the thirteen data sets. These are:

- evaluating the performance of the 13 algorithms shown in Table 2,
- experimenting with various combinations of individual classifiers using DS, in which the classifier outputs are represented by triplet and quartet functions, respectively,

Instances	φ_1	$\oplus$	φ_2	$\longrightarrow$ combined φ
x_1	triplet_{11}	$\oplus$	triplet_{21}	$= \text{triplet}_1$
x_2	triplet_{12}	$\oplus$	triplet_{22}	$= \text{triplet}_2$
...	...		...	...
x_{T1}	triplet_{1T1}	$\oplus$	triplet_{2T1}	$= \text{triplet}_{T1}$
				?/T1

Fig. 3. An example: the combination of two individual classifiers φ_1 and φ_2 with the triplet structure in a fold of a ten-fold cross validation, T1 is a number of instances within a fold of training data.

- examining the performance of combining 13 individual classifiers in the three different orders of decreasing, corresponding increasing and increasing using DS,
- experimenting with combinations of individual classifiers represented by the dichotomous structure using DS, where the dichotomous mass functions were defined on the basis of the performance of classifiers in terms of recognition, substitution and rejection rates [49],
- conducting experiments on combining the individual classifiers represented by the simplet structure using DS. The simplet mass functions were defined based on Rogova's method (see Section 4.2; here classifiers were generated by the learning algorithms shown in Table 2 rather than neural networks),
- experimenting with combinations of classifiers using DS, where classifier outputs are represented by the full list of decisions (contenders),
- experimenting with combinations of individual classifiers using MV, in which the individual outputs are single class labels.

It is noted that the ensemble construction involves 2^{13} combinations of 13 individual classifiers for one evidential structure with one data set in one fold. The computational cost for all the structures using a ten-fold cross validation requires $2^{13} \times 13 \times 5 \times 10$ combinations in total. Instead of exhausting all the combinations of classifiers, we ranked all the individual classifiers based on their performance and then combined them in decreasing order as suggested in [1,8]. For example, we took the best individual classifier, and then combined it with the second best, the third best, and so forth, until we achieved the best accuracy of the combined classifiers. During the course of combination, a hybrid order (and non consecutive ordering) was also involved in combining classifiers in order to find out the best combination of classifiers. Fig. 3 presents an example of combining two individual classifiers where the classifier outputs are represented by triplets.

To compare the classification accuracies between the individual classifiers and the ensemble classifiers across all the data sets, we employed ranking statistics in terms of the *win*/ *draw*/ *loss* (W/D/L) record used by Webb [53]. The win/draw/loss record presents three values, the number of data sets for which classifier A obtained better, equal, or worse than classifier B with respect to classification accuracy. All collected classification accuracies were measured by the averaged F-measure [55]. A paired t-test across all these domains was also carried out to determine whether the differences between the base classifiers and combined classifiers (ensembles of classifiers) are statistically significant at the 0.05 level.

6.2. The basics of Kappa (κ) statistics

To examine the reliability of the ensemble performance, we performed a κ (Kappa) statistic analysis on the extent (level) of agreement between the combined classifiers under the different evidential structures using DS, and also the extent of the agreement between majority voting and Dempster's rule in combining the base classifiers.

The κ statistic is the most widely used pairwise method to measure the level of agreement (or disagreement) between raters/classifiers [29]. It can be thought of as chance-corrected proportional agreement [20]. Formally, given two classifiers φ_1 and φ_2 and a testing data set T, we can construct a global contingency table where entry E_{ij} contains the number of instances $x \in T$ for which $\varphi_1 = i$ and $\varphi_2 = j$. If φ_1 and φ_2 are identical on the data set, then all non-zero counts will appear along the diagonal of the table, otherwise there will be a number of counts off the diagonal. Now we define

$$\mu_1 = \frac{\sum_{i=1}^{L} E_{ii}}{|T|},$$

$$\mu_2 = \sum_{i=1}^{L} \left(\sum_{j=1}^{L} \frac{E_{ij}}{|T|} \times \sum_{j=1}^{L} \frac{E_{ji}}{|T|} \right),$$

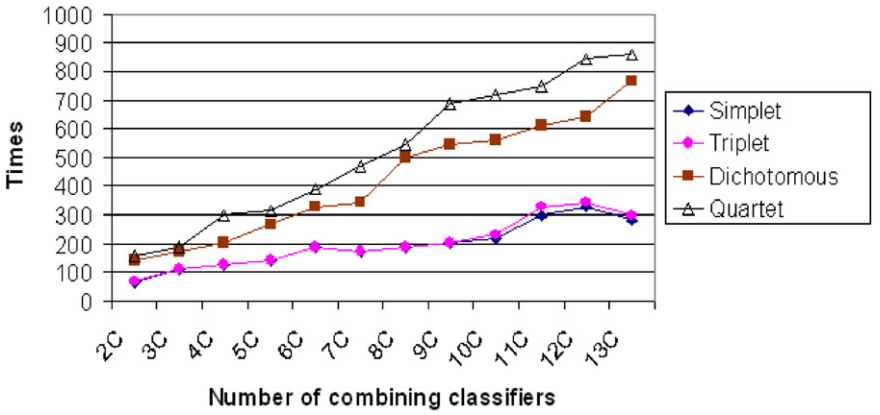

Fig. 4. Comparison of running time among simplet, triplet quartet and dichotomous structures.

where μ_1 is the probability that two classifiers agree and μ_2 is a correction term for μ_1, estimating the probability that the two classifiers agree simply by chance. Then the κ statistic is defined as follows:

$$\kappa = \frac{\mu_1 - \mu_2}{1 - \mu_2}$$

$\kappa = 0$ when the agreement of two classifiers equals that expected by chance, $\kappa = 1$ when two classifiers agree on all the testing instances, and negative values of κ mean that an agreement is less than expected by chance.

6.3. Special cases for triplets and quartets

In our experiments, to ensure that Dempster's rule is properly applied to combine triplet mass functions (and quartet mass functions), i.e., making sure $N \neq 0$ as required in Definition 3, adjustments have been made for the following three special cases.

The first one is that given two mass functions m_1 and m_2 along with two pairs of singletons $\{x_1\}, \{y_1\}$ and $\{x_2\}, \{y_2\}$, and the whole set C, if $x_1 \neq x_2$, $y_1 \neq y_2$, $x_i \neq y_j$ $(i, j = 1, 2)$, and $m_1(\{x_1\}) = 1$ or $m_1(\{y_1\}) = 1$ and $m_2(\{x_2\}) = 1$ or $m_2(\{y_2\}) = 1$, then the intersection of m_1 and m_2 is committed to the empty set $\emptyset$, and consequently the condition of $N \neq 0$ does not hold. Thus it is necessary to redistribute masses over the triplets where the uncertainties lie. In this situation, we discount the masses of x_i and y_j $(i, j = 1, 2)$ to $m_i(C)$ $(i = 1, 2)$ by a small value α which has been defaulted as 0.001 in our experiments.

The addition of α represents the uncertainty involved in a classification process, which can be justified in two ways: 1) generating classifiers is an approximation process, so that the class conditional probabilities estimated by classifiers is not of 100% certainty; 2) α will play a role in improving the combined performance (or at least, in not deteriorating the combined effect).

The second case is where, as usual, m is a mass function derived from a classifier output, $\{x\}$ and $\{y\}$ are a pair of singletons and C is the whole set of classes. If the classifier output for an instance is a nil one, this means the classifier is not able to assign a class label to the instance, thus $m(\{x\}) = m(\{y\}) = 0$. For such a situation, we reallocate 1 to $m(C)$. Making $m(C) = 1$ can be regarded as the representation of uncertainty in classifying the instance by the classifier; in fact, it only ensures $N \neq 0$ when m is combined with another mass function m' and it does not affect the value of $m \oplus m'$.

The last case is where we are given a resulting triplet mass function m together with singletons $\{x\}, \{y\}, \{z\}$, and $m(\{x\}) = m(\{y\}) = m(\{z\})$. To approximate m as a new triplet mass function or determine the best focus, we have no criterion for identifying the best choice. All we can do is pick two focuses up at random for constructing a triplet, or randomly take one as a final decision.

Similar treatments have also been administered for the special cases of quartet mass functions.

6.4. Experimental results

6.4.1. Run time

The first experimental result is the comparison of running time required in combining different evidential structures using DS. As expected, combining simplets was more efficient than the others as there is less computation involved. The empirical results show the time required for combining triplet functions is 3.5% longer than that of combining simplets. The time required for combining dichotomous functions is 119.6% longer than that of simplets on average, and the time for combining quartets is 169.1% longer than that for simplets on average. The comparative results are illustrated in Fig. 4.

From the above results, it can be seen that although the triplet and dichotomous structures both have three focal elements, the computational time required for combining triplet functions is significantly less than that required for combining

Table 3
The classification accuracy of the best base classifier, the best combined classifiers based on the structures of simplet, triplet, quartet, dichotomous structure and fullist using DS and MV over 13 data sets

Datasets	Individual	Simplet	Triplet	Quartet	Dichotomous	MV	Fullist
Anneal	80.23	**81.57**	**81.57**	79.77	80.68	**81.14**	69.88
Audiology	48.67	**53.67**	**57.44**	**57.44**	51.97	**54.30**	49.32
Balance	65.67	63.17	63.17	64.44	65.67	62.72	21.68
Car	89.62	**95.40**	**94.29**	**96.96**	**91.92**	**91.75**	90.03
Glass	65.36	64.91	66.81	64.91	66.26	66.69	62.75
Autos	77.59	78.46	79.17	78.08	78.78	77.94	66.80
Iris	95.33	**96.67**	**96.67**	**96.67**	**96.67**	**96.67**	60.71
Letter	92.05	92.41	**92.91**	92.54	**93.41**	92.77	68.38
Heart	35.48	36.02	37.09	37.03	35.37	34.37	34.26
Segment	96.69	97.48	97.28	96.68	**97.74**	96.55	88.40
Soybean	95.89	**96.92**	96.69	96.88	96.85	96.17	91.11
Wine	98.90	**100.00**	**100.00**	**100.00**	98.90	98.97	96.70
Zoo	90.62	**93.61**	**93.61**	**93.61**	**93.61**	**93.61**	64.39
Average	79.39	80.79	81.30	81.15	80.60	80.28	66.49
W/D/L		11/0/2	12/0/1	10/0/3	10/2/1	10/0/3	–
Sig win		7	7	5	5	4	–

Table 4
The extent of agreement among the best combined classifiers under the four structures along with the agreement between combination methods DS and MV

	Simplet	Quartet	Dichotomous	Triplet	Average
Triplet (DS)	0.9325	0.9396	0.9349	–	0.9357
MV	0.9427	0.9273	0.9432	0.9437	0.9392

dichotomous functions. This is mainly due to the second element of the dichotomous structure, i.e., it is a subset of the whole set C which contains only one class less than the whole set of classes C. Examining the calculation process of Dempster's rule, it is not difficult to find that for the optimized case, combining two dichotomous mass functions requires $(|C|-2)^2$ more computations than combining two triplet mass functions using DS.

6.4.2. Performance summary

The seven groups of experimental results are summarized in Table 3. The first column lists all the data sets used, the second column gives the classification accuracies of the best individual classifiers and the rest of the columns represent the accuracies of the best combined classifiers using DS or MV over the data sets. If the difference between the best combined classifier and the best individual or base classifier on the same data set is statistically significant, then the larger of the two is shown in *bold*.

The bottom of the table provides summary statistics comparing the performance of the best base classifiers with the best combined classifiers across the data sets. From this summary, it can be observed that the accuracy of the combined classifiers based on the *triplet* structure using DS is better than any of the five others on average. It has more wins than loses over the simplet, quartet, dichotomous structure and the best combined classifiers using MV compared to the best individual classifiers. This conspicuous superiority is further supported by the number of statistically significant wins—the triplet has two more than the quartet and dichotomous structure, and three more than MV.

6.4.3. κ statistics

For examining the reliability of the ensemble performance, we selected six data sets (Anneal, Audio, Car, Iris, Wine, Zoo) where the ensemble performance is statistically significant and better than that of the best individuals, and carried out a pairwise analysis on the level of agreement between the best combined classifiers along with the level of agreement between DS and MV on the testing data. Table 4 shows the statistical results averaged on the six data sets. Within the table, the first row consists of the four best combined classifiers, denoted by Simplet, Quartet, Dichotomous structure and Triplet, each of which corresponds to the six data sets. The first column names the two best combined classifiers with DS and MV, simply denoted by Triplet (DS) and MV, respectively. They are associated with the six data sets as well. Each remaining cell of the table is an averaged κ value, which represents the level of agreement between a pair of the classifiers on classifying the instances of the six data sets. For example, in the cell of (Triplet (DS), Simplet), 0.9325 is the averaged κ value of the best combined classifiers of Triplet and Simplet which agree on classifying the instances across the six data sets. The κ values presented in Table 4 indicate that the agreement on classifying the instances by the pairs of classifiers is *substantial* according to the rough guide provided in [20]. This establishes that the performance of these combined classifiers is reliable.

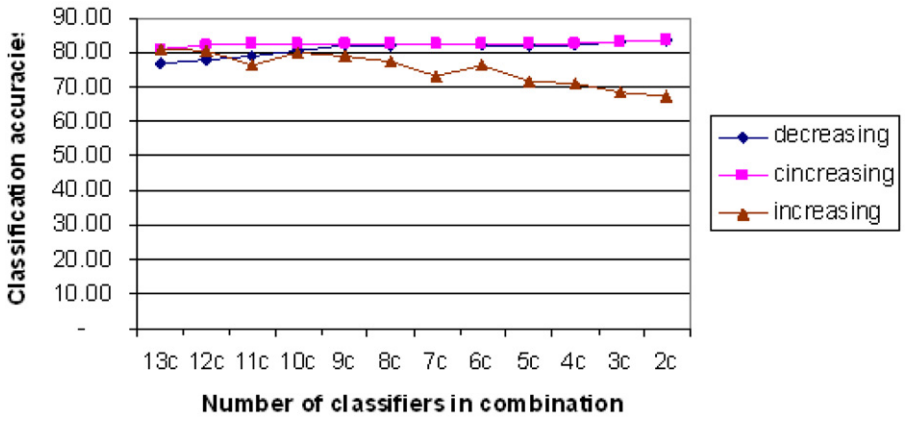

Fig. 5. Performance of different numbers of classifiers in combination with different orders: decreasing, corresponding increasing and increasing.

6.4.4. Order of combining classifiers under the triplet

As explained in Sections 5.2 and 5.3, combining two triplet mass functions does not in general result in a triplet mass function. The combined results may need to be approximated to a new triplet using the focusing operator. However such an approximation may make the combination of triplets no longer meet the associative property of Dempster's rule. Thus different orders of combination of classifiers may lead to the different combined performance.

To better understand how the chosen order of classifiers affects the combined performance in ensemble construction, we have conducted a further experiment on combining classifiers with three different orders of decreasing, corresponding increasing (*c-increasing*) and increasing. The combination process is as follows. We ranked the 13 best individual classifiers in decreasing and increasing orders, and then consecutively combined them one by one using Dempster's rule. With respect to the c-increasing order, it is a reverse process of *each of* the corresponding decreasing combinations. For instance, the first decreasing combination is to combine the best classifier with the second best, the second is to combine the best classifier with the second best and then with the third best; this process will be repeated until all decreasing combinations are completed. The reverse process of the decreasing combination is, correspondingly, to combine the second best classifier with the first, to combine the third best classifier with the second and then with the best, and so forth.

Fig. 5 illustrates an effect comparison of three orders in combining classifiers under the triplet. It can be observed that the combined classifiers with the c-increasing order outperforms the other two, and the performance of the increasing combination is worst among the three orders of combination. This suggests that combining the better classifiers would outperform combining the worse classifiers. Consider the c-increasing order. It should be noted that although its performance is better that of the decreasing combination, this mainly occurs on the combination of 9 classifiers and more; for the combination of 3 to 8 classifiers, the decreasing performance tends to be the same as that of the c-increasing. This experimental result leads to the conclusion that the order of the classifiers in combination has an impact on the performance of the combined classifiers, but it is not dramatic. For both the decreasing and the c-increasing orders, the smaller the number of classifiers to be combined, the less the impact of the combination order.

6.4.5. Ensemble size

Fig. 6 presents the sizes of the best combined classifiers across all the data sets. With the different structures, the construction of these ensembles involve 2–7 classifiers, and most of the ensembles are composed of only two classifiers. This result is consistent with some previous studies [1,8,19], but different from others [36,38] where experiments showed that ensemble accuracy increased with ensemble size and the performance levels out with ensemble sizes of 10–25. The sizes of ensembles of classifiers which are generated using different learning algorithms to operate on a single data set are not necessarily the same as those of the combined classifiers constructed using a single learning method to manipulate different portions of features or instances.

6.4.6. Contribution to the ensembles

Fig. 7 presents the contribution to the ensemble construction of each individual classifier with the structures of simplet triplet, quartet and dichotomous structure across all the domains. There are eleven aggregated columns with four different colors in the figure. Each column represents the number of a classifier which contributes to the construction of the thirteen best combined classifiers (corresponding to thirteen data sets), and each color presents an evidential structure. For example, the first column indicates that AOD occurs six times over the thirteen ensembles (thirteen data sets) with the simplet and triplet structures, and seven times over the thirteen ensembles with the quartet and dichotomous structure. It is observed that the classifiers that contribute to four or more ensembles with four structures are AOD, SMO, IBk and KStar. Of these, AOD is not the best performing individually, but it plays a more important role than the others in constructing the effective ensembles.

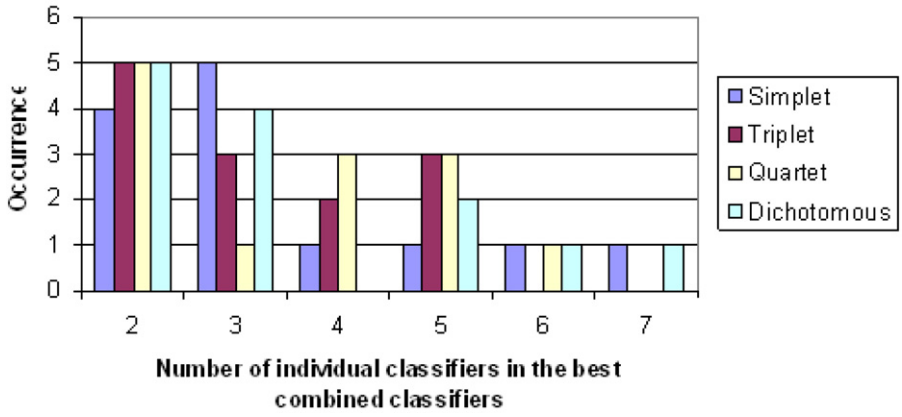

Fig. 6. Ensemble size: number of individual classifiers involved in the best combined classifiers across all the data sets with the evidential structures of simplet, triplet, quartet and dichotomous structure.

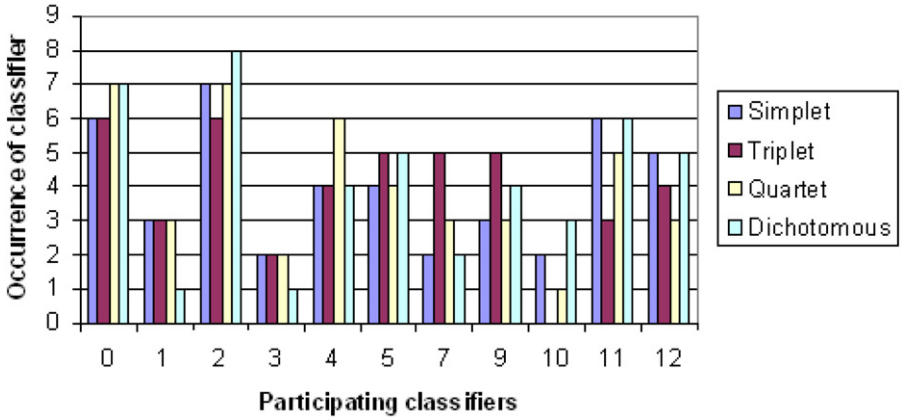

Fig. 7. Number of the individual classifiers (0: AOD; 1: NaiveBayes; etc. see Table 2) occurring in all thirteen combined classifiers across all the thirteen data sets with different structures of simplet, triplet, quartet and dichotomous structure.

7. Discussion

The structure of simplet consists of two elements. The first is a decision in a list of decisions and the simplet mass represents the degree of support for that decision, which is derived from a decision profile using a cosine function. The second is the whole list of decisions C, its mass representing the uncertainty of the singleton element. With this structure, a list of decisions is associated with a list of simplet mass functions where each of the simplet functions corresponds to one of the decisions, and both the lists share the same order after ranking. It is easy to illustrate that the combined effect of two simplet functions will be highly affected by the larger of the simplet masses.

Consider the combination of two simplet mass functions m_1 and m_2 along with two singletons $\{x\}$ and $\{y\}$ in the case where x and y are not equal. By using formula (3) to combine m_1 and m_2, x is the better supported decision if $m_1(\{x\}) > m_2(\{y\})$, otherwise y will be better, and $m_1(\Theta)$ and $m_2(\Theta)$ do not affect the combination effect. Thus, we find that which of $\{x\}$ and $\{y\}$ becomes the better supported decision depends merely on the mass values of $\{x\}$ and $\{y\}$. This effect can be generalized to combining two simplet classifiers $\varphi_1 \oplus \varphi_2$; that is, the performance of $\varphi_1 \oplus \varphi_2$ is dominated by a single simplet classifier φ_1 or φ_2 which repeatedly provides larger mass functions m_1 or m_2.

Although the way of deriving mass values for simplets is different from that for triplets and quartets, in the broadest sense, the triplet and quartet can be regarded as extensions of the simplet structure. The key difference between both the triplet and the quartet and the simplet is that the former makes use of a wide range of information that is contained in the second and third best decisions in combining classifiers. The performance of combining two triplet classifiers, for example, will be determined by the first and second elements along with ignorance. We conjecture that the use of more information plays an important role in overcoming the problem we identified earlier—that a single error produced by a single classifier which repeatedly provides high confidence values of classes can occur when combining simplets.

Now we look at a theoretical justification of our claim. We state formally the conditions for the first or second decision in either of two triplets to become the better supported decision. Assume that two triplet functions m_1 and m_2 fall into the

category where a pair of singletons $\{x_1\}, \{y_1\}$ is equal to a pair of $\{x_2\}, \{y_2\}$, i.e., $x_1 = x_2$ and $y_1 = y_2$ (see Section 5.1). By using formulae (23)–(26), we have the following inequality when x is the better choice:

$$m_1(\{x\})m_2(\{x\}) + m_1(\{x\})m_2(C) + m_1(C)m_2(\{x\}) > m_1(\{y\})m_2(\{y\}) + m_1(\{y\})m_2(C) + m_1(C)m_2(\{y\}). \tag{48}$$

Substituting for C in formula (48) and rearranging it, we have the condition for the better support for x:

$$m_1(\{x\}) > 1 - \frac{[1 - m_1(\{y\})][1 - m_2(\{y\})]}{[1 - m_2(\{x\})]}. \tag{49}$$

Likewise we can derive the condition for y being the better supported decision:

$$m_2(\{y\}) > 1 - \frac{[1 - m_1(\{x\})][1 - m_2(\{x\})]}{[1 - m_1(\{y\})]}. \tag{50}$$

We can obtain the conditions for the other two cases in two triplets in a similar manner.

Formulae (49) and (50) present two conditions for determining which of x and y is the better choice. The first condition indicates that, for example, when x is in the second position of a list of decisions, it can be ranked as the first decision when two triplets are combined as long as this condition is met. This effect provides an insight into the process of combining triplet classifiers where a single classifier cannot dominate the combined performance and the ignorance derived also plays an important role in deciding the best supported decisions. This is one explanation of the superior performance of the triplet over the simplet. A similar analysis for the case of combining quartets can be carried out in the same way, obtaining a possible account of when the quartet outperforms the simplet.

With respect to the dichotomous structure, its drawback is in its way of measuring evidence, where it ignores the fact that classifiers normally do not have the same performance on different classes, which could cause a deterioration in the performance of the combined classifiers (see Section 4.1).

It is not a surprise that the performance of combining classifiers in the form of triplets and quartets is better than that of combining classifiers with a full list of decisions. The reason for this is that different individual classifiers produce various distributions of class-conditional probabilities for all the classes. The classes can have various predicted values in the range between zero and one. When two lists are combined using the orthogonal sum operation, if φ_1 and φ_2 produce two lists of decisions with a similar distribution, then many non-zero values will appear along the diagonal of an intersection table (for the orthogonal sum calculation). Otherwise there will be a large number of values off the diagonal. In the latter situation, the combination of the two lists is committed to more conflicting class labels. Examining the calculation of orthogonal sums closely, we find that the conflict incurred in combining two classifiers in the form of a full list is larger than that in combining two triplet classifiers. Such conflict could thus result in poor combined performance of classifiers, using a full list. This finding is also consistent with the result illustrated in Fig. 5 and the previous studies where the combination of decisions with the lower degrees of confidence may not contribute to an increase of combined performance of classifiers, but only make the combination of decisions more complicated [48].

8. Independence assumption

Making an independence assumption for a set of variables or a set of attribute values is a common practice in many learning tasks, such as Bayesian belief networks and naive Bayes classifier [37]. Such an assumption dramatically reduces the complexity of learning classifiers and makes the computational process more tractable. This is also true in other applications, for instance, in the present context of modelling classifier outputs as independent bodies of evidence. The independence assumption is of practical value, but there is little information available about whether it has substantially deteriorated the effectiveness of classifiers in many applications.

As mentioned in Definition 3, one condition of using Dempster's rule is that pieces of evidence to be combined are independent. However the precise meaning of independence in practice is difficult to specify. In an effort to clarify this, Dempster [13] explained that "opinions of different people based on overlapping experiences could not be regarded as independent sources". This was subsequently complemented by a statement by Denoeux [14], who said that, "non-overlapping random samples from a population are clearly distinct items of evidence". In the present context, a possible argument could be used against independence of two pieces of evidence derived from two classifier outputs due to the fact that the two classifier outputs arise from the same sample instance. However, at the same time it can also be argued that two classifier outputs are distinct, because it is not clear that two classifiers have 'overlapping experiences' in determining class labels for an instance.

Somewhat less philosophically, the argument in favor of the independence assumption is that the classifiers involved in combination are generated by the distinct learning algorithms as shown in Table 2. These algorithms are built on different theories and they use their own mechanisms to search for a characterization of the data. So they do not share "experiences" in generating classifiers, nor is there correlative dependence between the internal structures of the classifiers. The inherent processes of producing outputs by classifiers are entirely distinct. This distinctness can be formally interpreted as follows. Let φ_1 and φ_2 be two distinct classifiers, for any instance d, $\varphi_1(d)$ does not logically imply $\varphi_2(d)$, and vice versa, hence they are mutually unrelated. Consequently as a natural case, denoting e_1 as the proposition corresponding to $\varphi_1(d)$ and e_2

to $\varphi_2(d)$ (see Section 4), then e_1 is independent of e_2 and the probability $P(e_1 \wedge e_2) = P(e_1) \cdot P(e_2)$ [52]. Therefore the assumption made for modelling classifier outputs as independent pieces of evidence is sensible.

There is a great deal of debate about the conditions for validly applying Dempster's rule and the precise meaning of distinct bodies of evidence in the literature [13,14,33,43,52], and so far there is no conclusive study to these issues. Shafer [43] pointed out that the task of sorting evidence into independent items is not always easy, and "a theory that directs us to this task is grappling with the real problems in the assessment of evidence". Over the past decades progress in several applications has been sufficient to show that although the independent condition is not explicitly specified in the applications, with an independence assumption Dempster's rule still demonstrates its effectiveness [1,6,17,39,49].

It should be emphasized that this study is focused on developing a more efficient and effective computational method for more convincing practical applications based on the DS theory, rather than on investigating alternative combination rules for dependent evidence sources and conflict management. More detailed discussions of these aspects can be found in recent studies [14,34].

9. Conclusion and future work

We have described an effective framework built on the Dempster–Shafer theory of evidence for combining multiple classifiers and expert opinions in classification and decision making systems. We have also developed a formalism for modelling classifier outputs in terms of *triplets* and *quartets* and the formulae for combining classifiers represented in the form of triplets that can be extended to the quartet. The distinguishing aspect of our class-indifferent method from class-aligned methods is in selecting the prioritized class decisions to be combined and in making use of ignorance in representing unknown and uncertain class decisions.

A range of experiments have been carried out over the thirteen UCI benchmark data sets. Our results show that the performance of the best combined classifiers is better than that of the best individuals on most of the data sets and the corresponding ensemble sizes are 2–7 where the ensemble sizes of 2 and 3 take 61.5% of the thirteen best ensembles. The comparative analysis among the structures of simplet, triplet, quartet and dichotomous structure identifies the triplet as best, and the comparison made between DS and MV shows that DS is better than MV in combining the individual classifiers.

We have used the κ statistic to examine the extent of agreement of the different combination methods in deciding classes for testing instances. The statistics reveal that the classification performance achieved by using our DS combination method under our evidential structures is reliable. However, in this work we did not touch the issue of classifier diversity. Although most successful ensemble methods encourage diversity to some extent, a recent study on ten diversity measures raised the question of the what role diversity plays in constructing effective ensembles of classifiers and how it could be measured [29]. To our knowledge there has not been a conclusive study showing which measure of diversity is best for use in evaluating ensembles. We are working on this and we will discuss our findings in a future paper. Moreover, we would like to carry out a comparative study with alternative combination rules in the future. These include a new combination rule for combining non distinct items of evidence more recently introduced in [14] and the unnormalized combination rule introduced in [45].

Acknowledgements

The authors would like thank the anonymous reviewers for their detailed constructive comments which have helped us improve the paper considerably.

References

[1] A. Al-Ani, M. Deriche, A new technique for combining multiple classifiers using the Dempster–Shafer theory of evidence, J. Artif. Intell. Res. 17 (2002) 333–361.
[2] J.A. Barnett, Computational methods for a mathematical theory of evidence, in: Proc. of 17th Joint Conference of Artificial Intelligence, 1981, pp. 868–875.
[3] C.L. Blake, C.J.E. Keogh, UCI repository of machine learning databases, http://www.ics.uci.edu/mlearn/MLRepository.html.
[4] Y. Bi, D. Bell, J.W. Guan, Combining evidence from classifiers in text categorization, in: Proc. of KES04, 2004, pp. 521–528.
[5] Y. Bi, Combining multiple classifiers for text categorization using the Dempster–Shafer theory of evidence, PhD thesis, University of Ulster, UK, 2004.
[6] D. Bell, J.W. Guan, Y. Bi, On combining classifiers mass functions for text categorization, IEEE Trans. Knowledge Data Engrg. 17 (10) (2005) 1307–1319.
[7] Y. Bi, J.W. Guan, An efficient triplet-based algorithm for evidential reasoning, in: Proc. of the 22nd Conference on Uncertainty in Artificial Intelligence, 2006, pp. 31–38.
[8] Y. Bi, S.I. McClean, T. Anderson, On combining multiple classifiers using an evidential approach, in: Proc. of the Twenty-First National Conference on Artificial Intelligence (AAAI'06), 2006, pp. 324–329.
[9] L. Breiman, Bagging predictors, Machine Learning 24 (2) (1996) 123–140.
[10] L. Breiman, Random forests, Machine Learning 45 (1) (2001) 5–32;
T. Dietterich, Machine learning research: Four current directions, AI Magazine 18 (4) (1997) 97–136.
[11] T. Dietterich, An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting, and randomization, Machine Learning 1 (22) (1998).
[12] T. Dietterich, Ensemble methods in machine learning, in: Proc. 2nd Int. Workshop on Multiple Classifier Systems MCS2000, LNCS, vol. 1857, 2000, pp. 1–15.
[13] A.P. Dempster, Upper and lower probabilities induced by a multivalued mapping, Ann. Math. Stat. 38 (1967) 325–339.

[14] T. Denoeux, Conjunctive and disjunctive combination of belief functions induced by nondistinct bodies of evidence, Artif. Intell. 172 (2–3) (2008) 234–264.
[15] T. Denoeux, A. Ben Yaghlane, Approximating the combination of belief functions using the fast Moebius transform in a coarsened frame, Internat. J. Approx. Reason. 31 (1–2) (2002) 77–101.
[16] T. Denoeux, A neural network classifier based on Dempster–Shafer theory, IEEE Trans. Systems Man Cybernet. A 30 (2) (2000) 131–150.
[17] T. Denoeux, A *k*-nearest neighbor classification rule based on Dempster–Shafer theory, IEEE Trans. Systems Man Cybern. 25 (5) (1995) 804–813.
[18] R.P.W. Duin, D.M.J. Tax, Experiments with classifier combining rules, in: J. Kittler, F. Roli (Eds.), Multiple Classifier Systems, 2000, pp. 16–29.
[19] S. Dzeroski, B. Zenko, Is combining classifiers with stacking better than selecting the best one? Machine Learning 54 (3) (2004) 255–273.
[20] J.L. Fleiss, J. Cuzick, The reliability of dichotomous judgments: unequal numbers of judgments per subject, Appl. Psycholog. Meas. 3 (1979) 537–542.
[21] Y. Freund, R. Schapire, Experiments with a new boosting algorithm, in: Machine Learning: Proceedings of the Thirteenth International Conference, 1996, pp. 148–156.
[22] J.W. Guan, D.A. Bell, Evidence Theory and Its Applications, vols. 1–2, Studies in Computer Science and Artificial Intelligence, vols. 7–8, Elsevier, North-Holland, 1991–1992.
[23] J.W. Guan, D.A. Bell, Efficient algorithms for automated reasoning in expert systems, in: The 3rd IASTED International Conference on Robotics and Manufacturing, 1995, pp. 336–339.
[24] R. Haenni, Are alternatives to Dempster's rule of combination real alternatives?: Comments on "about the belief function combination and the conflict management problem" Lefèvre et al., Information Fusion 3 (3) (2002) 237–239.
[25] L.K. Hansen, P. Salamon, Neural network ensembles, IEEE Trans. Pattern Anal. Machine Intell. 12 (10) (1990) 993–1001.
[26] T.K. Ho, The random subspace method for constructing decision forests, IEEE Trans. Pattern Anal. Machine Intell. 20 (8) (1998) 832–844.
[27] J. Kittler, M. Hatef, R.P.W. Duin, J. Matas, On combining classifiers, IEEE Trans. Pattern Anal. Machine Intell. 20 (3) (1998) 226–239.
[28] L. Kuncheva, Combining classifiers: Soft computing solutions, in: S.K. Pal, A. Pal (Eds.), Pattern Recognition: From Classical to Modern Approaches, 2001, pp. 427–451.
[29] L. Kuncheva, C.J. Whitaker, Measures of diversity in classifier ensembles, Machine Learning 51 (2003) 181–207.
[30] A.K. Jain, R.P.W. Duin, J. Mao, Statistical pattern recognition: A review, IEEE Trans. Pattern Anal. Machine Intell. 22 (1) (2000) 4–37.
[31] L. Lam, C.Y. Suen, Application of majority voting to pattern recognition: An analysis of its behavior and performance, IEEE Trans. Systems Man Cybernet. 27 (5) (1997) 553–568.
[32] L.S. Larkey, W.B. Croft, Combining classifiers in text categorization, in: Proceedings of SIGIR-96, 19th ACM International Conference on Research and Development in Information Retrieval, 1996, pp. 289–297.
[33] W. Liu, J. Hong, Reinvestigating Dempster's idea on evidence combination, Knowledge Inform. Syst. 2 (2) (2000) 223–241.
[34] W. Liu, Analyzing the degree of conflict among belief functions, Artif. Intell. 170 (11) (2006) 909–924.
[35] E.J. Mandler, J. Schurmann, Combining the classification results of independent classifiers based on Dempster–Shafer theory of evidence, Pattern Recogn. Artif. Intell. X (1988) 381–393.
[36] P. Melville, R.J. Mooney, Constructing diverse classifier ensembles using artificial training examples, in: Proc. of IJCAI-03, 2003, pp. 505–510.
[37] T. Mitchell, Machine Learning, McGraw Hill, 1997.
[38] D. Opitz, Feature selection for ensembles, in: Proc. of AAAI-99, AAAI Press, 1999, pp. 379–384.
[39] B. Quost, T. Denoeux, M.-H. Masson, Pairwise classifier combination using belief functions, Pattern Recogn. Lett. 28 (5) (2007) 644–653.
[40] F. Sebastiani, Machine learning in automated text categorization, ACM Comput. Surv. 34 (1) (2002) 1–47.
[41] G. Rogova, Combining the results of several neural network classifiers, Neural Networks 7 (5) (1994) 777–781.
[42] G. Shafer, R. Logan, Implementing Dempster's rule for hierarchical evidence, Artif. Intell. 33 (3) (1987) 271–298.
[43] G. Shafer, Belief functions and possibility measures, in: J.C. Bezdek (Ed.), The Analysis of Fuzzy Information, vol. 1: Mathematics and Logic, CRC Press, 1987, pp. 51–84.
[44] G. Shafer, A Mathematical Theory of Evidence, Princeton University Press, Princeton, NJ, 1976.
[45] Ph. Smets, The combination of evidence in the Transferable Belief Model, IEEE Trans. Pattern Anal. Machine Intell. 12 (5) 447–458.
[46] K.M. Ting, I.H. Witten, Issues in stacked generalization, J. Artif. Intell. Res. (JAIR) 10 (1999) 271–289.
[47] D.M.J. Tax, M. van Breukelen, R.P.W. Duin, J. Kittler, Combining multiple classifiers by averaging or by multiplying, Pattern Recognition 33 (9) (2000) 1475–1485.
[48] K. Tumer, G.J. Robust, Combining of disparate classifiers through order statistics, Pattern Anal. Appl. 6 (1) (2002) 41–46.
[49] L. Xu, A. Krzyzak, C.Y. Suen, Several methods for combining multiple classifiers and their applications in handwritten character recognition, IEEE Trans. System Man Cybernet. 2 (3) (1992) 418–435.
[50] L.M. Zouhal, T. Denoeux, An evidence-theoretic *k*-NN rule with parameter optimization, IEEE Trans. Systems Man Cybernet. C 28 (2) (1998) 263–271.
[51] Y. Yang, T. Ault, T. Pierce, Combining multiple learning strategies for effective cross validation, in: Proc. of ICML'00, 2000, pp. 1167–1182.
[52] F. Voorbraak, On the justification of Dempster's rule of combination, Artif. Intell. 48 (2) (1991) 171–197.
[53] G.I. Webb, MultiBoosting: A technique for combining boosting and wagging, Machine Learning 40 (2) (2000) 159–196.
[54] D. Wolpert, Stacked generalization, Neural Networks 5 (2) (1992) 241–259.
[55] I.H. Witten, E. Frank, Data Mining: Practical Machine Learning Tools and Techniques, second ed., Morgan Kaufmann, San Francisco, 2005.