December 2006

Information Market Based Decision Fusion

Johan Perols
University of South Florida

Kaushal Chari
University of South Florida

Manish Agrawal
University of South Florida

INFORMATION MARKET BASED DECISION FUSION

Design Science

Johan Perols
University of South Florida
Tampa, FL
jperols@coba.usf.edu

Kaushal Chari
University of South Florida
Tampa, FL
kchari@coba.usf.edu

Manish Agrawal
University of South Florida
Tampa, FL
magrawal@coba.usf.edu

Abstract

In this paper, we present Information Market based Fusion (IMF), a novel, multi-classifier combiner method for decision fusion that is based on information markets. IMF does not require training or a static ensemble composition, adjusts to changes in base-classifier accuracy, provides incentives for the base-classifiers to present truthful information, and integrates with existing multi-agent system (MAS) coordination mechanisms. We compare the effectiveness of two different IMF implementations to Majority (MAJ), Average (AVG), and Weighted Average (WAVG) schemes, using computational experiments involving 16 datasets from the UCI Machine Learning Repository and 20 different base-classifiers from Weka.

Keywords: Multi-classifier combination, decision fusion, information markets, software agents

Introduction

Multi-classifier combination (MCC) is a technique used to improve the classification performance of base-classifiers in various pattern recognition problems. Performance improvements can have important ramifications in a wide variety of applications such as document classification for spam blocking; character recognition for zip code detection; prognosis of head injury patients and injury severity determination of pediatric trauma patients in the medical field; and military surveillance. In MCC, individual classifiers commonly referred to as base-classifiers, classify objects based on inputs consisting of object feature vectors (see Figure 1). These classifications or decisions are then combined using a combiner method into a single decision about an object's class label.

The basic idea behind MCC is that different classifiers in an ensemble have different strengths and weaknesses, and therefore provide complementary information about the classification problem. These differences can be leveraged to improve classification performance by combining base-classifiers' decisions (Kittler, et al. 1998). Different combiner methods have been proposed and examined in the literature and have been categorized based on whether they require training or not. For example, Naive Bayes, Decision Templates and WAVG are combiner methods requiring training while AVG, MAJ and Product do not require training. Details on various combiner methods in the literature can be found in (Suen and Lam 2000). Existing results from the literature indicate that MCC overall provides performance benefits, and that MAJ and AVG perform either at similar level or significantly better than trained methods (Kittler, et al. 1998). Combiner methods that require training assume that ensemble base-classifier composition remains constant and that training performance is a good proxy for subsequent actual performance. To our knowledge, all current combiner methods assume that the base-classifiers truthfully provide their actual classification decisions and do not integrate with various MAS coordination mechanisms that could be used in MCC systems.

In this paper we propose an information market fusion approach (IMF) for multi-classifier combination that 1) does not require training, 2) can adapt to changes in ensemble composition and base-classifier accuracy, and 3) does not assume that the base-classifiers are cooperative agents. In evaluating the effectiveness of our proposed approach, we test two IMF implementations and explore whether factors such as the number of base-classifiers in the

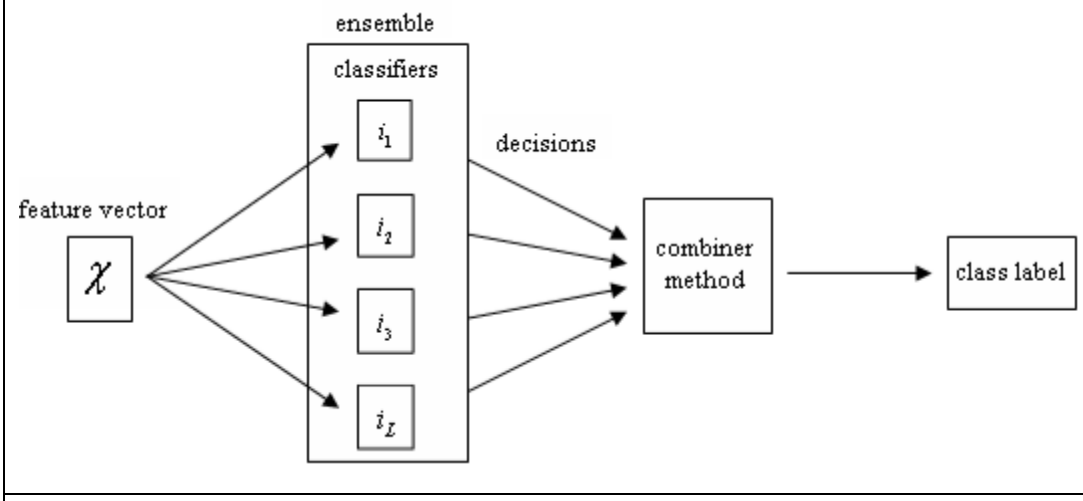

Figure 1. Generic Classifier Combiner Architecture

ensemble, nature of dataset, cutoff threshold, and costs and benefits associated with false and true positives, have any bearing on the performance of IMF implementations.

Related Research

Decision Fusion Overview

A classifier is a model that makes decisions about an object's class membership based on the object's feature set. Examples of classifiers include neural networks, logistic regression, decision trees, and Bayesian classifiers. The performance of any classifier is typically dependent on the problem domain as well as on the calibration of the classifier. Multiple classifiers are therefore tested in order to identify the best classifier for a given problem domain. Classifier comparisons have revealed that although a specific classifier may provide an overall relative performance advantage for a given classification problem, the classification errors it makes for certain cases may be avoided by an "inferior" classifier (Kittler, et al. 1998). Thus, there exists a potential to improve the overall prediction accuracy by combining the decisions of diverse classifiers. This is the basis for decision fusion.

In MCC, the decisions from multiple base-classifiers are combined with the goal of improving classification accuracy. MCC research has primarily focused on two areas: (1) what classifiers to include in the ensemble and how to train these classifiers; and (2) how to combine base-classifier decisions, the focus of this paper. Methods such as bagging and boosting fall into the first category, while combiner methods such as MAJ, AVG and WAVG fall into the second category. Bagging and boosting can use any combiner method, but normally use MAJ or AVG depending on the base-classifier output. As stated earlier, our research does not focus on classifier selection and training; instead we focus on the combiner method. Details on various combiner methods in the literature can be found in (Suen and Lam 2000). Prior research has found that: methods that use measurement data are typically more accurate than methods that handle unique labels; methods that require training typically outperform methods that do not require training (Suen and Lam 2000); and simple combiner methods MAJ and AVG perform either at almost the same level or significantly better than trained more complex methods (Kittler, et al. 1998).

Combiner Method Design Considerations

Another important, but largely overlooked, aspect of combiner methods is how well they fit with different system architectures. MCC problems are distributed in nature, but existing MCC designs view base-classifiers and combiner-methods as systems with centralized control. On the other hand, distributed problem solving (DPS) is of primary focus in MAS research and one of the main benefits of MAS architectures is the support they provide for DPS. Given the increasing popularity of MAS for various applications (Nissen and Sengupta 2006) and the support

MAS provides for DPS, MAS is a suitable architecture for MCC where the base-classifiers, combiner method and providers of object features can be implemented as agents. Existing combiner methods do not provide adequate support for two important MAS design considerations: 1) they assume that the agents, implementing classifiers, are cooperative, and 2) they do not integrate well with different MAS coordination mechanisms. Given that agents are competitive, and the combiner methods do not provide any incentives for the agents to provide their private information truthfully, it is not clear how existing combiner methods would get accurate information they need from base-classifiers implemented within competitive autonomous agents. Although there are workarounds for these problems, we do not believe that existing combiner methods naturally fit with MAS architectures based on competitive agents that operate in dynamic environments. Moreover, there is a lack of natural fit between MAS coordination mechanisms and existing combiner methods. For example, if we use a market based coordination mechanism and we want the better performing base-classifier agents to have more control over how tasks and resources are assigned to agents that gather and deliver object features, then the better performing base-classifier agents must somehow receive a relatively larger share of funds that they can use to purchase features. Existing combiner methods were not developed with this in mind and therefore do not support, without workarounds, such market-based coordination mechanisms.

Furthermore, existing trained combiner methods, as opposed to methods that do not require training, assume that the ensemble is made up of a static group of base-classifiers, i.e., classifiers are not added or deleted from the ensemble, and individual classifier performance does not change subsequent to training and validation. We next introduce two IMF combiner methods that overcome the problems discussed in this section.

Information Market Based Fusion

The combiner method proposed in this paper is theoretically grounded in information markets. More specifically the aggregation mechanisms used in IMF are based on Pari-mutuel betting markets.

Information Markets

Information markets are markets designed specifically for the purpose of information aggregation. Equilibrium prices, derived using conventional market mechanisms, provide information based on the private and public information maintained by the market participants about a specific situation, future event or object of interest (Hanson 2003). Although the concept of information markets is fairly recent, the underlying notion of markets being capable of aggregating information is not new. Hayek argued in the 1940s that prices in a competitive market efficiently aggregate information held by market participants (Hayek 1945). The efficient market hypothesis states that all private and public information is reflected in equilibrium prices, this even includes insider information under strong efficiency (Fama 1970).

More recently the idea of using markets for the specific purpose of aggregating information has received more attention. Experimental information market research has generally found support for the efficient market hypothesis (Forsythe and Lundholm 1990). Empirical research has found support for information aggregation in information markets (Berg and Rietz 2003) and research focusing on pari-mutuel betting has found support for the efficient market theory (Ali 1979). Based on these theoretical and empirical findings we believe that an information market can be designed to effectively combine the decisions of base-classifiers in an ensemble.

Pari-Mutuel Betting

Pari-mutuel betting originated in horse race gambling in France in 1865 and has since become a very popular betting mechanism in the horseracing world. Pari-mutuel means "wager mutual" and comes from the fact that in pari-mutuel betting, a winning wager (i.e., bet) receives a share of the total wagers (winning and loosing bets less track commission) as a proportion of this winning wager to all winning wagers.

Plott, et al. (2003) describe different pari-mutuel betting behaviors using two private information models (Decision Theory Private Information (DTPI) and Competitive Equilibrium Private Information (CEPI)), and one model with updated beliefs (Competitive Equilibrium Rational Expectations). They then investigate which one of these models best captures the trading behavior in a pari-mutuel market as observed in a laboratory setting. Plott, et al. (2003) found that DTPI, followed by CEPI overall best described the behavior of the market participants.

IMF

We propose two IMF algorithms based on DTPI and CEPI.

IMF_{dtpi}

In IMF_{dtpi} (see Figure 2 and Table 1), while classifying any object t, agent i (i.e. base-classifier i) maximizes its expected utility (see Figure 3) by betting q_{itj} on outcome j based on the agent's probability estimates p_{itj}. Agents maximize their expected utility by betting the budget amount b_{it} on the outcome j that they believe, given the current cutoff, is most likely to take place. The budget consists of three items: (1) an endowment of \$2 for each t as in (Plott, et al. 2003)[1]; (2) the agents' accumulated wealth; and (3) a risk adjustment of one thirtieth[2]. For each t the outcome with the highest total bets is selected. Note that agents behave myopically with regards to published odds as they place their bets based solely on their probability estimates p_{itj} that are derived by their internal models (such as neural network). Agents determine odds from a cutoff value that does not change across objects. Therefore IMF_{dtpi} only contains one betting round for each t where the odds are set at the cutoff.

IMF_{cepi}

In IMF_{cepi}, agents do not learn from published prices, i.e., they do not update their beliefs based on market odds (prices), but do base

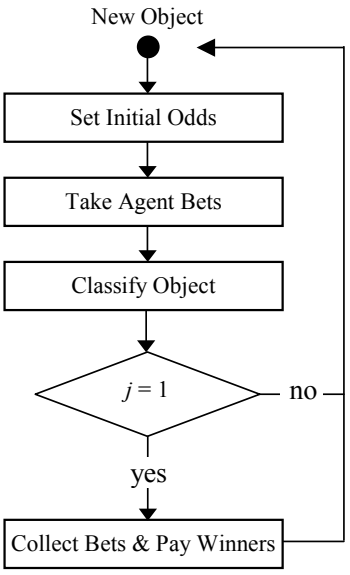

Figure 2. IMF_{dtpi} Algorithm

Table 1. IMF Notation

D	Index set of classifiers in an ensemble E.
t	Index of objects to classify.
J	Index set of all classes in which t can be classified.
p_{itj}	Posterior probability estimate of classifier $i \in D$, that t belongs to $j \in J$. p_{itj} in [0,1].
P_{tj}	Overall probability estimate of ensemble E that t belongs to $j \in J$. P_{tj} in [0,1].
O_{tj}	Overall odds for t belonging to $j \in J$. $O_{tj} = 1/P_{tj}$.
r_i	Risk aversion factor of classifier i. $0 < r_i \leq 1$.
m_{it}	Endowment received by i when classifying t.
w_{it}	Wealth of i prior to receiving m_{it}.
B_{it}	Total budget available to i when classifying t.
q_{itj}	Amount bet by i that t belongs to $j \in J$.
Q_{tj}	Total amount bet by all classifiers in E that t belongs to $j \in J$. $Q_{tj} = \sum_{i \in D} q_{itj}$.
Q_t	Total amount bet by all classifiers for all $j \in J$. $Q_t = \sum_{j \in J} Q_{tj}$.
c_j	Cutoff value for P_{tj} used to classify t.
P^l	Lower bound on P_{tj} in binary search.
P^u	Upper bound on P_{tj} in binary search.
O^u	Upper bound on O_{tj} in optimization.
O^l	Lower bound on O_{tj} in optimization.

[1] Note that the exact amount of this endowment is trivial (assuming that all agents are given the same endowment and it is consistent over time) as a specific fraction of wealth is always bet and the endowment is the only source of money in the market.

[2] In portfolio theory, risk adverse investors construct portfolios in order to optimize market risk for expected returns (Markowitz 1952) and in general portfolios consisting of between 20 and 30 stocks will provide close to optimal solutions. We recognize that these recommendations are based on the average diversity and volatility of stocks, but we believe that it is a reasonable rule of thumb to apply in our pari-mutuel market design.

their bets on current market prices. As depicted in Figure 4, for each new object the market odds are first set based on the current cutoff value (1):

$$O_{t1} = 1/C_1 \text{ and } O_{t2} = 1/(1-C_1) \quad (1)$$

The agents then decide their bets q_{itj} (see Figure 3) and the object is classified based on how the agents placed their bets given the cutoff based odds:

$$\text{if } (O_{t1}Q_{t1} \geq O_{t2}Q_{t2}) \text{ then} \quad (2)$$

$$\text{classify } t \text{ as class } j = 1 \quad (3)$$

$$\text{else classify } t \text{ as class } j = 2 \quad (4)$$

$$\max_{q_{itj}} \sum_{j \in J} p_{itj} q_{itj} O_{tj} - \sum_{j \in J} q_{itj}$$

$$\text{S.T.} \quad \sum_{j \in J} q_{itj} \leq B_{it}$$

$$B_{it} = m_{it} + w_{it} r_i$$

$$q_{itj} \geq 0 \quad \forall j \in J$$

Figure 3. Agent Bet

If the potential payout for $j = 1$ is higher than or equal to the potential payout for $j = 2$ then the current odds O_{t1} is not too low (2), the inverse of the equilibrium odds is higher than the cutoff used to determine O_{t1}, t is classified as being a member of the positive class (3) and the equilibrium odds are determined next. Otherwise t is classified as a member of the negative class, the processing of the object end and a new object is processed (4).

The equilibrium odds, i.e. when the potential payouts $Q_{tj}O_{tj}$ for each $j \in J$ and the total amount bet on all events Q_t are equal, are found for each t by iteratively updating the odds and agents placing bets using these odds until the odds provided to the agents and their subsequent bets result in $Q_{tj}O_{tj} = Q_t$, at which time the market closes. Note that 1) the bets placed before the final iteration are only used for the purpose of updating the odds, and 2) if agent bets are discontinuous over O_{tj} then the existence of equilibrium odds cannot be guaranteed (Carlsson, et al. 2001). Therefore, we use a combination of binary search and optimization to update the odds. The binary search algorithm (see Figure 5) updates the odds until the search space is deemed narrow enough (P^u - P^l <= 0.0001). Optimization is then used to attempt to find the optimal odds within this search space (see Figure 6). The odds found using optimization are then used to classify t. After the equilibrium odds have been determined the agents place their final bets (see Figure 3). The true class of t is then determined and the agents' wealth is decreased by the amount of the final bets and increased by the amount of winnings.

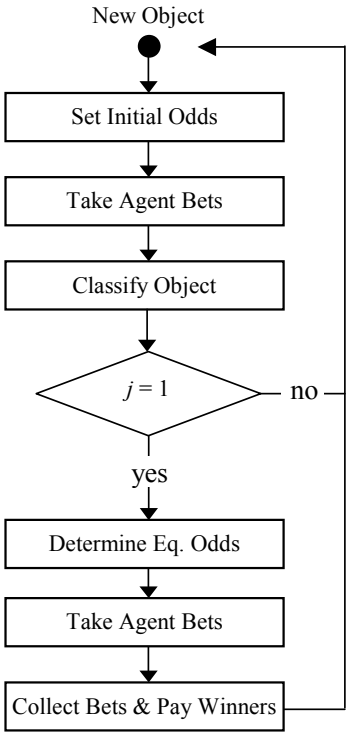

Figure 4. IMF$_{cepi}$ Algorithm

Note that the agents' wealth is not updated when t is classified as j=2 (4), the reason being that we will only find out the actual object class if the classification warrants further investigation. For example, in fraud detection applications, the true class of an object will only be known if the object is classified as a potential fraud. This is a conservative assumption that works against IMF$_{dtpi}$ and IMF$_{cepi}$.

Experiment

In this section we describe the base-classifiers and datasets used in the computational experiments. We then discuss the experimental design and the experimental factors, and present our hypotheses. This section is concluded with a brief description of the experimental procedure.

Base-Classifiers and Data

Set search space limits, $P^u = 1$ and $P^l = 0$
set $O_{t1} = 2/(P^l + P^u)$
Take agent bets
Do
  If $Q_{t1}O_{t1} \geq Q_t$ then
    set $P^l = 1/O_{t1}$
  else if $Q_{t1}O_{t1} < Q_t$ then
    set $P^u = 1/O_{t1}$
  else
    set P^u and P^l to $1/O_{t1}$
  set $O_{t1} = 2/(P^l + P^u)$
  Take agent bets
until odds are in equilibrium or search space is narrow enough

Figure 5. Binary Search

Set optimization boundaries $O^u = 1/P^l$ and $O^l = 1/P^u$
Set O_{t1} using optimization:

$$\min_{O_{ij}M_j} \sum_{j \in J} M_j$$

S.T. $Q_{t1}O_{t1} - M_1 = Q_t$

$Q_{t2}O_{t2} - M_2 = Q_t$

$O^u \geq O_{t1} \geq O^l \quad \forall j \in J$

$O_{t2} = O_{t1}/(O_{t1} - 1)$

Figure 6. Odds Optimization

Using Weka (version 3.4.6) 20 heterogeneous base-classifiers were created using their default settings (see Table 2). The base-classifiers were trained and evaluated using 10-fold cross validation on each of 16 datasets that were obtained from the UCI Machine Learning Repository (Table 3). Datasets that included more than two class labels were modified by either creating multiple subsets with only two class labels in each subset or by combining class labels. Furthermore, in order for stronger base-classifiers to complete the classification using an acceptable amount of resources (primarily memory and time) datasets with a large number of observation and/or attributes were filtered randomly based on records and/or attributes.

Experimental Design and Factors

The primary purpose of the computational experiments was to compare IMF and other combiner methods for their effectiveness. As such, combiner method is our primary factor of interest. Three additional factors, number-of-agents, cutoff and dataset, were also manipulated to examine how these factors affected the relative performance of these methods. Number-of-agents in the ensemble was manipulated at 3, 9, 15 and 20 agents. The cutoff, which refers to the probability threshold used in determining the class label prediction, was manipulated at 5 levels: 0.1,

Table 2. Base-Classifiers

ADTree	NaiveBayes
BayesNet	NBTree
ConjunctiveRule	NNge
DecisionStump	OneR
DecisionTable	PART
IBk	RandomForest
J48	RBFNetwork
JRip	Ridor
LWL	SimpleLogistic
MultilayerPerceptron	SMO

Table 3. Datasets (Positive Ratio)

W_BreastCancer (34%)	PimaDiabetes (35%)
ContraceptiveMethod (24%)	Thyroid (8%)
H_Colic (37%)	Labor (65%)
Covertype1&2 (73%)	Mushrooms (48%)
Covertype3&4 (7%)	Sick (6%)
Covertype5&6 (67%)	Spambase (39%)
aCredit (56%)	SpliceGeneSequences (52%)
gCredit (30%)	Waveform (49%)

0.3, 0.5, 0.7 and 0.9. Based on these factors and respective treatment levels we had a 5×4×5×16 factorial design.

Dependent Measures

Five outcome measures: sensitivity, specificity, false positive, false negative and hit-rate were used to measure the performance of different combiner methods. Each measure emphasizes a particular facet of performance whose importance could vary based on the application domain. Due to space limitations, we will confine our discussions in this paper to hit-rate only, however the results were similar for the other ratios as well. Hit-rate refers to the overall percentage of correctly classified events and nonevents.

Combiner Method Factor

IMF_{cepi} and IMF_{dtpi} were compared to MAJ, AVG and WAVG, given that prior research indicated that AVG and MAJ performed relatively well, while WAVG was included primarily because of its similarity to IMF_{cepi}. IMF and WAVG were only allowed to learn from objects that were classified into the positive class (i.e., fraud). Because of this learning constraint we could not implement the commonly used WAVG implementation, as this required training data. We therefore implemented a dynamic version of WAVG where the weights were determined based on individual classifiers' relative accuracy in the "investigated" cases.

Interacting Factors

The inclusion of dataset, cutoff and the number-of-agents allows us to examine potential interactions between the combiner method factors and these additional factors. Significant interactions indicate that the main effect results are sensitive to the moderating factor and we investigate these interactions rather than the main effect results to evaluate the relative performance of the combiner methods.

Hypotheses

Based on these discussions our first three hypotheses based on the interactions were:

H1 *The combiner method performance is moderated by the dataset.*

H2 *The combiner method performance is moderated by the cutoff.*

H3 *The combiner method performance is moderated by the number-of-agents.*

The hypotheses based on the method main effect, which is only investigated if H1, H2 and H3 are insignificant, were:

H4 *There is a performance difference among the combiner methods.*

H4a *IMF_{cepi} outperforms the other combiner methods.*

H4b *IMF_{dtpi} outperforms the other combiner methods.*

Table 4. Effect Tests

Source	p-value
Method	<.0001
Data	<.0001
Cutoff	<.0001
Number-of-Agents	<.0001
Method*Cutoff	<.0001
Method*Data	<.0001
Method*Number-of-Agents	0.0523

Experimental Procedure

In the experiment the classification output from Weka (320 classifier output tables, 16 datasets times 20 classifiers) was imported into Microsoft Access where combiner methods created in Visual Basic and LINGO combined the data. Each dataset was combined 100 times (4×5×5) and a total of 1600 observations were obtained based on 16 datasets being analyzed.

Results

We started by evaluating H1, H2 and H3, by analyzing the effects of the interactions between method and the other factors on hit-rate (see Table 4). With alpha=0.05 and a Bonferroni adjustment (0.05/7=0.0071) the method*data ($p<0.001$) and method*cutoff ($p<0.001$) interactions were significant, while method*number-of-agents ($p=0.0523$) was insignificant. As two of the interaction hypotheses (H1 and H2) were supported, H4 was not evaluated further.

To gain a better understanding of the nature of the significant interactions we plotted the interactions against the Least Square Mean (LSM) hit-rate, see Figure 7 and Figure 8[3] for the method*dataset and method*cutoff interactions respectively. Note from Figure 7 that the performance differences are fairly consistent across different datasets, i.e., IMF_{dtpi} appears to be the highest performer while IMF_{cepi} performs the worse in all datasets except for Splice where MAJ performs the worse. Furthermore, it appears as if the overall significance is due to MAJ, AVG and WAVG exhibiting interaction effects with respect to datasets. We confirmed this by running an ANOVA with method as the independent variable, hit-rate as the dependent variable and dataset as a blocking variable. The ANOVA showed that there was a significant method difference ($p<0.001$) and a Tukey HSD (alpha=0.05) post-hoc analysis showed that IMF_{dtpi} significantly outperformed, and that IMF_{cepi} was significantly outperformed by the other combiner methods, while there was no difference between AVG, MAJ and WAVG. Turning to the method*cutoff interaction, note in Figure 8 how IMF_{dtpi} outperformed the other methods at all cutoffs and that the differences became larger at extreme cutoffs. Also note that IMF_{cepi} performs worse than other methods at all cutoffs except for at 0.5, and that again this trend is more apparent at extreme cutoffs. These observations were confirmed using Tukey HSD (alpha=0.05). At cutoff 0.1, all method performance differences were significant except for the difference between WAVG and AVG, while there were no significant differences between the

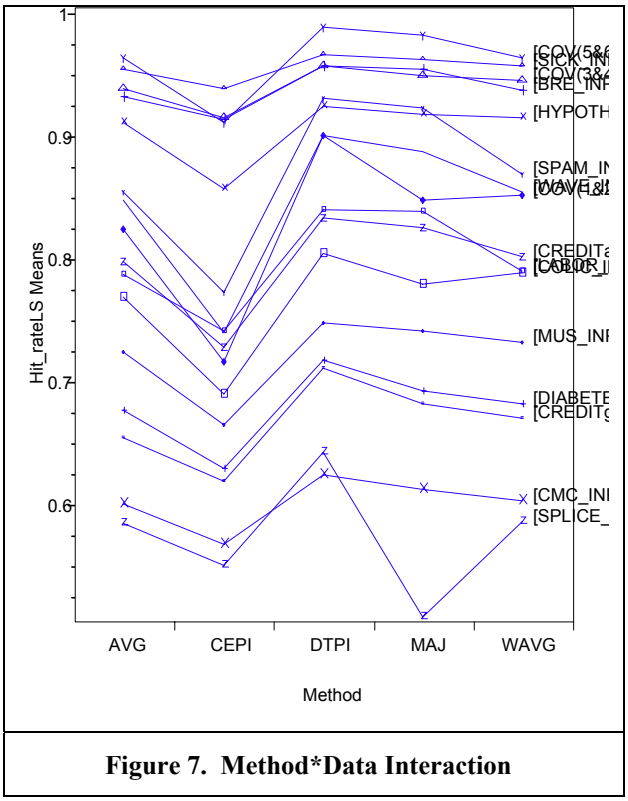

Figure 7. Method*Data Interaction

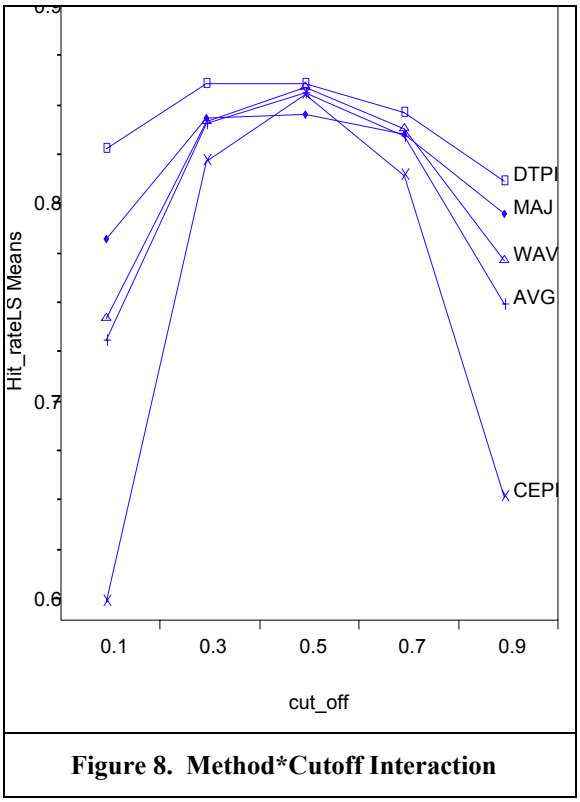

Figure 8. Method*Cutoff Interaction

[3] When using hit-rate as performance measure we implicitly assume that the ensemble prediction is not cost-sensitive and consequently that the cutoff is 0.5 (we thank one of the reviewers for pointing this out). Although this means that we do not need to investigate the sensitivity of the hit-rate results in regards to cutoff we included this analysis as it illustrates why we decided to use net benefit instead of hit-ratio (see the Discussion section). Considering that this is a research in progress article we believe that it is appropriate to show this progression in our research.

methods at cutoffs 0.3, 0.5 and 0.7. At cutoff 0.9 IMF_{dtpi} significantly outperformed AVG and IMF_{cepi}, while IMF_{cepi} was significantly outperformed by all the other methods.

Discussion

The interaction results presented thus far show that IMF_{dtpi} is the most effective combiner method while IMF_{cepi} is the least effective method. Although the analysis above is inline with prior combiner method comparisons, and arguable even provides more insight as the analysis includes an investigation of the sensitivity of the results, we do not believe that the conclusions we have made so far about IMF_{dtpi} and IMF_{cepi} take all important aspects into account, such as tradeoff between benefits and costs derived from true positives and false positives respectively.

Given that all combiner methods have their highest performance at cutoff level 0.5 (see Figure 8), one may question why a cutoff other than 0.5 would be chosen. With a lower cutoff, more objects are classified as positives, which has the effect of increasing both the number of true positives and false positives. As we can see in Figure 8 the hit-rate drops when the cutoff is lowered thereby suggesting that the number of false positives increases more than the number of true positives. This is a desirable effect as long as the marginal benefit of the additional true positives is higher than the marginal cost of additional false positives (the opposite is true when the cutoff is raised). Therefore, based on hit-rate we cannot conclude whether sharp drops at the extreme cutoffs exhibited by IMF_{cepi} as compared to the other methods is undesirable or not. In order to determine which combiner method has the best performance we need to compare the net benefit that each of the methods provide given various benefits and costs associated with true and false positives, and true and false negatives[4]. Alternatively, we can use Receiver Operator Characteristics (ROC) curves to investigate the trade-offs between sensitivity and specificity that show how much the false positive rate increases given a certain increase in the true positive rate. We have not yet started the ROC analysis but we are currently in the middle of running the cost analysis, which we describe next.

Cost Analysis

We compared the relative performance of the different combiner methods using seven different true positive to false positive ratios (500:1; 100:1; 10:1; 1:1; 1:10; 1:100; and 1:500). To better understand these ratios consider the 100:1 ratio as an example. In the case of a fraud audit the ratio could represent that the average benefit of detecting a fraud is $10,000 while the average investigation is $100. Say that we investigate 100 cases and 5 of these cases turn out to be fraudulent, then the net benefit from these investigations is $40,000 (10,000*5-100*100).

We used these ratios to calculate the net benefits based on an experiment with eight datasets (colic, labor, aCredit, gCredit, hypothyroid, diabetes, wave and splice), four number-of-agent treatments (3, 9, 15 and 20) and 99 treatment levels for cutoff (0.01, 0.02…0.99). With net benefit as the response variable we found the cutoff*method interaction to be significant ($p<0.001$) and the number-of-agents*method interaction to be insignificant (p-values between 0.49 and 0.93).

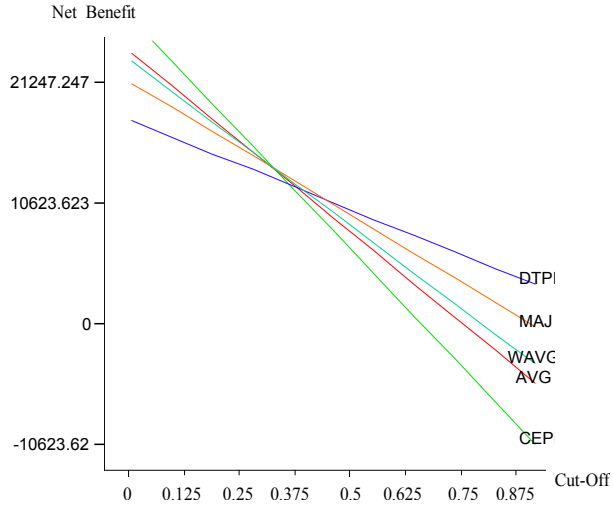

Figure 9. Net Benefit Interaction Profile for 100:1

[4] The specific cost matrix assumption also impacts the models generated by base-classifiers outputting crisp labels. These base-classifiers were however not retrained based on the different cost assumptions. Although this impacts the ensemble results, just as choosing different parameters for the base-classifiers does, we do not believe that this introduces a systematic bias in the relative performance of the combiner methods as the methods use the same base-classifiers. To validate this claim we reran the experiment using only base-classifiers with measurement level output and found similar results as earlier.

To get a better idea of the nature of the significant interaction we plotted the interaction LSM, see Figure 9 for an example where the cost ratio is 100:1. Other graphs were similar to Figure 9 except for 1:1 (the net benefit based on 1:1 is actually the same measure as hit-rate so this was expected). Figure 9 shows that when the cutoff decreases the net benefit increases for all combiner methods and that IMF_{cepi} has the highest overall performance and outperforms the other methods if the cutoff is selected between 0.01 through 0.33. After this point IMF_{dtpi} has the best performance.

Figure 9 is based on net benefit regression predictions that are assumed to be straight lines rather than curve linear and as such do not show the relative performance of the methods at optimal cutoff levels. To gain additional insights into the relative performance differences we have implemented an algorithm that dynamically selects the cutoff for each method. This algorithm periodically changes the cutoffs one step up and then one step down from a base cutoff using a delta factor. The cutoff generating the highest net benefit is then assigned as the base level and the cycle is repeated.

Table 5. Research Objectives and IMF

Objective	Solution
Ensemble training	Agent wealth is updated based on positive classifications only.
Agent cooperativeness	The market mechanism uses the amount bet on each outcome and not probabilities to determine equilibrium odds.
Changes in ensemble composition	The market mechanism works independent of the specific agents participating in the market.
Coordination mechanism integration	The currency used in the market can be used for coordination. Better performing agents then have a larger influence on the coordination.
Changes in base-classifier accuracy	Agents improving their performance do better in the market and vice versa, and have an increasingly larger influence on the ensemble's decision.

We are currently in the middle of running this experiment and have at this point completed the 500:1 and 100:1 ratios. Results show that while IMF_{cepi} significantly outperforms MAJ and IMF_{dtpi}, there is no significant difference between IMF_{cepi}, WAVG and AVG (Tukey HSD alpha=0.05). These results support the interactions depicted in Figure 9. While the performance differences between IMF_{cepi}, AVG and WAVG are insignificant the results so far show that IMF_{cepi} consistently, though insignificantly, outperforms AVG and WAVG, which is also supported by the results presented in Figure 9.

Summary and Future Research Opportunities

In this paper we have presented and evaluated IMF, a novel combiner method that is grounded in information market theory. While we are still evaluating the effectiveness of IMF, we do believe that our design addresses other research objectives (see Table 5). IMF is based on a market mechanism, a fundamentally different approach compared to prior combiner methods. IMF adapts to changes in ensemble composition without having to assume that base-classifiers are cooperative agents. IMF can be integrated with various MAS coordination mechanisms.

Our next goal in this project is to complete the statistical net benefit analysis to get a better idea of the effectiveness of IMF. We are also working on incorporating more commonly used utility functions, such as natural logarithm and constant relative risk aversion, to replace the current betting function. A logical extension to this research would be to modify IMF for classification problems with more than two classes. Considering that odds in actual pari-mutuel betting markets are established for more than two potential outcomes (i.e. multiple horses can potentially wine a race) it appears that such extension is possible.

References

Ali, M. M. "Some Evidence of the Efficiency of a Speculative Market," *Econometrica* (47:2), March 1979, pp. 387-392.

Berg, E. J., and Rietz, A. T. "Prediction Markets as Decision Support Systems," *Information Systems Frontiers* (5:1), January 2003, pp. 79-93.

Carlsson, P., Ygge, F., and Andersson, A. "Extending Equilibrium Markets," *IEEE Intelligent Systems* (16:4), July/August 2001, pp. 18-26.

Fama, E. "Efficient capital market: a review of theory and empirical work," *Journal of Finance* (25:2), May 1970, pp. 383-417.

Forsythe, R., and Lundholm, R. "Information Aggregation in an Experimental Market," *Econometrica* (58:2), March 1990, pp. 309-347.

Hanson, R. "Combinatorial Information Market Design," *Information Systems Frontiers* (5:1), 2003, 107–119.

Hayek, A. F. "The Use of Knowledge in Society," *The American Economic Review* (35:4), September 1945, pp. 519-530.

Kittler, J., Hatef, M., Duin, R. P. W., and Matas, J. "On combining classifiers," *IEEE Transactions on Pattern Analysis and Machine Intelligence* (20:3), March 1998, pp. 226-239.

Markowitz, H. "Portfolio Selection," *The Journal of finance* (7:1), 1952, pp. 77-91.

Nissen, E. M., and Sengupta, K. "Incorporating Software Agents into Supply Chains: Experimental Investigation with a Procurement Task," MIS Quarterly (30:1), March 2006, pp. 145-166.

Plott, R. C., Wit, J., and Yang, C. W. "Parimutuel betting markets as information aggregation devices: experimental results," *Economic Theory* (22:2), September 2003, pp. 311-351.

Suen, Y. C., and Lam, L. "Multiple Classifier Combination Methodologies for Different Output Levels," *Lecture Notes in Computer Science* (1857), January 2000, pp. 52-66.

Wolfers, J., and Zitzewitz, E. "Interpreting Prediction Market Prices as Probabilities," in *Proceedings of the Allied Social Science Association Annual Meeting*, American Economic Association, Boston, MA, January 2006.