The Characterisation of Predictive Accuracy and Decision Combination

Kai Ming Ting
Department of Computer Science,
The University of Waikato,
Private Bag 3105, Hamilton, New Zealand.
kaiming@cs.waikato.ac.nz

Abstract

In this paper, we first explore an intrinsic problem that exists in the theories induced by learning algorithms. Regardless of the selected algorithm, search methodology and hypothesis representation by which the theory is induced, one would expect the theory to make better predictions in some regions of the description space than others. We term the fact that an induced theory will have some regions of relatively poor performance *the problem of locally low predictive accuracy.* Having characterised the problem of locally low predictive accuracy in Instance-Based and Naive Bayesian classifiers, we propose to counter this problem using a composite learner that incorporates both classifiers. The strategy is to select an estimated better performing classifier to do the final prediction *during* classification. Empirical results show that the strategy is capable of partially overcoming the problem and at the same time improving the overall performance of its constituent classifiers. We provide explanations of why the proposed composite learner performs better than the cross-validation method and the better of its constituent classifiers.

1 INTRODUCTION AND MOTIVATION

One important measure of performance used in the machine learning community for learning algorithms is estimated *accuracy* on unseen cases. This measure indicates the estimated overall performance of a learned theory over the whole population from which the training dataset used to induce the theory is generated. However, no one would assume that every prediction made by a theory would have the same probability of being correct (i.e., proportional to its overall estimated accuracy). Regardless of the selected algorithm, search methodology and hypothesis representation by which the theory is induced, one would expect the theory to make better predictions in some regions of the description space than others. Holte, Acker and Porter (1989) were the first to address this intrinsic problem in learning systems that describe the induced theory as a disjunction of conjunctions of conditions. They use the coverage of each rule (or disjunct) to characterise this intrinsic problem. They observe that rules that cover a small number of instances often entail high error rates; thus they name the phenomenon *the problem of small disjuncts.* We would anticipate the existence of a similar problem in Instance-Based Learning (IBL) algorithms (Aha, Kibler & Albert, 1991) and Naive Bayesian classifiers or Naive Bayes (Cestnik, 1990). Because both of these algorithms employ different representations from decision trees and rules, measures other than disjuncts are needed to characterise this intrinsic problem in these representations. This consideration has prompted us to rename the problem so that it can be brought to a more general perspective. We term the fact that an induced theory would have some regions of relatively poor performance **the problem of locally low predictive accuracy**.

Very little research has been carried out to gain insight into this problem that is intrinsic to all learning algorithms. The difficulties in this research are (i) how to characterise the various regions of differing predictive accuracies, and (ii) how to estimate predictive accuracy in these regions. The general strategy of our research is to probe into the underlying characteristics of the learning methods to look for an answer. We use the term – *the characterisation of predictive accuracy* (hereafter shortened to *the characterisation*) to mean the use of a measure in an induced theory as an indicator for its predictive accuracy. To address the second difficulty, we estimate the predictive accuracy of the characterisation using a cross-validation method.

Once we can characterise predictive accuracy of each prediction of the induced theories, we can overcome the problem by replacing low performance regions in one theory (from one algorithm) with those of other theory (from another algorithm) which have higher predictive accuracy. The proposed decision combination approach dynamically selects a learning algorithm in each prediction *during* classification.

We will describe how predictive accuracy of each prediction in a IBL algorithm and a Naive Bayesian classifier can be characterised in Section 2. Then in Section 3, we show how the characterisation can be used to overcome the problem of locally low predictive accuracy by working cooperatively between the two algorithms in a composite learner framework. Section 4 describes some related work and Section 5 summarises the findings.

2 CHARACTERISING PREDICTIVE ACCURACY IN IBL AND NAIVE BAYES

We investigate two intuitive methods of characterising the accuracy of each prediction in this section. For Naive Bayes, the natural method is *posterior probability* – predictions made with high *posterior probability* would be more likely to be correct than those made with low *posterior probability*. This characterisation agrees with the Bayesian approach in decision making. We explore an intuitive method for characterising the accuracy of each prediction in a IBL algorithm, namely *typicality*. This characterisation has its root in cognitive psychology (Rosch & Mervis, 1975). In our setting, the *typicality* of an instance (Y) is defined as (Zhang, 1992; Ting, 1994b):

$$Typicality(Y) = \frac{(\sum_{i=1}^{n} Euclid(X_i, Y))/n}{(\sum_{j=1}^{p} Euclid(X_j, Y))/p} \qquad (1)$$

where the numerator denotes the *inter-concept distance* of an instance (Y) which is defined as its average (Euclidean) distance to instances of different classes (n), and the denominator denotes the *intra-concept distance* of an instance which is defined as its average distance to other instances of the same class (p).

Thus, an atypical instance would have a low value of the *typicality* measure and vice versa. In the framework of IBL, we would expect a prediction made with a typical nearest neighbour is more likely to be correct than one made with an atypical nearest neighbour.

The two algorithms IB1-MVDM* and NB* (Ting, 1994c; 1996) are used to test the above characterisations. IB1-MVDM* is a variant of IB1 (Aha, Kibler & Albert, 1991) that incorporates the modified value-difference metric (Cost & Salzberg, 1993) and NB* is an implementation of the Naive Bayes (Cestnik, 1990) algorithm. Both algorithms include a method (Fayyad & Irani, 1993) for discretising continuous-valued attributes in the preprocessing (indicated by "*"). This preprocessing improved the performance of the two algorithms in most of the continuous-valued attribute domains studied by Ting (1994c). We use the nearest neighbour for making prediction in IB1-MVDM* and the default settings are as used in IB1[1] in all experiments. No parameter settings are required for NB*. Specifically, we conduct experiments to test the following hypotheses in the next section:

H1: *the higher the posterior probability of a prediction, the higher its predictive accuracy* in NB*.

H2: *the more atypical a nearest neighbour used in a prediction, the lower its predictive accuracy* in IB1-MVDM*.

2.1 EXPERIMENTS

We use a *cross-validation* approach to estimate predictive accuracy of the characterisation. Three-fold cross-validation is used in all experiments instead of ten-fold cross-validation (Breiman et al, 1984; Schaffer, 1993), because it is faster and provides comparable results when used to combine the two algorithms in Section 3.

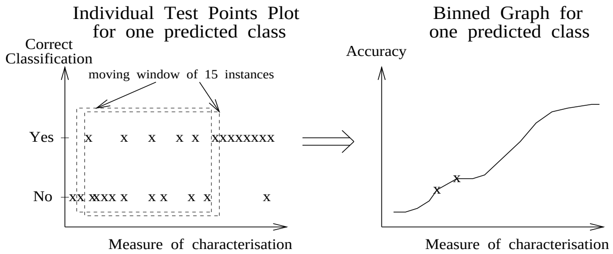

Figure 1: Transforming individual cross-validation test points to a binned graph for one predicted class

The individual test results from the three-fold cross-validation are first sorted according to the values of the characterisation (i.e., *posterior probability*, or *typicality*), and then they are used to produce a binned graph that relates the average value of the characterisation to its binned predictive accuracy for each class. Figure 1 depicts the process of transforming the cross-validation test results of individual instances (i.e., correct or incorrect classifications) to a binned graph for one predicted class. Each bin contains a fixed number of instances (half of the total number for each class or fifteen[2], whichever is bigger). Each point in the graph

[1] IB1 stores all training instances and uses maximum differences for attributes that have missing values, and computes Euclidean distance between any two instances.

[2] This number is about 10% of one of the small datasets. It is rather ad hoc, but it should not be too big to accommodate small datasets or too small to avoid graph fluctuation due to minor changes in the bin.

is produced from a "moving window", i.e., the next bin is obtained by dropping the leftmost instance and adding an instance adjacent to the rightmost instance of the current bin[3]. When the bin size is bigger than fifteen, the end points of the graph are extended by reducing the bin size one instance at a time until the bin size reaches fifteen. This is to ensure all graphs produced have the same bin size of fifteen at the end points. Note that we use the binned graphs for classification rather than function approximation problems.

We conduct the experiments in eleven well-known domains obtained from the UCI repository of machine learning databases (Murphy & Aha, 1994). They are the breast cancer Wisconsin (bcw), pima diabetes (diab), waveform-40 (wave), Cleveland heart disease (heart), hypothyroid (hypo), hepatitis (hepa), echocardiogram (echo), horse colic, soybean (soyb), glass and automobile (the last two domains have six classes and the soybean domain has nineteen classes). The example binned graphs of two domains are shown in Figures 2 and 3, which are produced from one trial using 90% of the dataset in each domain. Separate graphs are produced for each class in each domain for the two methods of characterisation, i.e., typicality for IB1-MVDM* and probability for NB*.

We also use a fairly robust (sometimes termed "distribution-free") test of significance, i.e., a Wilcoxon rank-sum test (see e.g., Howell (1982)) on the raw data (e.g., the left diagram depicted in Figure 1) to see whether the measure of characterisation is related to trends of correct classification. For each individual predicted class, we rank the individual test points (from three-fold cross-validation) by the measure of characterisation, e.g., the lowest value of typicality (or probability) is ranked 1. The statistic, W_s is the sum of ranks of all test points that have correct classifications. The distribution of W_s approximates a normal distribution. We can then calculate z from the mean and the standard error of a normal distribution, i.e., $z = (W_s - mean)/(standard\ error)$. We consider a significance level of 90% ($|z| \geq 1.28$) with a one-tailed test. The results for the test are tabulated in Table 1. A positive value for z indicates that the classifications are more likely to be correct with higher values of typicality/probability. A negative value for z indicates a reverse trend. The insignificant trends are marked with †.

[3] When there are ties, they are resolved randomly.

Table 1: The results of a Wilcoxon rank-sum test for trends of correct classification with respect to the measure of characterisation depicted in Figure 1

	PC	*Typicality*			*Probability*		
		n_1	n_2	z	n_1	n_2	z
bcw	0	399	6	4.03	396	3	2.96
	1	214	10	1.35	217	13	2.26
diab	0	355	102	8.22	369	87	6.68
	1	142	92	5.16	157	78	3.98
wave	0	75	25	2.08	52	6	1.97
	1	53	21	† -0.91	59	27	2.56
	2	78	18	3.05	93	33	6.22
heart	0	116	27	2.12	127	27	3.68
	1	99	30	6.48	99	19	4.13
glass	0	53	20	† 0.80	56	29	2.50
	1	50	17	2.85	47	21	3.80
	2	4	6	† 1.07	1	3	1.34
	3	5	3	1.34	1	0	–
	4	5	2	† 0.00	5	2	1.94
	5	22	5	2.87	23	4	2.73
hypo	0	111	33	4.68	123	27	4.56
	1	2673	29	7.35	2679	17	6.17
hepa	0	15	13	2.28	17	8	† 0.12
	1	98	13	1.87	103	11	3.19
auto	0	16	9	† 0.40	14	21	1.95
	1	44	10	-1.56	36	18	† -1.01
	2	41	19	1.85	31	13	† 0.22
	3	17	3	† 0.26	11	15	† 0.54
	4	21	3	† -0.83	20	5	1.43
echo	0	64	24	2.76	68	21	2.64
	1	13	16	1.45	16	12	† -0.05
horse	0	180	31	2.30	177	32	1.54
	1	89	31	4.39	88	34	4.63
soyb	0	17	0	–	17	1	1.64
	1	17	0	–	17	1	1.64
	2	19	0	–	19	14	4.66
	3	78	0	–	60	1	1.70
	7	79	6	1.89	69	10	4.39
	8	19	1	1.65	19	2	1.32
	11	39	0	–	39	5	3.61
	12	18	4	† 0.17	15	4	2.10
	13	70	15	† 0.74	80	26	3.99
	14	65	12	4.65	58	4	3.09
	18	8	4	† 0.51	8	0	–

PC : predicted class.
n_1/n_2: number of correct/incorrect classifications.
† indicates $|z| < 1.28$ (i.e., an insignificant trend).

We summarise the findings as follows.

For hypothesis **H1**, all classes in eight domains are found to conform to the hypothesis that the prediction made with high *posterior probability* is more likely to be correct. However, insignificant trends appear in one class in each of the hepatitis and echocardiogram domains, and three classes in the automobile domain.

For hypothesis **H2**, all classes in seven domains provide consistent evidence to the hypothesis that the prediction made with a *typical* nearest neighbour (i.e., with high value of typicality) is more likely to be correct. However, the trend seem to be the opposite in one class in the automobile domain. Insignificant trends appear in one class in the waveform domain, and in three classes in each of the automobile, glass and soybean domains.

The overall result is that most classes in the eleven domains provide consistent evidence to the hypotheses. The *typicality* and the *posterior probability* have been shown to be a satisfactory characterisation of predictive accuracy in IB1-MVDM* and NB*, respectively.

2.2 DISCUSSION

There are a number of reasons for these methods of characterisation to show insignificant trends in some classes in some domains. First, when the underlying concept is hard to learn for these learning algorithms we do not expect any methods of characterisation to provide a good accuracy predictor. Put it in another way, the models or assumptions of these learning algorithms could be all wrong in these cases. An example is the parity concept. The second reason could be attributed to the nature of the characterisation itself. *Typicality* measures the global property of instances of a category. In some cases, local properties in the description space might also be important in classification; exceptions or rare cases and the parity problem are examples where *typicality* would fail to characterise. Another important reason is the estimation method used to generate the graphs. Both the bin size and the "moving window" method have an effect on the shape of the graphs. Sparse data in some classes in some domains further compounds the problem.

The predictive accuracy estimation described here can be viewed as a means of calibrating an algorithm's estimate (i.e., typicality and posterior probability) with its 'true' predictive accuracy. Cross-validation is used here as a tool in performing calibration. In statistics, the idea of calibration is presented as a way of realigning one's estimates with reality or some standard (e.g., Dawid, 1982; Dunn, 1989; Efron & Tibshirani, 1993).

For Instance-Based classifiers, there are other choices with regard to the method of characterisation. For example, one may intend to use some method of kernel density estimation (Silverman, 1986), which may solve the problem faced by using *typicality* mentioned above. We have attempted another method of characterisation that uses the *nearest neighbour distance*. However, this method does not perform well. Noise is the main factor that would obscure the *nearest neighbour distance* as a good accuracy predictor in IB1-MVDM*.

Note that using these methods of characterisation, one does not explicitly know the exact regions in the description space where the predictive accuracy is high or low, as in the case of small disjuncts in decision trees or rules. But this is the result of the underlying representation rather than the methods of characterisation. The decision boundaries of IBL and Naive Bayes are intrinsically implicit in their representations. One does not know the exact decision boundaries unless one takes extra effort to draw them out (e.g., into a Voronoi diagram (Watson, 1981) for IBL).

While the methods of characterisation and estimation are not perfect for one reason or another, it suffices for our purpose here to show that the methods can be used to characterise and estimate predictive accuracy in most of the real-world domains studied. The real test is how effective are these methods in overcoming the problem of locally low predictive accuracy in the two learning algorithms, which is the topic of the next section.

3 THE METHOD OF COMBINING IB1-MVDM* AND NB*

Having characterised and estimated predictive accuracy in IB1-MVDM* and NB*, we can now make use of their individual predictive accuracy estimation for each classification for the purpose of decision combination. The combining strategy that we use is to select an estimated better performing classifier to do the final prediction and we name the resultant classifier the composite learner (CL). Because only high performance predictions are selected, the CL's performance is expected to be better than the individual algorithms. Employing the characterisations, combining the two algorithms using this strategy is straightforward. Both IB1-MVDM* and NB* are first trained independently. During training, the algorithms perform three-fold cross-validation to estimate predictive accuracy of each prediction from the characterisations. This is done by using the test results from the three-fold cross-validation to produce a binned graph that relates the average value of the characterisation to its binned predictive accuracy (as described in the last section). For each classification, the algorithm which has the higher estimated predictive accuracy (the predictive accuracy is obtained by referring to the binned graph with the corresponding value of the characterisation) is chosen to make the final prediction.

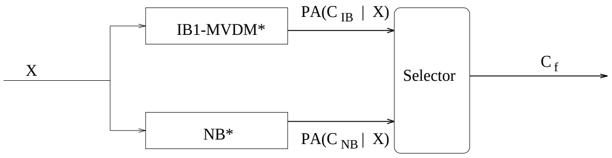

Figure 4: CL's classification process

Figure 4 depicts the classification process. Formally, the selection process can be defined as follows.

Table 2: Average error rates of BETTERof2, CL and BESTof3

	*IB**	*NB**	*BETTERof2*	*CL*	*BESTof3*
bcw	4.6	2.9	2.9	2.8	2.9
diab	28.8	25.1	25.2	⊕ 24.1	24.3
wave	21.7	21.8	21.6	⊕ 18.4	⊕ 18.4
heart	19.4	18.5	18.5	17.7	18.6
glass	27.7	29.9	28.5	26.6	28.4
hypo	1.7	1.5	1.6	⊕ 1.2	⊕ 1.2
hepa	20.6	14.9	14.6	15.5	14.9
auto	14.8	31.0	14.8	⊖ 18.9	14.8
echo	36.0	28.1	28.9	29.3	29.3
horse	20.0	19.9	20.4	⊕ 16.9	⊕ 17.2
soyb	5.8	7.7	5.8	⊕ 5.0	5.6

⊕,⊖ : two-tailed, paired t-test with the better of IB1-MVDM* (indicated as IB*) and NB*; $p<0.05$.

$$\begin{aligned} C_f &= C_{IB} \text{ if } \mathrm{PA}(C_{IB}|X) > \mathrm{PA}(C_{NB}|X), \\ C_f &= C_{NB} \text{ if } \mathrm{PA}(C_{IB}|X) < \mathrm{PA}(C_{NB}|X), \\ &\text{else random select.} \end{aligned}$$

where C_f : final prediction;
C_{IB} : IB1-MVDM*'s prediction;
C_{NB} : NB*'s prediction;
$\mathrm{PA}(C|X)$: predictive accuracy of prediction C given instance X.

3.1 PERFORMANCE OF THE COMPOSITE LEARNER

We conduct experiments in the eleven real-world domains to compare the performance of the composite learner and its constituent algorithms. CL combines IB1-MVDM* which uses typicality and NB* which uses posterior probability.

We randomly split the dataset into 90% and 10% disjoint subsets. Each experimental trial uses the 90% subset as a training set and the 10% subset as a testing set, and it is repeated over 50 trials. Table 2 shows the average classification error rates of IB1-MVDM*, NB*, BETTERof2, CL and BESTof3 – the best algorithm selected among IB1-MVDM*, NB* and CL based on the test results of three-fold cross-validation on training data. BETTERof2 employs the same method to select the better of IB1-MVDM* and NB*. The ⊕ (or ⊖) symbol in front of a figure indicates that it is significantly better (or worse) than the better of IB1-MVDM* and NB*.

CL achieves significantly better performance in five domains (pima diabetes, waveform, hypothyroid, horse colic and soybean) but significantly worse performance in the automobile domain. Note that this is the domain that both IB1-MVDM* and NB* show poor characterisations in three out of five classes (see Table 1). BESTof3 achieves significantly better results in three domains and has no significantly worse results.

In domains where the performance of IB1-MVDM* and NB* differ substantially (i.e., by more than 5%), CL performs in between the performance of its constituent algorithms but never performs better than the better of the two algorithms. Such domains are the hepatitis, automobile and echocardiogram domains. The other eight domains have small differences in terms of performance between IB1-MVDM* and NB*. In all of these domains (bcw, pima diabetes, waveform, heart, glass, hypothyroid, horse colic and soybean), CL achieves better results than the better of its constituent algorithms.

BETTERof2 performs close to selecting the better of IB1-MVDM* and NB*. While this is a strong point of the method, it also demonstrates that this method cannot produce better results than the better primitive algorithm. CL achieves significantly better results than BETTERof2 in five domains and significantly worse in one domain.

To gain insight into the factors that affect the performance of the composite learner, we examine (i) the classification overlaps made by the constituent algorithms and (ii) the correct classifications made by only one of them. Table 3 shows a complete breakdown of average correct classifications in the pima diabetes domain and Table 4 summarises the results in the eleven domains. Figures in the last row of Table 3 are computed using Equations (2) and (3). All figures in Table 4 are calculated using Equations (4)-(7) which correspond to the last four columns in the table. All equations are as follows.

$$CC[Total] = CC[0] + CC[1] + CC[2] \quad (2)$$

$$CC_{CL}[Total] = CC_{CL}[1] + CC_{CL}[2] \quad (3)$$

$$\%\ \text{Overlap} = \frac{CC[0] + CC[2]}{CC[Total]} \times 100 \quad (4)$$

$$\text{R} = \frac{CC[1]_{IB1-MVDM*}}{CC[1]} \times 100 \quad (5)$$

$$\text{S} = \frac{CC[1]_{NB*}}{CC[1]} \times 100 \quad (6)$$

$$W = \frac{CC_{CL}[1]}{CC[1]} \times 100 \quad (7)$$

where $CC[N]$ is the average number of correct classifications made by N algorithms. $CC_{CL}[N]$ is the average number of correct classifications out of $CC[N]$ made by the composite learner. $CC[1]_{AL}$ is the portion of $CC[1]$ for which only algorithm AL makes the correct classification. R (S) denotes the percentage of IB1-MVDM* (NB*) only correct classifications. W denotes the percentage of CL correct classifications from R+S.

CL improves performance if it makes the correct selection most of the time when only one of its constituent algorithms makes the correct classification; thus, $CC_{CL}[1] \leq CC[1]$. CL can do nothing when both algorithms make incorrect or correct classifications. $CC[0]$ is just the average number of classifications in which both algorithms are incorrect. Since CL always makes incorrect classifications in this portion of predictions, whichever algorithm is selected, $CC_{CL}[0]$ must be zero. Equation (4) gives the percentage of classification overlap between IB1-MVDM* and NB* over all classifications. The proportions of either IB1-MVDM* and NB* making correct classification and CL making the right choice over $CC[1]$ are given in Equations (5)-(7). The last four equations are used to tabulate the results in Table 4. The results reveal that the degree of classification overlap does not affect the performance of CL. Summing the average percentage of CL correct classification over eleven domains in Table 4 (the last column), the mean is 68.5% (an oracle would achieve 100% and a random choice is 50%). This suggests that *in all seven domains in which either IB1-MVDM* or NB* covers less than 68% of the total only one algorithm correct classification, CL achieves better performance than either of its constituent algorithms in our experiments.* The ratio of the average error rate for CL to that obtained from the better constituent algorithm measures the extent of the benefit (values less than one) associated with using the former. In these domains, the values of this ratio range from 0.97 (bcw) to 0.8 (hypothyroid), the average being 0.90.

Table 3: Average #correct classifications in the pima diabetes domain

N	$CC[N]$ (%)	*#CL correct classifications* $CC_{CL}[N]$ (%)
0	12.3 (16.0%)	0.0 (0.0%)
1	16.9 (21.9%)	10.6 (13.8%)
2	47.8 (62.1%)	47.8 (62.1%)
Total	77.0 (100.0%)	58.4 (75.9%)

N: # algorithms making correct classifications.

Table 4: Average percentage of classification overlap and correct classification

	% Overlap (#test inst.)	*R*	*S*	*% CL Correct from R+S, W*
bcw	97.4 (70)	16.7	83.3	**86.7**
diab	78.1 (77)	41.6	58.4	**63.1**
wave	79.3 (30)	50.2	49.8	**66.2**
heart	84.1 (31)	47.2	52.8	**57.3**
glass	77.8 (22)	55.7	44.3	**61.3**
hypo	98.6 (317)	45.3	54.7	**74.7**
hepa	85.0 (16)	30.8	**69.2**	65.0
auto	72.5 (21)	**84.9**	15.1	67.3
echo	83.3 (14)	26.5	**73.5**	66.7
horse	86.1 (37)	49.6	50.4	**72.5**
soyb	92.1 (69)	62.5	37.5	**72.8**

Boldface indicates the maximum value in the last three columns for each domain.

3.2 DISCUSSION

Because the composite learner employs a strategy that probes into the intrinsic problem of the induced theories of its constituent algorithms and uses only those predictions that are more likely to be correct, it is able to improve the performance of the individual algorithms. Using a cross-validation algorithm such as BETTERof2 does not get better performance than the better primitive learning algorithm. Cross-validation uses global information (i.e., the overall estimated error rate) to do model selection *before* classification, whereas CL uses local information (i.e., the characterisation of predictive accuracy of each prediction) to select which model to use *during* classification.

The fact that CL achieves only 68% correct on average (in the portion where only one of IB1-MVDM* and NB* has correct classification) explains why the current implementation of the composite learner can only achieve better results in domains where the constituent algorithms have comparable performance. This indicates that there is still room for further improvement. First, the methods for characterising predictive accuracy can be improved. Some of the problems with *typicality* have been discussed in Section 2.2. Use of an exponential distance function rather than Euclidean distance function in Equation (1), or even a kernel density estimation (Silverman, 1986) may improve the characterisation. Second, the predictive accuracy estimation method of producing binned graphs can be improved. Current weaknesses can be mitigated by employing cross-validation to select the best algorithm among the composite learner and its constituent algorithms. This method, BESTof3 is very effective in domains such as hepatitis and automobile where the performance between IB1-MVDM* and NB* differs substantially. Third, in two-class domains, an improvement can possibly be made by simply reverse the prediction when the binned graph for a class is below 50% accuracy (see the bottom left portions of the binned graphs for typicality in Figures 2 and 3).

There are two situations when one algorithm performs comparably to the other. The first situation is when the algorithms are highly correlated; in this case, the classification overlap would be very high. In the second situation, as the classification overlap is relatively low, the proportion classified correctly by one algorithm while the other algorithm is incorrect is about the same for both algorithms. Nevertheless, CL is capable of performing well under both situations. Classification correlation between the constituent algorithms in some domains does not affect the CL's performance. This is shown in the bcw and hypothyroid domains where the classification overlap is over 97%.

The composite learner can be viewed as an "informed" simple generaliser that makes its decision to select a classifier based on the weight of the prediction. The weight of the prediction can be regarded as the classifier's "confidence" concerning the prediction made.

The proposed composite learner can be easily extended to include other learning algorithms, provided appropriate characterisations of those induced theories can be found. For example, the probabilities of a neural network's outputs could be used as the characterisation. In such cases, a scheme that combines the weights/evidences of the predictions might need to be incorporated into the system.

A separate empirical evaluation shows that CL outperforms a scheme of stacked generalisation (Wolpert, 1992), using either of the algorithms described as the high level generaliser, in most of the domains studied here. The results are not shown here due to a lack of space (see Ting (1995) for details).

4 RELATED WORK

The most closely related work that exploits local information in combining different learning algorithms is Ting's (1994a) treatment of the problem of small disjuncts in decision trees. The key idea of this work is to use Instance-Based methods to solve the problem of small disjuncts in decision trees by replacing the small disjunct regions with Instance-Based methods. In that framework, decision trees only work in large disjunct regions and Instance-Based methods only work in small disjunct regions. This arrangement is based on the assumption that decision trees perform better than Instance-Based methods in the regions of large disjuncts. In domains where this assumption does not hold, the performance of this composite learner will be worse than Instance-Based methods. KBNGE (Wettschereck, 1994) in many respects resemble Ting's work without using an explicit characterisation of the problem of small disjuncts. It uses BNGE (i.e., a nested generalised exemplars algorithm (Salzberg, 1991) in regions that clearly belong to one class and a k-nearest neighbour algorithm otherwise.

In an independent work, Merz (1995) describes a method of stacked generalisation where selection of algorithms can be done during classification. The DS (dynamic selection) algorithm builds two m by n matrices during training, where m is the number of training instances and n is the number of learning algorithms. The first matrix records the predictions of the induced theories and the second matrix records the test results of w v-fold cross-validations for each training instance. All induced theories are used in classifying a new instance to construct a row-vector that can be compared in the first matrix to identify the rows "closest" to the row-vector of the new instance. The accuracies for each induced theory are computed from the corresponding rows (by averaging) in the second matrix. The induced theory having the highest accuracy is selected for final prediction.

MCS (Brodley, 1993) uses a hand-crafted rule to guide the construction of three different models at the nodes of a decision tree, and each model is trained on part of the training set. In contrast, the two algorithms in the composite learner are trained independently on the total training data and work cooperatively according to the characterisation of predictive accuracy in each learned theory. Thus, no hand-crafted knowledge is used in the composite learner. CL selects a model during classification; MCS fixes the various models at the nodes of a decision tree during training.

Methods of selecting a learning algorithm for a given domain (e.g., Schaffer, 1993; Aha, 1992; Breiman et al 1984) using the entire set of training data only choose the best performing algorithm, at best. Some other methods split the data into (i) mutually exclusive subsets (Tcheng et al, 1989; Utgoff, 1989; Chan & Stolfo, 1995) or (ii) resampling (with replacement) subsets (Drucker, Schapire & Simad, 1993). Different classifiers are trained using these subsets. Several methods of re-ordering ranks when combining multiple models have been proposed in the literature. Buntine (1991) introduced an algorithm design strategy based on the approximating Bayesian decision theory of learning class probability decision trees. The class ranking was re-ordered after averaging the class probabilities from the multiple trees. Perrone and Cooper's (1993) ensemble methods work in a similar manner by merely (weighted) averaging the corresponding outputs of the multiple neural networks. Smyth, Goodman & Higgins (1990) and Kononenko & Kovačič (1992) use Naive Bayesian combination of decisions of several different rules in re-ordering ranks. Ho, Hull and Srihari (1994) employ logistic regression to re-rank classes across different types of classifiers, and only the ranks of the classes in the predictions are considered.

5 CONCLUSION

The main contribution of this paper is two-fold. First, we have characterised an intrinsic problem, i.e., the problem of locally low predictive accuracy in IBL and Naive Bayes. Two intuitively sound methods, the *typicality* and the *posterior probability* have been found to be a satisfactory characterisation of predictive accuracy in IB1-MVDM* and NB*, respectively. Second, knowing the weaknesses in each algorithm, the simple strategy of selecting as the final prediction the one that has the higher estimated predictive accuracy has been demonstrated to partially overcome the problem of locally low predictive accuracy. When IB1-MVDM* and NB* demonstrate comparable performance, we strongly recommend that the proposed composite learner is used to improve the overall performance. In the same circumstances, the composite learner is found to outperform a cross-validation method for model selection, BETTERof2. BESTof3 incorporating the composite learner achieves results better than or comparable to the better performance of the composite learner's constituent learning algorithms in all the experimental domains.

Acknowledgements

This research was done when this author was in Basser Department of Computer Science, The University of Sydney, Australia. It was partially supported by an Australia Research Council grant (to J.R.Quinlan) and by a research agreement with Digital Equipment Corporation. Numerous discussions with J.R. Quinlan, R.M. Cameron-Jones, Z. Zheng and P. Langley have been very helpful. Thanks to D.W. Aha for providing IB1 algorithm. This author was partially supported by the Equity and Merit Scholarship Scheme. Comments from anonymous referees are very helpful.

References

Aha, D.W. (1992), Generalizing from case studies: A Case Study, in *Proceedings of the Ninth International Conference on Machine Learning*, pp. 1-10. Morgan-Kaufmann.

Aha, D.W., D. Kibler & M.K. Albert (1991), Instance-Based Learning Algorithms, *Machine Learning, 6*, pp. 37-66.

Breiman, L., J.H. Friedman, R.A. Olshen & C.J. Stone (1984), *Classification And Regression Trees*, Belmont, CA: Wadsworth.

Brodley, C.E. (1993), Addressing the Selective Superiority Problem: Automatic Algorithm/Model Class Selection, in *Proceedings of the Tenth International Conference on Machine Learning*, pp. 17-24.

Buntine, W. (1991), Classifiers: A Theoretical and Empirical Study, in *Proceedings of the Twelfth International Joint Conference on Artificial Intelligence*, pp. 638-644, Morgan-Kaufmann.

Cestnik, B. (1990), Estimating Probabilities: A Crucial Task in Machine Learning, in *Proceedings of the European Conference on Artificial Intelligence*, pp. 147-149.

Chan, P.K. & S.J. Stolfo (1995), A Comparative Evaluation of Voting and Meta-learning on Partitioned Data, in *Proceedings of the Twelfth International Conference on Machine Learning*, pp. 90-98, Morgan Kaufmann.

Cost, S & S. Salzberg (1993), A Weighted Nearest Neighbor Algorithm for Learning with Symbolic Features, *Machine Learning, 10*, pp. 57-78.

Dawid, A.P. (1982), The Well-Calibrated Bayesian, *Journal of the American Statistical Association*, Vol. 77, No. 379, pp. 605-610.

Drucker, H., R. Schapire & P. Simad (1993), Improving the performance in neural networks using a boosting algorithm, *Advances in Neural Information Processing Systems*, 5, pp. 42-49.

Dunn, G. (1989), *Design and Analysis of Reliability Studies*, Oxford University Press.

Efron, B. & R.J. Tibshirani (1993), Chapter 18 in *An Introduction to the Bootstrap*, Chapman & Hall.

Fayyad, U.M. & Irani, K.B. (1993), Multi-Interval Discretization of Continuous-Valued Attributes for Classification Learning, in *Proceedings of 13th International Joint Conference on Artificial Intelligence*, pp. 1022-1027.

Ho, T.K., J.J. Hull & S.N. Srihari (1994), Decision Combination in Multiple Classifier Systems, in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.16, no.1, pp. 66-75.

Holte, R.C., L.E. Acker & B.W. Porter (1989), Concept Learning and the Problem of Small Disjuncts, in *Proceedings of the 11th International Joint Conference on Artificial Intelligence*, pp. 813-818.

Howell, D.C. (1982), *Statistical Methods for Psychology.* PWS Publishers.

Kononenko, I. & M. Kovačič (1992), Learning as Optimization: Stochastic Generation of Multiple Knowledge, in *Proceedings of the Ninth International Conference on Machine Learning*, pp. 257-262, Morgan Kaufmann.

Merz, C.J. (1995), Dynamic Learning Bias Selection, in *Proceedings of the Fifth International Workshop on Artificial Intelligence and Statistics*, Ft. Lauderdale, FL: Unpublished, pp. 386-395.

Murphy, P.M. & D.W. Aha (1994), *UCI Repository of machine learning databases* [http://www.ics.uci.edu/ mlearn/MLRepository.html]. Irvine, CA: University of California, Department of Information and Computer Science.

Perrone, M.P. & L.N. Cooper (1993), When Networks Disagree: Ensemble Methods for Hybrid Neural Networks, in *Artificial Neural Networks for Speech and Vision*, R.J. Mammone (editor), Chapman-Hall.

Rosch, E. & C.B. Mervis (1975), Family Resemblances: Studies in the Internal Structures of Categories, *Cognitive Psychology*, vol. 7, pp. 573-605, Academic Press.

Salzberg, S. (1991), A Nearest Hyperrectangle Learning Method, *Machine Learning*, 6, pp. 251-276.

Schaffer, C. (1993), Selecting a Classification Method by Cross-validation. *Preliminary Papers of the Fourth International Workshop on Artificial Intelligence and Statistics*, pp. 15-25.

Silverman, B.W. (1986), *Density Estimation for Statistics and Data Analysis*, Chapman and Hall.

Smyth, P., R.M. Goodman & C. Higgins (1990), A Hybrid Rule-based/ Bayesian Classifier, in *Proceedings of the Ninth European Conference on Artificial Intelligence*, pp. 610-615.

Tcheng, D., B. Lambert, C-Y. Lu & L. Rendell (1989), Building Robust Learning Systems by Combining Induction and Optimization, in *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence*, pp. 806-812, Morgan Kaufmann.

Ting, K.M. (1994a), The Problem of Small Disjuncts: its remedy in Decision Trees, in *Proceedings of the Tenth Canadian Conference on Artificial Intelligence*, pp. 91-97.

Ting, K.M. (1994b), The Problem of Atypicality in Instance-Based Learning, in *Proceedings of the Third Pacific Rim International Conference on Artificial Intelligence*, pp. 360-366, International Academic.

Ting, K.M. (1994c), *Discretization of Continuous-Valued Attributes and Instance-Based Learning*, Technical Report No.491, Basser Department of Computer Science, University of Sydney.

Ting, K.M. (1995), *Common Issues in Instance-Based and Naive Bayesian Classifiers*, PhD Thesis, Basser Dept of Computer Science, University of Sydney.

Ting, K.M. (1996), Discretisation in Lazy Learning Algorithms to appear in the special issue of Lazy Learning in *Journal of Artificial Intelligence Review.*

Utgoff, P.E. (1989), Perceptron Trees: A case study in hybrid concept representations, *Connection Science*, 1, pp. 337-391.

Watson, D.F. (1981), Computing the *n*-dimensional Delaunay Tessellation with application to Voronoi Polytopes, *The Computer Journal*, vol.24, no.2, pp. 167-172, Heyden & Son Ltd.

Wettschereck, D. (1994), A Hybrid Nearest-Neighbor and Nearest-Hyperrectangle Algorithm, in *Proceedings of the Seventh European Conference on Machine Learning, LNAI-784*, pp. 323-335, Springer Verlag.

Wolpert, D.H. (1992), Stacked Generalization, *Neural Networks*, vol.5, pp. 241-259, Pergamon Press.

Zhang, J. (1992), Selecting Typical Instances in Instance-Based Learning, *Proceedings of the Ninth International Conference on Machine Learning*, pp. 470-479, Morgan Kaufmann.

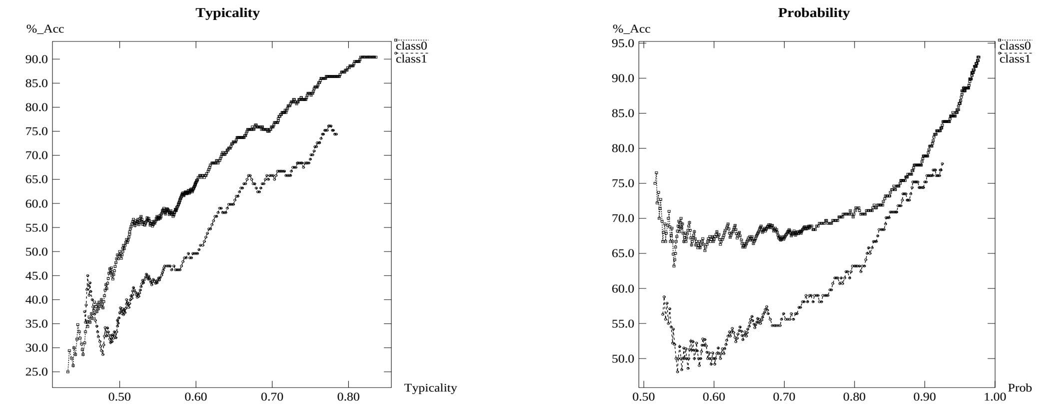

Figure 2: Binned graphs for typicality and probability in the Pima Diabetes domain

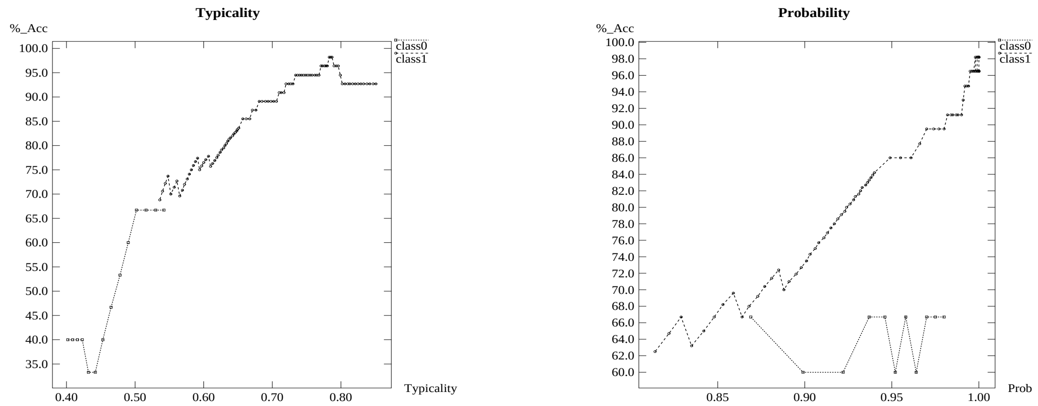

Figure 3: Binned graphs for typicality and probability in the Hepatitis domain