Constructing Reliable Classifiers for Road Side Assistance

Evgueni N. Smirnov SMIRNOV@MICC.UNIMAAS.NL
Mark H.M. Winands M.WINANDS@MICC.UNIMAAS.NL
Pieter H.M. Spronck P.SPRONCK@MICC.UNIMAAS.NL
Maarten P.D. Schadd MAARTEN.SCHADD@MICC.UNIMAAS.NL
Universiteit Maastricht, MICC, P.O. Box 616, 6200 MD Maastricht, The Netherlands

Abstract

This paper considers the problem of constructing classifiers for road side assistance capable of providing reliability values for classifications of individual instances. In this context we analyze the existing approaches to reliable classification based on the typicalness framework (Vovk et al., 2005; Smirnov & Kaptein, 2006). As a result we propose an approach that allows the framework to be applied to any type of classifiers so that the classification-reliability values can be computed for each class. The experiments show that the approach outperforms the existing approaches to reliable classification for road side assistance.

1. Introduction

In the last ten years machine-learning classifiers have been applied for many classification problems (Perner, 2006). Nevertheless, only few classifiers have been employed in critical-domain applications (Smirnov & Kaptein, 2006). This is due to the difficulty to determine whether a classification assigned to a particular instance is reliable or not.

The importance of the reliable-classification problem can be demonstrated in the context of the on-going EU project MYCAREVENT [1]. One of the goals of the project is to implement *a road side assistance decision support system* capable of providing manufacturer-specific car-repair information according to the problems identified by cars's Off-/On-Board-Diagnosis systems. One of the core parts of the system is a machine-learning classifier for road side assistance. The classifier has to predict the status of a car that experiences problems. Due to substantial costs involved, in addition to good generalization performance, the classifier has to provide reliability values for each possible classification of a car problem. In this way a system user can estimate the risks of actions s/he takes for each possible classification (car-problem status).

In this paper we present our first attempt to construct machine-learning classifiers for road side assistance that are capable of providing classification-reliability values. We start in Section 2 where we formalize the classification task in the context of reliable classification. In Section 3 we briefly describe the existing approaches to reliable classification that we find useful for our problem, namely the typicalness framework (Proedru et al., 2002; Saunders et al., 1999; Vovk et al., 2005) and the meta-classifier typicalness approach (Smirnov & Kaptein, 2006; Smirnov et al., 2006b) [2]. After analyzing both approaches we propose in Section 4 a novel approach to reliable classification that we call single-stacking typicalness approach. The performance of the three approaches is compared in the context of our problem in Section 5. The comparison shows that the single-stacking typicalness approach outperforms the typicalness framework and the meta-classifier typicalness approach for road side assistance. Finally, Section 6 concludes the paper.

2. Classification Task

Let $\mathbf{X}$ be an instance space and Y a class set. Training data D is a set $\{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_n, y_n)\}$ of n labelled instances $(\mathbf{x}_i, y_i)$ where instance $\mathbf{x}_i$ is in $\mathbf{X}$ and class y_i is in Y. Given a space H of classifiers h ($h : \mathbf{X} \rightarrow Y$), the classification task is to find classifier $h \in H$ that correctly predicts future, unseen instances.

[1] MYCAREVENT (Mobility and Collaborative Work in European Vehicle Emergency Networks) is a 3-year research project financed by the IST (Information Society Technology) program of the European Commission within the Sixth Framework Program. For more information, please visit http://www.mycarevent.com/.

[2] Although version space support vector machines are a successful approach to reliable classification (Smirnov et al., 2006a), they are not considered in the paper since they do not provide classification-reliability values.

Proceedings of the 18th Benelearn
P. Adriaans, M. van Someren, S. Katrenko (eds.)

If a classifier h has to be used for reliable classification, we need to solve additional two tasks (Baskiotis & Sebag, 2004; Duda et al., 2000; Hullermeier, 2004; Proedru et al., 2002; Saunders et al., 1999; Vovk et al., 2005):

- to obtain reliability values for classification of individual instances; and
- to obtain a threshold on these values.

If the reliability value for a classification of an instance provided by the classifier h is greater than the threshold, the classification is considered to be reliable. Otherwise, it is unreliable and the instance is left unclassified.

3. Existing Approaches to Reliable Classification

When we have to construct a classifier capable of providing reliability values for individual instance classifications, we can employ either the typicalness framework (Vovk et al., 2005) or the meta-classifier typicalness approach (Smirnov & Kaptein, 2006). Both approaches are briefly described in this section.

3.1. The Typicalness Framework

The typicalness framework was proposed in (Proedru et al., 2002; Saunders et al., 1999; Vovk et al., 2005) for constructing classifiers for reliable classification. The framework assumes that the data D with n training instances and an instance $\mathbf{x}_{n+1}$ to be classified are drawn from the same unknown distribution. Given a classifier h, we compute the typicalness value of the instance $\mathbf{x}_{n+1}$ for each class $y_{n+1} \in Y$ that the classifier h can assign to $\mathbf{x}_{n+1}$. The typicalness value of $\mathbf{x}_{n+1}$ for a class $y_{n+1} \in Y$ is a p-value computed as follows:

$$p(y_{n+1}) = \frac{\#\{i : \alpha_i \geq \alpha_{n+1}\}}{n+1} \tag{1}$$

where α_i is the strangeness value of instance $(\mathbf{x}_i, y_i)$ in the bag $\{(\mathbf{x}_1, y_1), ..., (\mathbf{x}_{i-1}, y_{i-1}) (\mathbf{x}_{i+1}, y_{i+1}), ..., (\mathbf{x}_{n+1}, y_{n+1})\}$.

The class y_{n+1} with the largest p-value is the classification of $\mathbf{x}_{n+1}$. The credibility of this classification is the largest p-value and the confidence is one minus the second largest p-value.

To compute strangeness values α_i for each instance $(\mathbf{x}_i, y_i)$ we need to construct an instance-strangeness function α for the classifier h used. If we have access to the internal structure of classifiers, Vovk et al. (2005) showed that instance-strangeness functions can be constructed for nearest neighbor classifiers (Proedru et al., 2002), decision trees, neural networks, support vector machines (Saunders et al., 1999), and the naive Bayes classifier. To provide an example, let us consider the instance-strangeness function for nearest neighbor classifiers (Proedru et al., 2002). Given an instance $\mathbf{x}_i$ labeled to belong to class y_i, the instance-strangeness function returns for $\mathbf{x}_i$ a strangeness value α_i equal to $\frac{D_k^{y_i}}{D_k^{-y_i}}$, where $D_k^{y_i}$ is the sum of distances between $\mathbf{x}_i$ and k nearest neighbors of $\mathbf{x}_i$ that belong to class y_i, and $D_k^{-y_i}$ is the sum of distances between $\mathbf{x}_i$ and k nearest neighbors of $\mathbf{x}_i$ that do not belong to class y_i.

Although instance-strangeness functions were proposed for some basic types of classifiers (see above), there is no approach how to design these functions in general. Moreover, if we do not have access to the internal structure of classifiers (e.g., the classifier is a human expert), we cannot design instance-strangeness functions at all. Therefore, we may conclude that the applicability of the typicalness framework is restricted.

3.2. Meta-Classifier Typicalness Approach

We proposed the meta-classifier typicalness approach in (Smirnov & Kaptein, 2006; Smirnov et al., 2006b) to allow the typicalness framework to be applied for any type of classifiers. Assume that we have a classification problem that requires computing p-values for instance classifications but our best classifier B is not capable of providing such values. Then, if we have any typicalness-based classifier M, the approach is to train M as a meta classifier that predicts the correctness of each instance classification of B. In this way, the p-values of the meta predictions can be viewed as estimates of the p-values of the instance classifications of the base classifier B. More precisely, if B assigns class y to instance $\mathbf{x}$, then the p-value of class y is set equal to the p-value p_0 of the meta class "correct classification" and the sum of p-values of all the remaining classes $Y \setminus \{y\}$ is set equal to the p-value p_1 of the meta class "incorrect classification". Thus, using the ensemble of B and M, denoted by *MCT(B:M)*, we can estimate the p-value (typicalness) and confidence of each classification y provided by B.

An obvious drawback of the meta-classifier typicalness approach is that when the base classifier B assigns a class y to the instance $\mathbf{x}$ we cannot estimate the p-value of each class in $Y \setminus \{y\}$. Thus, if $p_0 < p_1$, the classification with the highest p-value among classes $Y \setminus \{y\}$ cannot be identified. To overcome this drawback we propose in the next section a single-stacking typicalness approach.

4. Single-Stacking Typicalness Approach

We propose the single-stacking typicalness approach to allow the typicalness framework to be applied for any type of classifiers so that the p-values can be computed for each class involved in classification. As the name suggest the key idea behind the approach is to employ a stacking ensemble (Wolpert, 1992) consisting of one base classifier B and one meta classifier M (see Figure 1). In this way, if the classifier M is based on the typicalness framework , the p-values of the meta predictions can be considered as the p-values of the instance classifications of the base classifier B.

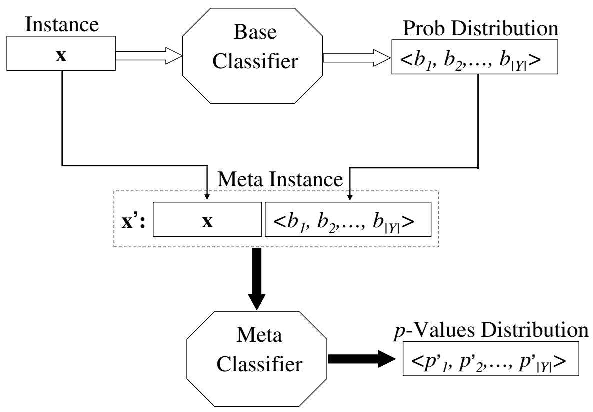

Figure 1. Single-Stacking Typicalness Approach.

The meta classifier M plays a central role in a single-stacking ensemble (see Figure 1). For each instance to be classified it has to receive classification information from the base classifier B in terms of the class probability distribution and then to compute the desired p-value for each class $y \in Y$. Therefore,

- meta instance space X' is defined by the attributes of the original instance space X plus $|Y|$ attributes representing the class probability distribution that the base classifier B computes for any instance in X;
- the meta class set Y' coincides with the class set Y.

The meta data D' are formed in $X' \times Y'$ using internal k-fold cross validation as follows: a labelled meta instance $(\mathbf{x}'_i, y_i) \in D'$ is formed from the labelled instance $(\mathbf{x}_i, y_i) \in D$ s.t. $\mathbf{x}'_i$ is a union of $\mathbf{x}_i$ and the class probability distribution computed by B for $\mathbf{x}_i$. Once meta data D' have been formed, the meta classifier M is trained on these data.

The single-stacking ensemble of the base classifier B and the meta classifier M is denoted by *SST(B:M)*. *SST(B:M)* is used for classification of an instance $\mathbf{x}$ as follows (see Figure 1). First, B classifies $\mathbf{x}$ by providing the class probability distribution $\langle b_1, b_2, ..., b_{|Y|}\rangle$ for $\mathbf{x}$. Then, the instance $\mathbf{x}$ and the distribution $\langle b_1, b_2, ..., b_{|Y|}\rangle$ are concatenated to form the meta instance $\mathbf{x}'$. The meta classifier M classifies the meta instance $\mathbf{x}'$ by providing the class probability distribution for $\mathbf{x}'$. If M is based on the typicalness framework, the class probability distribution of M is a class p-values distribution $\langle p'_1, p'_2, ..., p'_{|Y|}\rangle$ consisting of p-values p'_i for the classes in Y. In this case, the single-stacking typicalness approach approximates for the base classifier B the p-value p_i for each class $y_i \in Y$ using the p-value p'_i of the meta classifier M. This approximation rule of the p-values of B guarantees that the classification rule of the meta classifier M is preserved. Since the meta classifier M does contain information about misclassifications of B, the approximation rule can cause eventually correcting the classifications of B.

5. Road Side Assistance Classifiers

In this section we provide our experiments in constructing road side assistance classifiers using the typicalness framework, the meta-classifier typicalness approach, and the single-stacking typicalness approach. In subsection 5.1 we describe the classification task of road side assistance. Then, in subsection 5.2 we specify and experiment with the classifiers based on the three approaches to reliable classification described in the paper.

5.1. Road Side Assistance Classification Problem

In the context of the MYCAREVENT project the road side assistance classifier has to be trained on historical patrol-car data of previously diagnosed faults and their symptoms provided by RAC (a UK-based motoring organisation, originally formed by the Royal Automobile Club). The data were derived from the customer/CCO dialogue.

The data are described using four discrete attributes *Brand* (40 discrete values), *Model* (229 discrete values), *Primary Fault* (35 discrete values), *Secondary Fault* (80 discrete values), and class attribute *Status*. The class attribute have three values (classes):

1. **Fixed:** The problem is solved by road side assistance and the car can continue its journey safely (3366 instances).
2. **Required tow:** The car needs to be towed to the workshop (1077 instances).
3. **Other:** Some parts of the problem cannot be solved by road side assistance but the car is able to get to the workshop by its own (1477 instances).

5.2. Experiments

To construct road side assistance classifiers we employed three standard classifiers: C4.5 decision tree learner (*C4.5*) (Quinlan, 1993), k-nearest neighbor (*NN*) (Duda et al., 2000), and naive Bayes classifier (*NB*) (Domingos & Pazzani, 1996) as well as one typicalness-based nearest neighbor classifier (*TCMNN*) [3] (Proedru et al., 2002). *C4.5*, *NN*, and *NB* were used as independent classifiers and as base classifiers in typicalness ensembles. *TCMNN* was used as independent typicalness-based classifier and as meta classifier in typicalness ensembles. The meta-classifier typicalness approach was presented by three ensembles *MCT(C45:TCMNN)*, *MCT(NN:TCMNN)*, and *MCT(NB:TCMNN)*. The single-stacking typicalness approach was presented by three ensembles *SST(C45:TCMNN)*, *SST(NN:TCMNN)*, and *SST(NB:TCMNN)*. For the classification reliability values of *C4.5*, *NN*, and *NB* we used the classification probabilities of these classifiers. For the classification reliability values of *TCMNN* and the typicalness ensembles we used classification p-values these classifiers can generate.

We experimented with the classifiers by varying the reliability threshold r in the interval $[0, 1]$. If the reliability value for a classification of an instance was greater than r, the classification was considered as reliable; otherwise, the classification was considered as unreliable and the instance was left unclassified. For each value of the reliability threshold r we evaluated using the 10-fold cross validation:

- rejection rate: proportion of the *unclassified* instances;
- accuracy rate on the *classified* instances;
- rejection rate per class: proportion of the *unclassified* instances per class;
- true positive rate per class *TPr* (Fawcett, 2003) on the *classified* instances.

The results of our experiments are presented as accuracy/rejection and *TPr*/rejection graphs[4] (Ferri & Hernndez-Orallo, 2004). They are provided in figures 2, 3, and 4. To facilitate the comparison between the classifiers we extracted from the graphs the rejection rates for the accuracy rate of 1.0 and the *TPr* rate of 1.0 per class. These rates are presented in table 1.

An analysis of the results shows that for most of the cases the accuracy/rejection and TPr/rejection graphs of *TCMNN*, the *SST* ensembles, and the *MCT* ensembles dominate those of *C4.5*, *NN*, and *NB*. This implies the following two conclusions:

- the class probabilities of *C4.5*, *NN*, and *NB* are pure estimates of classification-reliability values. They can be used only for the majority class "Fixed"and fail bitterly for the minority classes "Required tow" and "Other"(see the last two columns of table 1).
- *SST* ensembles, *MCT* ensembles and *TCMNN* provide good classification-reliability values. They can be used for the majority class "Fixed" as well as for the minority classes "Required tow" and "Other".

Classifiers	R	R_F	R_R	R_O
NB	0.93	**0.56**	–	0.51
NN	0.98	0.90	–	0.97
C4.5	0.98	0.95	–	0.78
TCMNN	0.72	0.69	0.93	0.52
MCT(NB:TCMNN)	0.72	0.71	0.96	0.37
MCT(NN:TCMNN)	0.71	0.69	0.92	0.48
MCT(C4.5:TCMNN)	0.71	0.62	–	0.38
SST(NB:TCMNN)	0.68	0.67	0.88	**0.34**
SST(NN:TCMNN)	0.71	0.71	0.95	0.36
SST(C4.5:TCMNN)	**0.64**	0.68	**0.84**	**0.34**

Table 1. Rejection Rates for the Accuracy Rate of 1.0 and the *TPr* Rate of 1.0. R is the Rejection Rate for the Accuracy Rate of 1.0. R_F is the Rejection Rate for the *TPr* Rate of 1.0 for Class "Fixed". R_R is the Rejection Rate for the *TPr* Rate of 1.0 for Class "Required tow". R_O is the Rejection Rate for the *TPr* Rate of 1.0 for Class "Other". The Undefined Rejection Rates are Denoted by "–".

A further analysis shows that *SST* ensembles outperform *MCT* ensembles and *TCMNN* on accuracy/rejection graphs (see table 1). For the majority class "Fixed" one of the *MCT* ensembles outperforms *SST* ensembles and *TCMNN*. For the minority classes "Required tow" and "Other" two of the *SST* ensembles outperform *MCT* ensembles and *TCMNN* .

The best classifier is the *SST(C4.5:TCMNN)* ensemble. Its rejection rate for the accuracy rate of 1.0 outperforms those of *MCT* ensembles and *TCMNN* with values 0.007 and 0.08, respectively (see table 1). Although *SST(C4.5:TCMNN)* is outperformed by *MCT(C4.5:TCMNN)* on the majority class "Fixed", it provides the lowest rejection rates for the minority classes "Required tow" and "Other".

6. Conclusion and Future Research

In this paper we considered the problem of constructing machine-learning classifiers for road side assistance that are capable of providing classification-reliability values.

[3]The *TCMNN* strangeness function is given in section 3.1. *TCMNN* computes p-values according to formula 1.

[4]We note that the accuracy/rejection (*TPr*/rejection) graph of the "always-right" classifier is determined by the segment $\langle(0, 1), (1, 1)\rangle$. If two classifiers have the same accuracy rate, we prefer the classifier with lower rejection rate.

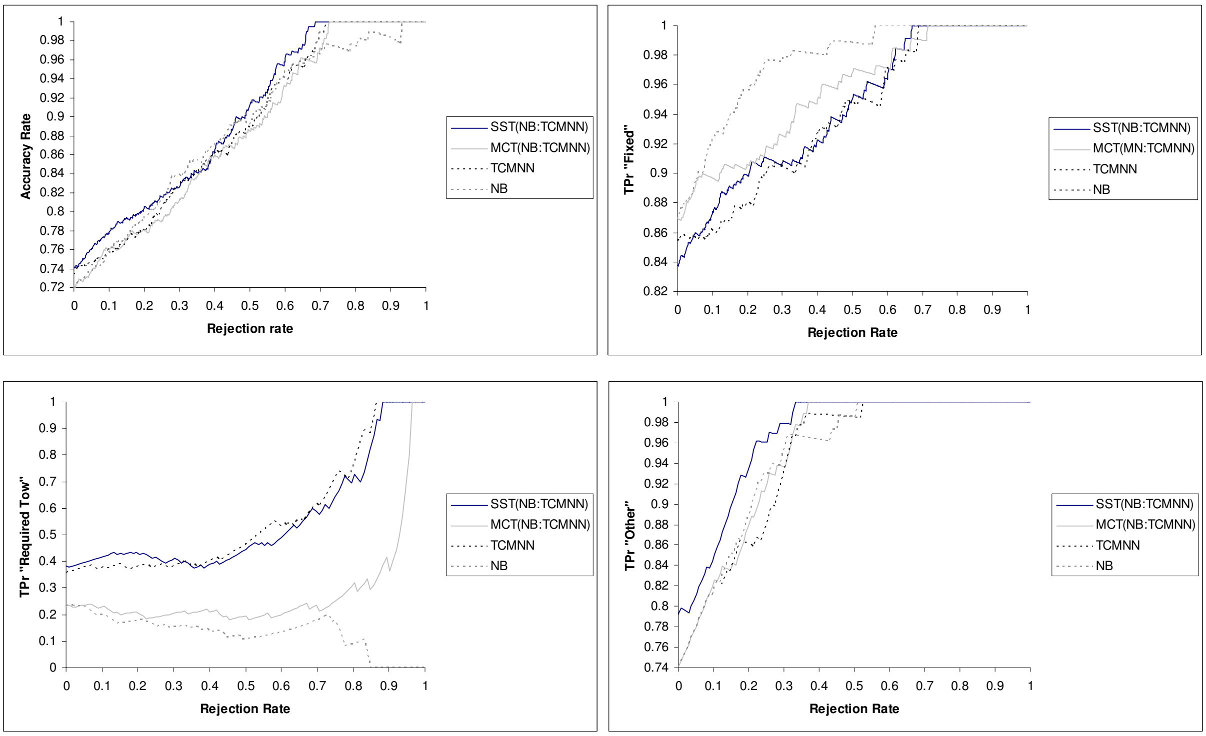

Figure 2. Accuracy/Rejection Graph and TPr/Rejection Graphs for *NB*, *TCMNN*, *SST(NB:TCMNN)*, and *MCT(NB:TCMNN)*.

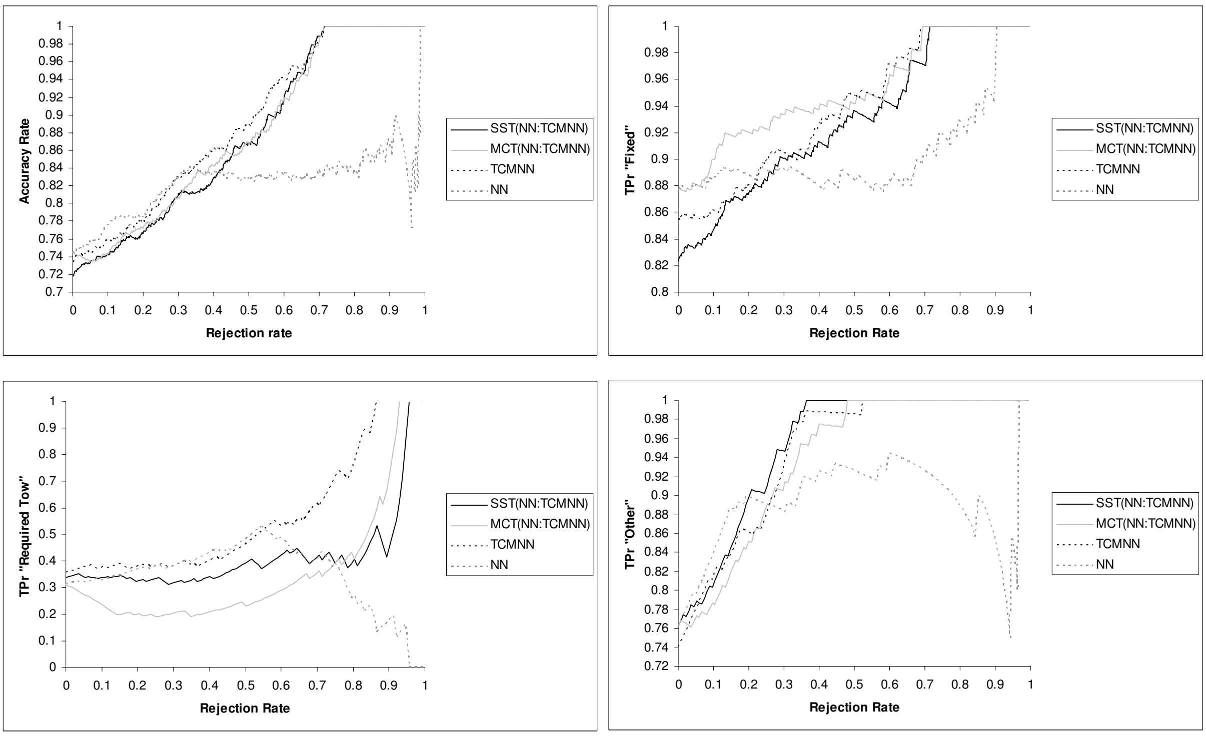

Figure 3. Accuracy/Rejection Graph and TPr/Rejection Graphs for *NN*, *TCMNN*, *SST(NN:TCMNN)*, and *MCT(NN:TCMNN)*.

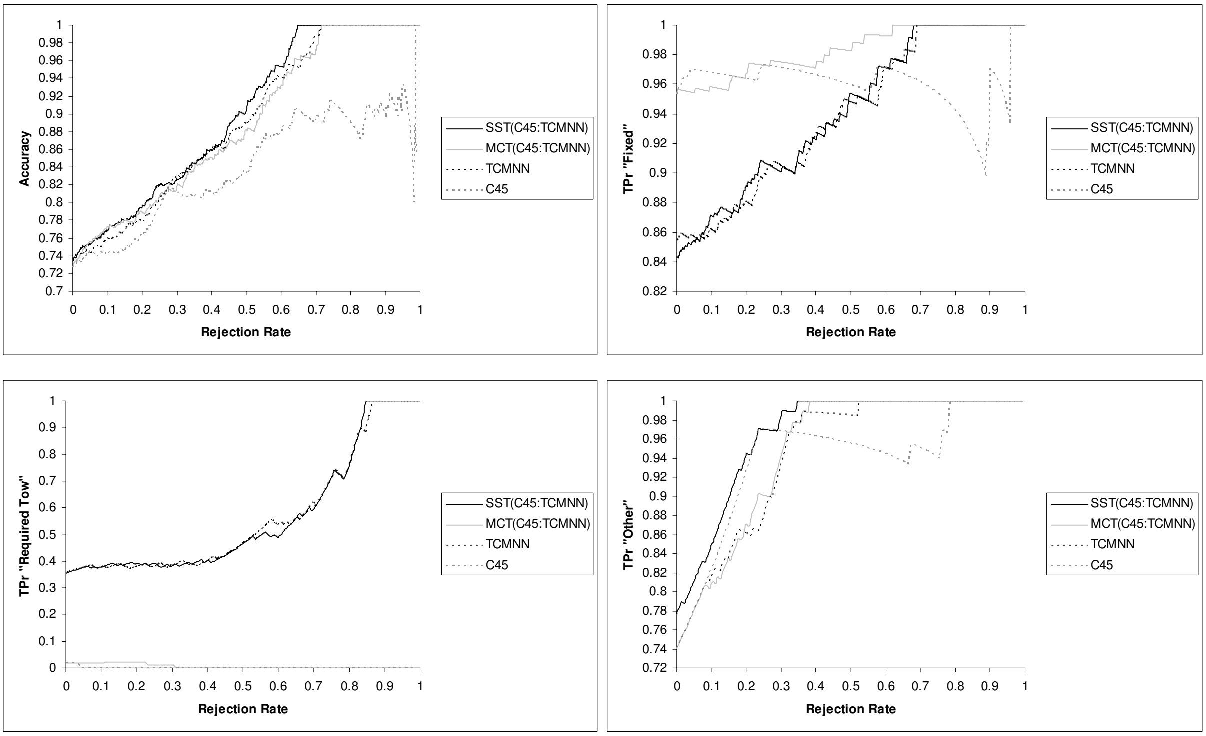

Figure 4. Accuracy/Rejection Graph and TPr/Rejection Graphs for *C4.5*, *TCMNN*, *SST(C4.5:TCMNN)*, and *MCT(C4.5:TCMNN)*.

We analyzed the typicalness framework and the meta classifier typicalness approach in the context of this problem. As a result we proposed the single-stacking typicalness approach that allows the typicalness framework to be applied to any classifier so that the classification-reliability values can be computed for each class.

The experiments show that the single-stacking typicalness approach allows constructing ensembles that are capable of outperforming a standard typicalness-based classifier (*TCMNN*) and meta classifier typicalness ensembles for road side assistance. We explain this result by correcting mechanism that the single-stacking typicalness approach employs. Another important result that follows from the experiments is that standard classifiers such as decision trees, nearest neighbor classifiers, and naive Bayes classifiers are not capable of providing good classification-reliability values. Thus, we may conclude that the typicalness framework and its accompanying meta-typicalness approaches are useful when classification-reliability values have to be plausibly estimated for practical problems like road side assistance.

Future research will focus on analysis of the single-stacking typicalness approach, especially on conditions when the approach can be successfully applied. The analysis can be used for extending the approach to a multi-stacking mechanism for assigning classification-reliability values.

Acknowledgments

We would like to thank Peter Skilton for providing the data set. Moreover, we thank the anonymous referees for their useful remarks.

References

Baskiotis, N., & Sebag, M. (2004). C4.5 competence map: a phase transition-inspired approach. *Proceedings of the 21-th International Conference on Machine Learning (ICML-2004)* (pp. 73–80).

Domingos, P., & Pazzani, M. (1996). Beyond independence: conditions for the optimality of the simple bayesian classifier. *Proceedings of the Thirteenth International Conference on Machine Learning (ICML-96)* (pp. 105–112). San Francisco, CA: Morgan Kaufmann.

Duda, R., Hart, P., & Stork, D. (2000). *Pattern classification (2nd ed.).* Wiley.

Fawcett, T. (2003). *Roc graphs: Notes and practical considerations for researchers* (Technical Report HPL-2003-4). HP Laboratories.

Ferri, C., & Hernndez-Orallo, J. (2004). Cautious classifiers. *Proceedings of the 1st International Workshop on ROC Analysis in Artificial Intelligence (ROCAI-2004)* (pp. 27–36).

Hullermeier, E. (2004). Instance-based prediction with guaranteed confidence. *Proceedings of the 16th European Conference on Artificial Intelligence (ECAI-2004)* (pp. 97–101).

Perner, P. (2006). *Proceedings of 6th Industrial Conference on Data Mining, (ICDM 2006), Leipzig, Germany, July 14-15, 2006*. IBaI Report.

Proedru, K., Nouretdinov, I., Vovk, V., & Gammerman, A. (2002). Transductive confidence machines for pattern recognition. *Proceedings of the 13th European Conference on Machine Learning (ECML-2002)* (pp. 381–390). Springer.

Quinlan, J. (1993). *C4.5: programs for machine learning*. San Mateo, CA: Morgan Kaufmann.

Saunders, C., Gammerman, A., & Vovk, V. (1999). Transduction with confidence and credibility. *Proceedings of the 16th International Joint Conference on Artificial Intelligence* (pp. 722–726).

Smirnov, E., & Kaptein, A. (2006). Theoretical and experimental study of a meta-typicalness approach for reliable classification. *Proceedings of the IEEE International Workshop on Reliability Issues in Knowledge Discovery (RIKD-06)* (p. (in print)). IEEE Computer Society.

Smirnov, E., Sprinkhuizen-Kuyper, I., Nalbantov, G., & Vanderlooy, S. (2006a). Version space support vector machines. *Proceedings of the 17th European Conference on Artificial Intelligence (ECAI-2006)* (pp. 809–810).

Smirnov, E., Vanderlooy, S., & Sprinkhuizen-Kuyper, I. (2006b). Meta-typicalness approach to reliable classification. *Proceedings of the 17th European Conference on Artificial Intelligence (ECAI-2006)* (pp. 811–812).

Vovk, V., Gammerman, A., & Shafer, G. (2005). *Algorithmic learning in a random world*. Springer.

Wolpert, D. (1992). Stacked generalization. *Neural Networks*, *5*, 241–260.