

Advances in Class Noise Detection

Borut Sluban¹ and Dragan Gamberger² and Nada Lavrač^{3,4}

Abstract. Noise filtering is usually used in data preprocessing to improve the accuracy of induced classifiers. Our goal is different: we aim at detecting noisy instances to be inspected by the domain expert in the phase of data understanding. Consequently, our noise detection algorithms should have high precision of class noise detection, where the precision-recall trade-off is modeled using the F -measure. New variants of class noise detection algorithms have been developed, including the high agreement random forest filter which ensures very high precision of identified erroneous data instances.

1 INTRODUCTION

The quality of real-world data is inevitably subject to errors and other irregularities that are usually referred to as noise. While data measurement errors represent noise which should best be repaired or discarded from the data, other irregularities such as outliers may actually lead to the discovery of intriguing new information.

Both types of irregularities, errors and outliers, appear in the data as class noise (errors in class labels) and/or attribute noise (errors in one or more attributes). Regardless of the type of noise, noise may have an adverse impact on the quality of information retrieved from the data, models created from the data, and decisions made based on the data. The impacts of both types of noise on reduced classification accuracy, increased time complexity and size of the constructed classification or representation model were studied by Zhu & Wu [6].

The main concern of standard noise handling approaches used in machine learning is improved classification accuracy of models induced from the data. These approaches include the filtering approach by Brodley & Friedl [2], Khoshgoftaar et. al [4] and Zhu et. al [7]. In contrast, the focus of this work is on detecting noisy instances to be inspected by human experts in the phase of data understanding and data cleaning. Therefore, our approach should result in highest possible precision of noise detection with a reasonable recall. In the experimental evaluation of noise detection methods presented in this work, a variant of the F -measure combining precision and recall is used: $F_{0.5}$ which weights precision twice as much as recall.

This paper presents several implementations of classification filtering algorithms for noise detection as well as a new high agreement random forest filter. In the experimental evaluation on two noiseless and two real-world datasets with randomly injected artificial class noise, the proposed high agreement random forest filter resulted in significantly improved noise detection compared to the other classification filtering approaches.

2 EXPERIMENTAL SETTING

We experimented with different classification filters for noise detection on two noiseless and two real-world datasets specified in Table 1. All the datasets have been injected with three different levels of class noise: in 2%, 5% and 10% of the dataset size, class values of instances were replaced. We have intentionally chosen two artificial noiseless UCI datasets containing instances of all feasible attribute-value configurations of these domains. The other two datasets are real-world datasets, containing their own noisy instances. All the datasets have two different class values.

For each of the four datasets and for each noise level we made ten datasets with randomly injected noise. In this way each filtering algorithm was tested not only on four different datasets but actually on 120 datasets, allowing for massive testing of noise detection algorithms despite the relatively small number of selected domains.

Table 1. Datasets used for experimental evaluation. Datasets TTT and KRKP are taken from the UCI machine learning repository [1], MCHD is described in [3], and NA is described in [5].

Dataset		Instances	Attributes	Type
TTT:	Tic Tac Toe	956	9	noiseless
KRKP:	King-Rook vs. King-Pawn	3196	36	noiseless
MCHD:	Medical-CHD	219	37	real-world
NA:	News Articles	464	500	real-world

3 NOISE FILTERING ALGORITHMS

This section presents modifications and implementations of noise filtering algorithms which have already been previously proposed in the literature [2, 4, 7] and a newly proposed high agreement random forest filter.

3.1 Classification Filters

The idea behind this filtering approach presented by Brodley & Friedl [2] is to use a classifier as a tool for detecting noisy instances in data. In this work the simple classifiers used in [2] were replaced by contemporary classifiers, because the assumption is that we should trust the classifiers that they are able to correctly predict the class of a data instance, and the incorrectly classified instances are considered to be noise. We performed classification filtering in a ten-fold cross-validation manner with five different classifiers:

Bayes: Naïve Bayes classifier (as the baseline filtering algorithm)

RF100: Random forest classifier with 100 decision trees

RF500: Random forest classifier with 500 decision trees

SVM: Support vector machine classifier (radial basis kernel)

SVMEasy: Support vector machine classifier which performs data scaling and parameter optimization

¹ Jožef Stefan International Postgraduate School, Ljubljana, Slovenia, email: borut.sluban@ijs.si

² Rudjer Bošković Institute, Zagreb, Croatia, email: dragan.gamberger@irb.hr

³ Jožef Stefan Institute, Ljubljana, Slovenia, email: nada.lavrac@ijs.si

⁴ University of Nova Gorica, Nova Gorica, Slovenia

Table 2. Results of different noise filtering algorithms on four datasets with various levels of injected class noise, where Pr stands for precision of detected artificially injected class noise and $F_{0.5}$ for the F -measure which weights precision twice as much as recall.

Dataset	Noise level (%)	(A) Classification filters										(B) High Agreement RF filter			
		Bayes		RF100		RF500		SVM		SVMEasy		RF570		RF580	
		Pr (%)	$F_{0.5}$	Pr (%)	$F_{0.5}$	Pr (%)	$F_{0.5}$	Pr (%)	$F_{0.5}$	Pr (%)	$F_{0.5}$	Pr (%)	$F_{0.5}$	Pr (%)	$F_{0.5}$
TTT	2	4.23	0.052	14.84	0.178	15.65	0.187	54.36	0.597	54.91	<i>0.603</i>	87.31	0.813	100.00	0.773
	5	10.96	0.132	30.98	0.357	32.19	0.370	75.58	0.793	75.36	0.791	92.24	0.847	99.44	0.759
	10	20.61	0.240	44.17	0.492	45.89	0.508	86.71	0.888	86.71	0.888	94.01	0.832	99.00	0.689
KRKP	2	12.44	0.150	60.10	0.652	60.20	0.653	18.92	0.225	79.25	<i>0.826</i>	87.79	0.892	92.62	0.921
	5	26.02	0.303	75.98	0.796	77.80	0.812	37.8	0.428	83.41	<i>0.861</i>	94.81	0.947	96.98	0.951
	10	40.93	0.457	84.20	0.866	84.92	0.872	56.97	0.616	85.00	<i>0.873</i>	96.38	0.950	98.39	0.936
MCHD	2	19.73	0.234	36.32	0.416	41.61	<i>0.471</i>	17.36	0.208	19.64	0.234	64.10	0.688	78.17	0.799
	5	40.16	0.454	60.15	0.650	61.05	<i>0.659</i>	36.51	0.415	37.56	0.427	85.40	0.854	90.62	0.866
	10	54.66	0.595	76.24	0.795	79.10	<i>0.822</i>	52.29	0.572	50.73	0.556	94.42	0.919	95.54	0.835
NA	2	18.78	0.224	24.90	0.292	24.65	0.289	27.25	0.318	33.40	<i>0.385</i>	92.14	0.902	100.00	0.872
	5	36.35	0.413	44.91	0.502	46.21	0.515	49.64	0.550	55.93	<i>0.611</i>	97.22	0.927	100.00	0.856
	10	53.13	0.580	64.37	0.685	64.37	0.686	67.42	0.715	70.33	<i>0.740</i>	97.25	0.910	100.00	0.772

The precision (Pr) of noise detection and the $F_{0.5}$ -score of different classification filters on the datasets with different levels of injected noise are presented in Table 2.

Concerning the basic classification filters (part (A) of Table 2) the results show that the filter with SVMEasy has the highest precision of noise detection and the highest $F_{0.5}$ -score on all datasets except the MCHD datasets where the RF100 and RF500 achieved better results. The best $F_{0.5}$ -scores for each dataset achieved among the basic classification filters are in italic.

3.2 High Agreement Random Forest Filter

When used for filtering, random forest (RF) classification filters act like an ensemble filter, since randomly generated decision trees vote for the class to be assigned to an instance. However, if the class that got the maximal number of votes has won only by a single vote, this result is not really convincing, since the decision trees are generated randomly. Therefore, we decided to demand high agreement of classifiers, i.e., a significantly increased majority of votes, namely, for two class domains used in our experiments this means that more than 60, 70, 80 or 90% of the randomly generated decision trees should classify an instance into the opposite class in order to claim that it is noisy.

We have tested this agreement based approach with RF classifiers with 100 and 500 decision trees, each with four different agreement levels: 60, 70, 80 and 90%. Since the results obtained did not vary a lot depending on the number of trees, we present only the results obtained with RF500.

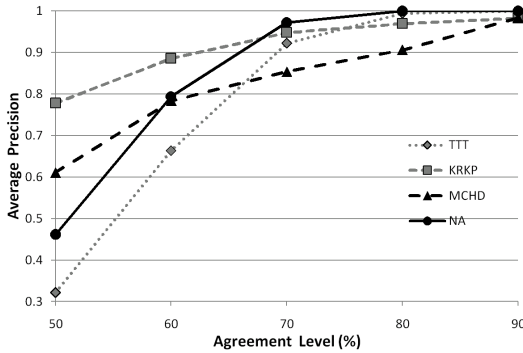


Figure 1. Precision of noise detection based on the agreement level for RF500 on all domains with 5% injected noise.

Figure 1 clearly indicates the increase in precision based on the increased agreement level. However, the best results in terms of $F_{0.5}$ -scores are achieved at agreement levels 70 and 80%. Therefore, only the high agreement random forest filters using RF500 with 70 and 80% agreement level — referred to as RF570 and RF580, respectively — are presented in part (B) of Table 2. For each dataset the highest $F_{0.5}$ -score achieved by the filtering algorithms evaluated in our work (concerning the entire Table 2) is in bold.

The results in Table 2 show that our newly proposed high agreement random forest filters RF570 and RF580 outperform all the other classification filters in terms of precision of noise detection and furthermore achieve the highest $F_{0.5}$ -scores on all except one of the datasets.

4 CONCLUSIONS

We developed several new implementations of classification based noise filtering algorithms and evaluated their performance using the $F_{0.5}$ measure which trades off precision and recall of noise detection. The evaluation shows that the winning approach is the new high agreement random forest filter (with 70% agreement level of decision tree classifiers in the random forest). It outperforms other basic classification filters in terms of precision and the $F_{0.5}$ score. In further work we will focus on this filtering approach and evaluate its capacity for filtering massive amounts of data.

References

- [1] A. Asuncion and D.J. Newman. UCI machine learning repository - <http://www.ics.uci.edu/~mlearn/MLRepository.html>, 2007.
- [2] C. E. Brodley and M. A. Friedl, 'Identifying mislabeled training data', *Journal of Artificial Intelligence Research*, **11**, 131–167, (1999).
- [3] D. Gamberger, N. Lavrač, and G. Krstajić, 'Active subgroup mining: A case study in a coronary heart disease risk group detection', *Artificial Intelligence in Medicine*, **28**, 27–57, (2003).
- [4] T. M. Khoshgoftaar, S. Zhong, and V. Joshi, 'Noise elimination with ensemble-classification filtering for software quality estimation', *Intelligent Data Analysis*, **9**(1), 3–27, (2005).
- [5] S. Pollak, 'Text classification of articles on kenyan elections', in *Proceedings of the 4th Language & Technology Conference: Human language technologies as a challenge for computer science and linguistics*, pp. 229–233, (2009).
- [6] X. Zhu and X. Wu, 'Class noise vs. attribute noise: A quantitative study of their impacts', *Artificial Intelligence Review*, **22**, 177–210, (2004).
- [7] X. Zhu, X. Wu, and Q. Chen, 'Eliminating class noise in large datasets', in *Proc. of the Int. Conf. on Machine Learning*, pp. 920–927, (2003).