An Ensemble of Rule-based Classifiers for Incomplete Data

Cao Truong Tran*†, Mengjie Zhang*, Peter Andreae*, Bing Xue* and Lam Thu Bui†
*Evolutionary Computation Research Group
Victoria University of Wellington, PO Box 600, Wellington 6140, New Zealand
Email: {cao.tran, mengjie.zhang, peter.andreae, bing.xue}@ecs.vuw.ac.nz
†Le Qui Don Technical University, 236 Hoang Quoc Viet Street, Ha Noi, Viet Nam
Email: lambt@lqdtu.edu.vn

***Abstract*—Many real-world datasets suffer from the problem of missing values. Imputation which replaces missing values with plausible values is a major method for classification with data containing missing values. However, powerful imputation methods including multiple imputation are usually computationally intensive for estimating missing values in unseen incomplete instances. Rule-based classification algorithms have been widely used in data mining, but the majority of them are not able to directly work with data containing missing values. This paper proposes an approach to effectively combining multiple imputation, feature selection and rule-based classification to construct a set of classifiers, which can be used to classify any incomplete instance without requiring imputation. Empirical results show that the method not only can be more accurate than other common methods, but can also be faster to classify new instances than the other methods.**

I. Introduction

Classification is a major task in data mining [10], which includes two main phases: a training phase and an application phase. The training phase constructs classifiers which are then used to classify new instances in the application phase. Classification has numerous applications in computer science, engineering, medicine, etc. There are still open problems, and one problem is incomplete data [8].

Missing values are an unavoidable problem in many real-world datasets [13]. For instance, 45% of the datasets in the UCI Machine Learning Repository [12], which is a popular benchmark database for data mining, have missing values [8].

Missing values lead to serious issues for classification. One of the most serious issues is that almost all classification algorithms cannot directly work with datasets which contain missing values. In addition, missing values usually result in big classification errors [8].

A common approach to tackling missing values is to utilise imputation methods to substitute missing values with suitable values. Imputation methods can generate complete data which can be then utilised by any classification algorithm. Simple imputation methods such as mean imputation are often fast, but inaccurate. In contrast, powerful methods such as multiple imputation are often accurate, but computationally expensive [15]. Therefore, how to effectively and efficiently use imputation should be investigated.

Rule-based classification algorithms have been widely used in data mining. The key reason is that rule-based classifiers are easy for people to understand and interpret, and have comparable performance to other classifiers [4], [7], [16]. However, the majority of rule-based classification algorithms cannot directly work with incomplete data. In order to deal with missing values, they must to be combined with imputation to generate complete data from incomplete data. The problem is that simple imputation methods such as mean imputation result in big classification error, but powerful (effective) imputation methods such as multiple imputation [17] are very computationally expensive. It is not straightforward how to combine rule-based classification algorithms and imputation in a way that is both effective and efficient, particularly in the application phase.

Feature selection is the process of selecting suitable feature subsets from original features. Feature selection is able to improve classification, but it is usually applied to datasets without missing values [18], [19]. This paper shows a method to use feature selection to improve classification with datasets containing missing values.

Ensemble learning is the process to build multiple classifiers instead of a single classifier for classification tasks, and has been proven to be able to improve classification accuracy [6]. Ensemble learning has also been applied to incomplete data by building multiple classifiers, which can be then utilised to classify new incomplete instances without requiring imputation [3], [11], [14]. However, when incomplete data contains numerous missing values, existing ensemble methods cannot work well, and require a large number of classifiers. This paper also shows how to construct a small number of classifiers which can directly classify incomplete instances, and perform very well even when incomplete data has numerous missing values.

A. Research goals

This paper proposes a method which combines imputation and rule-based classification in an effective and efficient way, exploiting the power of two well-known techniques: ensemble and feature selection. It uses multiple imputation to build a novel ensemble of specialised classifiers enhanced by feature selection, and then uses the ensemble to efficiently classify new instances without the need of imputation. The proposed

978-1-5386-0743-5/17/$31.00 ©2017 IEEE

method is compared with common methods for classification with incomplete data to address the following objectives:

1) Whether the proposed method can be more accurate, and can be faster than using imputation both in the training phase and the application phase; and
2) Whether the proposed method can be more accurate, and can be faster than existing ensemble methods for classification with incomplete data.

B. Organisation

The remainder is organized as follows. Section II shows related work. The proposed method is shown in Section III. Experiment design is presented in Section IV. Results and analysis are shown Section V. Section VI states conclusions and proposes future work.

II. Related Work

A. Imputation for Classification with Incomplete Data

The purpose of imputation is to substitute missing values with suitable values. Imputation can generate complete data that then is able to utilise by any classification algorithms. Therefore, imputation is one of the most common ways to tackle classification with data containing missing values [8].

Fig. 1 presents the main steps of applying imputation for classification with incomplete data. In the training phase, an imputation method is used to transform incomplete training data into complete training data, which is then utilised by an classification algorithm to build a classifier. In the application phase, complete instances will be directly classified by the learnt classifier. To classify an incomplete instance, first, it is put into the imputation method to produce an imputed instance, which is then classified by the learnt classifier.

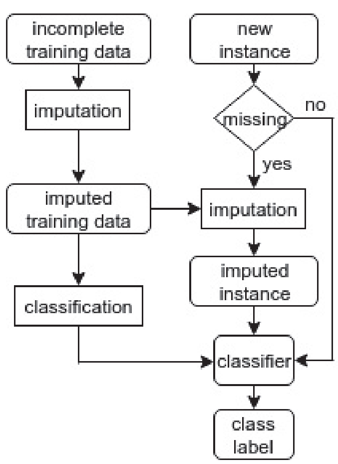

Fig. 1. Using imputation for classification with incomplete data.

Imputation can be categorised into single imputation and multiple imputation[13]. Single imputation estimates one value for each missing value. Multiple imputation estimates a set of values for each missing value. For example, kNN-based imputation is a popular single imputation method which estimates missing values in an incomplete instance by the average of k nearest instances [1]. Multiple imputation using chained equations (MICE) is a powerful multiple imputation method which uses regression methods to estimate missing values [17].

Both single imputation methods and multiple imputation methods have been widely applied to classification with incomplete data [15]. Multiple imputation methods usually obtain better accuracy than single imputation methods since multiple imputation methods can reflect the uncertainty of missing data better than single imputation methods. Nevertheless, multiple imputation methods are frequently more expensive than single imputation methods because multiple imputation methods have to estimate multiple values for each missing value rather than one value as single imputation methods [15].

B. Rule-based Classification

Rule-based classification refers to any classification scheme which uses IF-THEN rules for class prediction. Rule-based classification methods have been widely applied because they are easy to understand and interpret. Moreover, rule-based classifiers can be comparable to other methods such as decision trees, and can rapidly classify new instances [4], [7].

A rule-based classification scheme consists of there main components: a rule induction algorithm, a rule ranking measure and a class prediction algorithm. The rule induction algorithm refers to the process of constructing relevant IF-THEN rules. The rule ranking measure is used to evaluate the quality of each rule and rank them. The class prediction algorithm is used to predict the class of a new instance based on the rules along with their rankings from previous steps [16].

There are two main approaches to extracting rules. One approach is to extract rules from a decision tree by building a decision tree which is then transformed into a set of rules [7]. Another approach is to extract rules directly from data by using the “separate-and-conquer” strategy to gradually extract rules for each class label [4].

C. Feature Selection

The purpose of feature selection is to select suitable feature subsets from original features. Thanks to removing redundant and irrelevant features, feature selection is able to result in more accurate and simpler classifiers compared to using all features. However, feature selection is a NP-hard problem since the number of feature subsets is 2^n, where n refers to the number of original features [19]. Moreover, feature selection is mainly utilised for complete data, but it is seldom utilised for incomplete data.

Feature selection consists of a measure technique and a search technique. The measure technique is used to evaluate the goodness of feature subsets. The search technique is used to search for feature subsets. The quality of feature selection strongly depends both of the techniques [19].

Measure techniques for scoring feature subsets can be divided into filter techniques and wrapper techniques. A filter technique utilises a measure like information gain to score feature subsets, which is independent from any classification algorithm. A wrapper technique utilises a classifier to score feature subsets. Wrapper techniques are often more accurate, but more expensive than filter techniques [19].

Search techniques for exploring feature subsets can be categorised into tradition search techniques and evolutionary search techniques. Sequential forward and backward selection are two common traditional search techniques. Recently, evolutionary techniques such as genetic algorithms (GAs), particle swarm optimisation (PSO), and differential evolution (DE) have been widely applied to feature selection [19].

D. Ensemble Learning for Classification with Incomplete Data

Ensemble learning is a learning technique which constructs multiple classifiers instead of a single classifier for a classification task. Experiments show that an ensemble of classifiers can often perform better than any single classifier [6].

Ensemble methods also have been applied to classification with datasets containing missing values. In [3], [11], an ensemble of neural networks is constructed to deal with missing values. The ensemble can directly classify incomplete data, but it cannot work well when the incomplete data contains numerous missing values. Learn^{++}MF builds a set of classifiers for incomplete data in [14], where a single classifier is build on a subset of training data by randomly selecting a feature subset. To cover almost all missing cases, Learn^{++}MF needs to construct a huge number of classifiers. Although the proposed method can classify incomplete instances without requiring imputation, it possibly takes time to find suitable classifiers for each incomplete instance.

Existing ensemble methods for classification with incomplete data can address missing values to some extent. However, they require building a large number of classifiers, and cannot work well when datasets contain numerous missing values. Therefore, how to effectively use ensemble techniques for classification datasets containing missing values should be further investigated.

III. The Proposed Method

The proposed method has four key ideas. The first idea is to only use imputation in constructing the training data, but not to use imputation in the application phase. Multiple imputation produces sufficiently high quality imputed data, but multiple imputation is too expensive to use during the application phase. However, there is often no constrain on the training time so that the use of multiple imputation is feasible during training. The second idea is to use feature selection to not only further improve the training data by removing redundant and irrelevant features, but also to reduce the number of missing values in the instances in the application phase. The third idea is to identify patterns of missing values in the training data, and then build a classifier for each such pattern. These classifiers enable us to classify many new incomplete instances without requiring imputation. Finally, rule-based classification algorithms, which often only include a small number of features, are used to construct the classifiers. Therefore, the classifiers are tolerant of even a large number of missing values.

The proposed method has two phases: a training phase and an application phase. The training phase builds a set of classifiers that is then utilised to classify new incomplete instances without using imputation in the application phase. Fig. 2 shows the main steps of the proposed method.

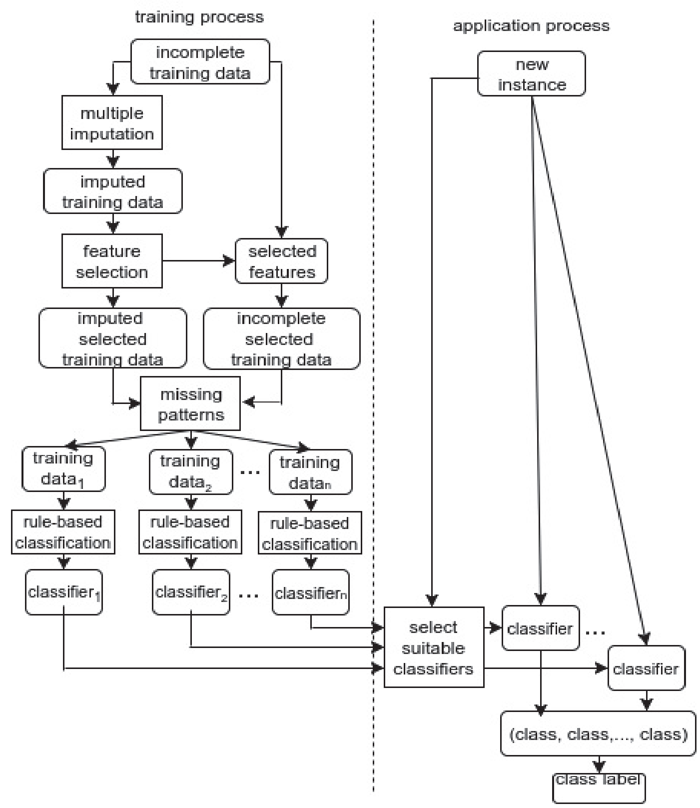

Fig. 2. The proposed method.

In the training phase, firstly, the incomplete training data is put through multiple imputation to construct an imputed training dataset that no longer has any missing values. After that, a feature selection method is used to remove redundant and irrelevant features and generate a training dataset with only relevant features and no missing values. We also remove the same redundant and irrelevant features from the original training data to generate an incomplete selected training dataset that still contains missing values. This incomplete selected training data is then searched to find all "missing patterns ". A missing pattern is a set of features such that there is at least one instance in the training dataset, which contains missing values for exactly the features in the pattern. Subsequently, we construct a separate training dataset corresponding to each missing pattern by removing the features in the pattern from the imputed selected training dataset. Each of these training datasets is then put into a rule-based classification algorithm to construct a classifier.

In the application phase, a *complete* instance will be directly classified by all the learnt classifiers, and then a majority vote is used to decide a final class label. To classify an *incomplete* instance, the proposed method first searches for all suitable classifiers which do not require features that are missing in the incomplete instance. Subsequently, the incomplete instance is classified by the suitable classifiers, and then a final class label is decided by using a majority vote. In the rare cases when there are no suitable classifiers, the method uses mean imputation to construct a complete vesion of the instance.

IV. Experiment Design

A. Comparison Methods

To investigate the effectiveness and efficiency of the proposed method, it is compared with two common approaches to classification with datasets containing missing values. The first compared method uses imputation in both the training phase and the application phase as shown in Fig. 1. The second compared method is Learn^{++}MF in [14], which builds multiple classifiers in the training phase, and does not require imputation in the application phase.

B. Datasets

The experiments evaluate the proposed method on five datasets. Table I shows main characteristics of the datasets: the number of instances, the number of features, the number of classes, and the proportion of incomplete instances which include at least one missing value.

TABLE I
BENCHMARK DATASETS

Name	#Instances	#Features	#Classes	Incomplete instances (%)
Arrhythmia	452	279	16	85.11
Horsecolic	368	23	2	98.1
Housevotes	435	16	2	46.67
Mammographic	961	5	2	13.63
Ozone	2536	73	2	27.12

None of the datasets was originally separated into training and test sets. Furthermore, some of them have a small number of instances. Therefore, 10-fold cross validation is utilised to separate the datasets into training sets and test sets. Because 10-fold cross validation is a stochastic technique, it is performed 30 times on each dataset. Consequently, for each dataset, 300 pairs of training and test sets are used to compare the proposed method and the benchmark methods.

C. Parameter Settings

1) Imputation: kNN-based imputation and MICE are used in the experiments. These imputation methods are chosen to represent two groups of imputation including a single imputation method and a multiple imputation method, respectively. kNN-based imputation with $k=1$ is utilised because it is fast and non-parametric. MICE in R [2] is used for MICE's implementation.

2) Rule-based Classification: PART [7] and JRip [4], which are two well-known rule-based classification algorithms, are used in the experiments. WEKA [9] is used to implement the classification algorithms.

3) Feature Selection: The experiments use a wrapper method to perform feature selection. A rule-based classifier is utilised to score feature subsets. Every serach method is able to use to explore feature subsets, but DE is used in the experiments because it is a powerful search technique and not complicated, so it is easy to understand and implement. DE's parameters for feature selection are set as follows. The population size is set to 50, and the number of iteration is set to 100. The crossover rate is 0.25, and the mutation factor is 1. The threshold θ , which is utilised to decide whether or no a feature is chosen, is set as 0.6

V. Results and Analysis

A. Classification Accuracy

Table II shows the mean along with the corresponding standard deviation of the proposed method and the compared methods over the 300 pairs of training sets and test sets. "*MICEFs*" refers to the proposed method as presented in Fig. 2. "*MICE*" and "*kNN*" refer to the compared method as shown in Fig. 1 by utilising MICE and kNN-based imputation, respectively. "*Learn^{++}MF*" refers to the compared method in [14].

On each dataset, classification accuracy is the average of accuracy over 30 times using 10-fold cross validation. The Friedman test [5], which is a popular non-parametric tests for multiple comparison, is used to statistically test whether having significantly difference in classification accuracy. The test shows that there exist differences in all datasets. Consequently, a post-hoc procedure using the Holm procedure [5] is used to perform pairwise comparisons. In Table II, "T" columns present significance test of the previous column against the proposed method, where "+" means the proposed method is significantly more accurate, "=" means no significant difference, and "-" means significantly less accurate.

As can be seen from Table II that the proposed method (*MICEFs*) can achieve significantly better classification accuracy than the other methods in almost all cases. *MICEFs* achieves significantly better classification accuracy in nine cases and similar classification accuracy in one case compared to *MICE*. *MICEFs* also achieves significantly better classification accuracy in eight cases and similar classification accuracy in two cases compared to *kNN* and *Learn^{++}MF*.

To confirm whether *MICEFs* is really significantly more accurate than the other methods, we do Friendman's test on the accurates of all algorithms in all cases. Table III presents the ranking of the algorithms by Friedman's test (smaller means better). It is clear from Table III that *MICEFs* is the best algorithm, followed by *MICE*, *kNN* and *Lean^{++}MF*. Moreover, we carry out post-hoc tests to perform pairwise comparisons. Table IV presents pairwise comparisons by Holm which is a post-hoc procedure . It is also clear from Table IV

TABLE II
CLASSIFICATION ACCURACY AND STANDARD DEVIATION.

Dataset	Classifier	MICEFs	MICE	T	kNN	T	Learn^{++}MF	T
Arrhythmia	PART	**68.15±2.77**	64.06±1.89	+	63.76±2.04	+	63.43±3.24	+
	JRip	**70.51±1.69**	68.81±2.06	+	68.99±1.93	+	65.17±1.98	+
Horse-colic	PART	**84.89±0.81**	79.25±1.77	+	79.23±1.32	+	76.23±1.60	+
	JRip	**85.55±0.66**	85.06±0.69	+	84.92±0.94	+	80.03±1.56	+
Housevotes	PART	**96.13±0.41**	95.72±0.79	+	95.74±0.63	+	92.13±0.53	+
	JRip	**95.82±0.47**	95.52±0.54	+	95.63±0.39	=	92.44±0.91	+
Mammographic	PART	**82.15±0.56**	81.52±0.70	+	81.45±0.59	+	77.87±2.09	+
	JRip	82.56±0.59	82.49±0.53	=	**82.68±0.75**	=	79.73±1.27	+
Ozone	PART	**97.11±0.02**	95.86±0.44	+	95.24±1.15	+	97.11±0.02	=
	JRip	**97.07±0.06**	96.16±0.45	+	96.03±0.52	+	97.06±0.06	=

that *MICEFs* is significantly more accurate than each of the other algorithms.

TABLE III
THE RANKING OF THE ALGORITHMS BY FRIEDMAN'S TEST

Algorithm	Ranking
MICEFs	1.15
MICE	2.60
kNN	2.7
Learn^{++}MF	3.55

TABLE IV
PAIRWISE COMPARISONS BY HOLM'S TEST

Algorithms	Holm
MICEFs vs. Learn^{++}MF	**0.0083**
MICEFs vs. Knn	**0.0100**
MICEFs vs. MICE	**0.0125**
MICE vs. Learn^{++}MF	0.0167
Knn vs. Learn^{++}MF	0.0250
MICE vs. Knn	0.0500

Bold values indicate the significant difference.

In summary, the proposed method can be more accurate than the both benchmark methods.

B. Computation Time

Table V shows the computation time of the proposed method and the compared method (in millisecond) to classify all instances in the application phase.

As can be seen clearly from Table V that the proposed method is remarkably faster than the other methods. *MICEFs* is thousand times faster than *MICE* because *MICEFs* does not have to spend time to substitute missing values when it classifies incomplete instance. In contrast, MICE has to spend a long time to substitute missing values before classifying the instance. *MICEFs* is also much quicker than *kNN* because *kNN* also has to spend time to estimate missing values in the application phase. *Learn^{++}MF* does not spend time to estimate missing values in the application phase, although it is still slower than *MICEFs*. The underlying reason is possibly that *MICEFs* uses feature selection to remove redundant and irrelevant features, so it often generates more compact classifiers than *Learn^{++}MF*. Furthermore, by removing redundant and irrelevant features, *MICEFs* reduces the number of incomplete instance, so it takes less time to search for suitable classifiers.

In summary, the proposed method can not only be more accurate, but can also be faster than the other methods.

C. Further Analysis

In order to know how *MICEFs* can achieve better classification accuracy than the other methods, we further analyse classifiers evolved by *MICEFs*. Fig. 3 shows the percentage of missing patterns reduction by using feature selection. As can be seen from Fig. 3, feature selection can remarkably remove missing patterns in training data. For example, feature selection can reduce more than 50% missing patterns in the *Horsecolic* dataset. Therefore, *MICEFs* only needs to build a small number of classifiers, but it still can classify incomplete instances without requiring imputation.

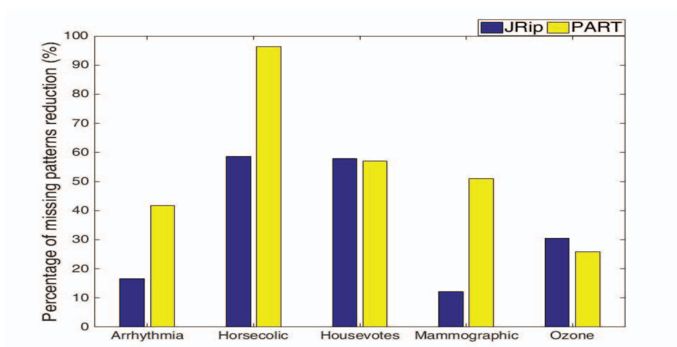

Fig. 3. Percentage of missing patterns reduction by using feature selection.

Fig. 4 shows the percentage of the learnt classifiers which are used to classify each incomplete instance. In training phase, *MICEFs* only constructs one classifier for each missing pattern. However, it is clear from Fig. 4 that more than 50% of the learnt classifiers are used to classify each incomplete instance. The underlying reason is that rule-based classification algorithms can implicitly remove redundant and irrelevant features. Therefore, rule-based classifiers can be used to classify

TABLE V
COMPUTATION TIME (MILLISECOND).

Dataset	Classifier	MICEFs	MICE	kNN	LearnMF
Arrhythmia	PART	3.4×10^1	7.5×10^7	4.9×10^2	3.6×10^3
	JRip	2.0	2.3×10^5	2.1×10^1	1.6×10^1
Horse-colic	PART	1.0	2.6×10^6	1.2×10^1	1.1×10^1
	JRip	6.0	2.5×10^6	9.0	2.1×10^1
Housevotes	PART	6.0	4.6×10^5	9.0	2.4×10^1
	JRip	5.0	5.1×10^5	1.0×10^1	2.7×10^1
Mammographic	PART	1.0	2.6×10^5	1.7×10^1	4.0
	JRip	4.0	1.9×10^5	3.2×10^1	7.0
Ozone	PART	2.7×10^2	1.2×10^9	1.3×10^3	2.9×10^5
	JRip	1.3×10^3	1.4×10^9	1.5×10^3	6.9×10^4

incomplete instances, which contain missing values in the removed features. In other words, a rule-based classifier for one missing pattern can be used for other missing patterns. By building a set of classifiers, *MICEFs* can not only classify incomplete instances without imputation, but can also obtain better accuracy than the other methods.

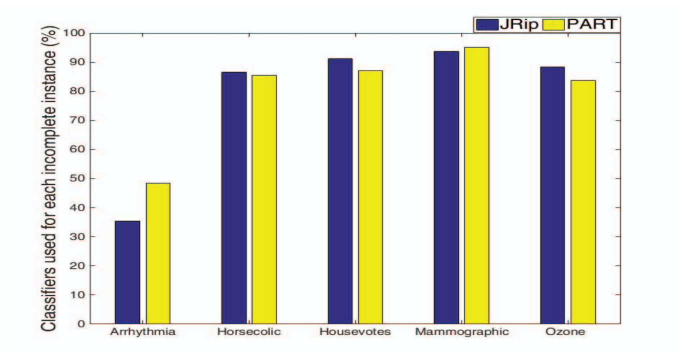

Fig. 4. Percentage of classifiers used to classify each incomplete instance

VI. CONCLUSIONS

This paper proposed an approach to constructing a small number of specialised effective classifiers for incomplete data. The proposed method uses multiple imputation to provide a high quality training data, which then is further improved by feature selection. Subsequently, it builds one classifier for each pattern of missing values identified in the training data. The method can use this set of classifiers to directly classify incomplete instances without the need for expensive imputation. Empirical results demonstrate that the proposed method is more accurate and faster than using imputation in both the training and application phasees. Furthermore, in comparison to existing ensemble methods, the new method can achieve better classification accuracy, and can classify new incomplete instances more quickly.

The method used a wrapper feature selection, future work could investigate using filter feature selection. The method was evaluated using PART and JRip, but there are other rule-based classification algorithms that could be explored in future work.

ACKNOWLEDGEMENT

This research is funded by Vietnam National Foundation for Science and Technology Development (NAFOSTED) under grant number 102.01-2015.12.

REFERENCES

[1] G. E. Batista and M. C. Monard. An analysis of four missing data treatment methods for supervised learning. *Applied artificial intelligence*, 17:519–533, 2003.
[2] S. Buuren and K. Groothuis-Oudshoorn. mice: Multivariate imputation by chained equations in r. *Journal of statistical software*, 45, 2011.
[3] H. Chen, Y. Du, and K. Jiang. Classification of incomplete data using classifier ensembles. In *Systems and Informatics (ICSAI), 2012 International Conference on*, pages 2229–2232, 2012.
[4] W. W. Cohen. Fast effective rule induction. In *Proceedings of the twelfth international conference on machine learning*, pages 115–123, 1995.
[5] J. Demšar. Statistical comparisons of classifiers over multiple data sets. *Journal of Machine learning research*, 7:1–30, 2006.
[6] T. G. Dietterich et al. Ensemble methods in machine learning. *Multiple classifier systems*, 1857:1–15, 2000.
[7] E. Frank and I. H. Witten. Generating accurate rule sets without global optimization. 1998.
[8] P. J. García-Laencina, J.-L. Sancho-Gómez, and A. R. Figueiras-Vidal. Pattern classification with missing data: a review. *Neural Computing and Applications*, 19:263–282, 2010.
[9] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten. The weka data mining software: an update. *ACM SIGKDD explorations newsletter*, 11:10–18, 2009.
[10] J. Han, J. Pei, and M. Kamber. *Data mining: concepts and techniques*. Elsevier, 2011.
[11] K. Jiang, H. Chen, and S. Yuan. Classification for incomplete data using classifier ensembles. In *Neural Networks and Brain, 2005. ICNN&B'05. International Conference on*, volume 1, pages 559–563. IEEE, 2005.
[12] M. Lichman. UCI machine learning repository, 2013.
[13] R. J. Little and D. B. Rubin. *Statistical analysis with missing data*. John Wiley & Sons, 2014.
[14] R. Polikar, J. DePasquale, H. S. Mohammed, G. Brown, and L. I. Kuncheva. Learn++MF: A random subspace approach for the missing feature problem. *Pattern Recognition*, 43:3817–3832, 2010.
[15] C. T. Tran, M. Zhang, and P. Andreae. A genetic programming-based imputation method for classification with missing data. In *EuroGP*, pages 149–163, 2016.
[16] A. K. Tung. Rule-based classification. In *Encyclopedia of Database Systems*, pages 2459–2462. Springer, 2009.
[17] I. R. White, P. Royston, and A. M. Wood. Multiple imputation using chained equations: issues and guidance for practice. *Statistics in medicine*, 30:377–399, 2011.
[18] B. Xue and M. Zhang. Evolutionary feature manipulation in data mining/big data. *ACM SIGEVOlution*, 10:4–11, 2017.
[19] B. Xue, M. Zhang, W. N. Browne, and X. Yao. A survey on evolutionary computation approaches to feature selection. *IEEE Transactions on Evolutionary Computation*, 20:606–626, 2016.