Volume 5, Issue 1, January 2015 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

Performance Analysis of Data Mining Algorithms

Firas Mohammed Ali*
Student, IT Department,
Libyan Academy, Libya

Dr. Prof. El-Bahlul Emhemed Fgee
Supervisor, Computer Department,
Libyan Academy, Libya

Dr. T. Gopikrishna
External Guide, Computer Dept.,
SIRT University, Hoon, Libya

Abstract— There is no single best algorithm since it highly depends on the data any one are working with. Nobody can tell what should use without knowing the data and even then it would be just a guess. Classification algorithms of data mining have successfully been applied in the recent years to predict values. There are many classifiers out there all of them try to achieve almost the same. Identifying the best classification algorithm among all available is a challenging task. This paper focuses on finding the right algorithm for classification of data that works better on diverse data sets. However, the accuracy of such methods differs according to the classification algorithm used. This work presents a comprehensive comparative analysis of the most ten different widely used classification algorithms. Moreover, the performances of these algorithms have been evaluated by using different data sets. Each technique has been evaluated with respect to accuracy, errors percentage and execution time, performance evaluation has been carried out with all the ten classification algorithms. The obtained results illustrated that the RandomTree classifier has significantly better performance and had the highest accuracy among these datasets.

Keywords— Data mining, classification algorithms, performance accuracy, WEKA, RandomTree.

I. INTRODUCTION

Data Mining is knowledge mining from data, knowledge extraction, and data analysis. Data Mining involves the various data analysis tools for identifying previously unknown, valid patterns and relationships in huge data set [1]. The term Data Mining, also known as Knowledge Discovery in Databases (KDD) is the process of discovering interesting patterns and knowledge from large amount of data. There are different data mining techniques like classification, association, preprocessing, transformation, clustering, and pattern evaluation [2]. Classification and Association are the popular techniques used to predict user interest and relationship between those data items which has been used by users association, preprocessing, transformation, clustering, and pattern evaluation. Classification and Association are the popular techniques used to predict user interest and relationship between those data items which has been used by users. In this thesis the accuracy of most widely used ten algorithms are statistically compared by means of an experiment. I first describe the datasets and classifiers which are used for this experiment Classifiers are one of the main aspects in Machine Learning. Throughout this thesis ten common classifiers are compared on three various datasets. The result revealed that the RandomTree algorithm is significantly better than other classifiers.

In this thesis I presented machine learning data mining tool used for different analysis, Waikato Environment for Knowledge Analysis (WEKA) is introduced by university of New Zealand. My work shows the process of WEKA the data mining classifications. I have provided an evaluation based on applying these classification methods to the datasets and measuring the accuracy of test results. I have compared various classifiers with three different types of data sets on WEKA; I presented their result as well as about tool and data set which are used in performing evaluation [4].

II. IMPLEMENTATION TASKS OF CLASSIFICATION METHODS ON THREE DATA SETS

The goal was to see how well the different algorithms performed, not just by comparing the number of correct classifications, but also by looking into the time required to construct the classification model depending on the size of the input data and number features used of as well as the time required to classify a data set using the generated classification model. It was entirely possible to implement these algorithms into classifiers from scratch since there was a lot of documentations describing them [2].

Mainly three data sets used in this thesis are again taken from the UCI data sets [4].

A. Task A1

Load the data set 'credit-gr.arff' and run 'classifier' on it. Set the test options to 'training set', 'cross-validation' and 'Percentage split'. Note down the all resulting accuracies.

B. Task A2

Load the data set 'ionosphere.arff' and run 'classifier' on it. Set the test options to options to 'training set', 'cross-validation' and 'Percentage split'. Note down the all resulting accuracies.

C. Task A3

Load the data set 'vote.arff' and run 'classifier' on it. Set the test options to 'training set', 'cross-validation' and 'Percentage split'. Note down the all resulting accuracies. for each Task , The test applied 10 and chosen the most repeated "build time" with its results.

D. Credit German Data Set Information

This dataset classifies people described by a set of attributes as good or bad credit risks.

Table I Credit german dataset information

Dataset	Instances	Attributes	Data Type
Credit-g	1000	21	String

E. Ionosphere.arff data type

The instances describe a Classification of radar returns from the Ionosphere.

Table II Ionosphere dataset information

Dataset	Instances	Attributes	Data Type
Ionosphere	351	35	Numeric

F. Vote.arff datatype

The instances describe Congressional Voting Records of US on 16 different bills.; Classify as Republican or Democrat.

Table III Vote dataset information

Dataset	Instances	Attributes	Data Type
Vote	435	17	Nominal

III. RESULTS AND DISCUSSIONS

In this paper to evaluate performance of selected tool using the given datasets, several experiments are conducted. For evaluation purpose, three test modes are used, the training set, the cross-validation mode and percentage split mode. At the end, the recorded measures are averaged. It is common to have 66% of the objects of the original database as a training set and the rest of objects as a test set. Once the tests is carried out using the selected datasets, then using the available classification and test modes ,results are collected and an overall comparison is conducted.

a. Results for Dataset Credit-G:

Firstly run the classifier on the dataset (Credit-g), the algorithm which has the lowest mean absolute error and higher accuracy is chosen as the best algorithm. After choosing the test option as Training set, I found JRip is lowest performance as shown in table- IV with highlighted in red color whereas RandomTree has shown highest performance accuracy with highlighted in blue colour.

Secondly choosing the test option as Cross validation where the number of folds is 10, OneR algorithm shown low performance as shown in table- IV with highlighted in red color, where is LMT shown high performance accuracy in classification model with the blue row color among ten algorithms.

And Finally, after choosing the test option as percentage split where the percentage about 66% by default, I found again this RandomTree is shown lower accuracy where is LMT shows higher accuracy in this model among ten algorithms.

The following table shows low and high accuracy performances on dataset Credit.g. I found LMT has considered highest performance classification accuracy in this observation as shown in the table.

Table IV Comparison of classifiers using german credit data set in training set mode

	Classifier	Time taken to build model (Sec)	Time taken to test model on training split (Sec)	Correctly Classified Instances	Incorrectly Classified Instances	Kappa statistic	Mean absolute error	Root mean squared error	Relative absolute error	Root relative squared error	Confusion Matrix
1	NaiveBayes	0.01	0.04	77.20%	22.80%	0.43	0.28	0.41	67.14%	88.97%	a b 611 89 | a = good 139 161 | b = bad
2	SMO	2.58	0.03	78.40%	21.60%	0.45	0.21	0.46	51.40%	101.42%	a b 626 74 | a = good 142 158 | b = bad
3	KStar	0	18.65	100.00%	0.00%	1	0	0	0.01	0.20%	a b 700 0 | a = good 0 300 | b = bad
4	AdaBoostM1	0.07	0.02	73.70%	26.30%	0.22	0.34	0.41	81.49%	89.97%	a b 669 31 | a = good 232 68 | b = bad
5	JRip	0.21	0.02	74.30%	25.70%	0.34	0.36	0.43	87.24%	93.42%	a b 605 95 | a = good 162 138 | b = bad
6	OneR	0	0.02	74.30%	25.70%	0.3	0.26	0.5	61.16%	110.62%	a b 637 63 | a = good 194 106 | b = bad
7	PART	0.25	0.03	89.70%	10.30%	0.75	0.16	0.28	38.20%	61.82%	a b 653 47 | a = good 56 244 | b = bad
8	J48	0.05	0.02	85.50%	14.50%	0.62	0.23	0.34	55.03%	74.20%	a b 669 31 | a = good 114 186 | b = bad
9	LMT	12.4	0.03	77.40%	22.60%	0.42	0.3	0.38	71.10%	83.80%	a b 625 75 | a = good 151 149 | b = bad
10	RandomTree	0.02	0.02	100	0	1	0	0	0	0	a b 700 0 | a = good 0 300 | b = bad

Table V Accuracy of credit.g dataset

TestMode	Low accuracy	High accuracy
TrainingSet	JRip	RandomTree
Crossfolds 10	OneR	LMT
Percentage split	RandomTree	LMT

b. Results for Dataset ionosphere

Firstly run the classifier on the dataset (ionosphere), the algorithm which has the lowest mean absolute error and higher accuracy is chosen as the best algorithm. After choosing the test option as Training set,NaiveBayes algorithm shown lowest performance as shown in the next table with the red row color, where RandomTree has shown high performance accuracy with the blue row color in classification model among ten algorithms. After choosing the test option as Cross validation where the number of folds is 10, NaiveBayes algorithm shown low performance as shown in the next table with the red row color, where is LMT shown high performance accuracy in classification model with the blue row color among ten algorithms. And finally, after choosing the test option as percentage

split where the percentage about 66% by default, I found again this oneR is shown lower accuracy where AdaboostM1 is shows higher accuracy in this model among ten algorithms.

Table VI Comparison of classifiers using ionosphere data set in training set mode

	Classifier	Time taken to build model (Sec)	Time taken to test model on training split (Sec)	Correctly Classified Instances	Incorrectly Classified Instances	Kappa statistic	Mean absolute error	Root mean squared error	Relative absolute error	Root relative squared error	Confusion Matrix
1	NaiveBayes	0	0.03	82.90%	17.09%	0.64	0.16	0.38	36.21%	80.26%	a b 109 17 | a = b 43 182 | b = g
2	SMO	0.14	0.02	91.45%	8.54%	0.8	0.08	0.29	18.56%	60.94%	a b 101 25 | a = b 5 220 | b = g
3	KStar	0	14.93	100.00%	0.00%	1	0	0	0.00%	0.00%	a b 126 0 | a = b 0 225 | b = g
4	AdaBoostM1	0.09	0	93.73%	6.27%	0.86	0.14	0.23	30.76%	48.89%	a b 104 22 | a = b 0 225 | b = g
5	JRip	0.09	0	91.17%	8.83%	0.81	0.16	0.28	34.41%	58.67%	a b 112 14 | a = b 17 208 | b = g
6	OneR	0	0.02	86.89%	13.10%	0.72	0.13	0.36	28.46%	75.47%	a b 105 21 | a = b 25 200 | b = g
7	PART	0.08	0	99.71%	0.28%	0.99	0.01	0.05	1.06%	10.30%	a b 125 1 | a = b 0 225 | b = g
8	J48	0.05	0.02	99.71%	0.28%	0.99	0.01	0.05	1.11%	10.56%	a 125 1 | a = b 0 225 | b = g
9	LMT	3.29	0.01	96.87%	3.13%	0.93	0.07	0.17	16.24%	35.57%	a b 115 11 | a = b 0 225 | b = g
10	RandomTree	0.01	0.01	100.00%	0.00%	1	0	0	0.00%	0.00%	a b 126 0 | a = b 0 225 | b = g

Table VII Accuracy of ionosphere dataset

TestMode	Low accuracy	High accuracy
TrainingSet	NaiveBayes	RandomTree
Crossfolds 10	NaiveBayes	LMT
Percentage split	oneR	AdaBoostM1

I found AdaBoostM1, LMT and RandomTree are considered highest performance classification accuracies in this observation as shown in the table

c. Results for Dataset Vote

Firstly run the classifier on the dataset (Vote), the algorithm which has the lowest mean absolute error and higher accuracy is chosen as the best algorithm. After choosing the test option as Training set, NaiveBayes algorithm shown lowest performance as shown in the next table with the red row color, where RandomTree has shown high performance accuracy with the blue row color in

classification model among ten algorithms. After choosing the test option as Cross validation where the number of folds is 10, NaiveBayes algorithm shown low performance as shown in the next table with the red row color, where is J48 shown high performance accuracy in classification model with the blue row color among ten algorithms. And finally, after choosing the test option as percentage split where the percentage about 66% by default, I found again this NaiveBayes is shown lower accuracy where AdaboostM1 is shows higher accuracy in this model among ten algorithms.

Table VIII Comparison of classifiers using vote data set with training set mode

	Classifier	Time taken to build model (Sec)	Time taken to test model on training split (Sec)	Correctly Classified Instances	Incorrectly Classified Instances	Kappa statistic	Mean absolute error	Root mean squared error	Relative absolute error	Root relative squared error	Confusion Matrix
1	NaiveBayes	0	0.02	90.34%	9.65%	0.8	0.1	0.29	20.55%	60.47%	a b <-- classified as 238 29 | a = democrat 13 155 | b = republican
2	SMO	0.06	0.02	97.01%	2.99%	0.94	0.03	0.17	6.30%	35.50%	a b <-- classified as 259 8 | a = democrat 5 163 | b = republican
3	KStar	0	1.56	99.08%	0.92%	0.98	0.01	0.07	3.53%	14.95%	a b <-- classified as 264 3 | a = democrat 1 167 | b = republican
4	AdaBoostM1	0.02	0.02	96.32%	3.68%	0.92	0.05	0.15	9.98%	31.97%	a b <-- classified as 258 9 | a = democrat 7 161 | b = republican
5	JRip	0.02	0.02	96.55%	3.45%	0.93	0.06	0.17	12.96%	36.00%	a b <-- classified as 257 10 | a = democrat 5 163 | b = republican
6	OneR	0	0.02	95.63%	4.37%	0.91	0.04	0.21	9.21%	42.92%	a b <-- classified as 253 14 | a = democrat 5 163 | b = republican
7	PART	0.02	0.02	97.47%	2.53%	0.94	0.05	0.14	10.08%	29.25%	a b <-- classified as 261 6 | a = democrat 5 163 | b = republican
8	J48	0	0.02	97.24%	2.76%	0.94	0.05	0.15	10.95%	30.93%	a b <-- classified as 261 6 | a = democrat 6 162 | b = republican
9	LMT	0.56	0.02	96.32%	3.68%	0.92	0.11	0.18	22.48%	37.81%	a b <-- classified as 259 8 | a = democrat 8 160 | b = republican
10	RandomTree	0	0.02	99.31%	0.68%	0.98	0.01	0.06	2.72%	12.79%	a b <-- classified as 266 1 | a = democrat 2 166 | b = republican

Table IX Accuracy of vote dataset

TestMode	Low accuracy	High accuracy
TrainingSet	NaiveBayes	RandomTree
Crossfolds 10	NaiveBayes	J48
Percentage split	NaiveBayes	AdaBoostM1

I found RandomTree, LMT and AdaBoostM1 and are considered highest Performance classification accuracies wheras NaiveBayes considered lowest performance classification for this observation in the three test modes.

IV. DISCUSSION

The results may differ if the different data set is used or another strategy is applied, which may be tested. There may be other determinants which may not be present in the dataset or may be overlooked by the authors while experimenting. Based upon the specific dataset, with above analysis and methodology I can have more specified results, which can be used for various strategic decisions. . I ended up using the ten classification algorithms. Naivebays It gave less accurate predictions, whereas LMT and RandomTree are the best classifiers in accurate predictions.

RandomTree has given best prediction results in all types of datasets with Training Set mode. This thesis suggested that RandomTree could classify even large datasets efficiently and could be used in different domains.

The second experiment also depicted more or less same result for both datasets "german.c" and "ionosphere" with respect to training set test mode, showed that the RandomTree had highest accuracy performed well with 100% correctly classified instances as shown in the Table IV and Table VI, from this experiment it has been observed best in the view of least time taken to build model.

In the final experiment observed the performance of both datasets "German Credit" with respect to the test mode as Cross-validation and "vote" with respect to the test mode as percentage split 66% are showed the results more or less very close in their performance highest and least accuracies. Remaining algorithms had showed performance as moderate.

Hopefully, however, some of any future work might find these results helpful when exploring the range of classification algorithms available in WEKA.

V. CONCLUSIONS

For each characteristic, I observed how the results vary whenever test mode is changed. My measure of interest includes the analysis of classifiers on different datasets, the results are described in value of Time taken to build model, correctly classified instances, incorrectly classified instances kappa statistic, mean absolute error, root mean squared error, relative absolute error, root relative squared error and confusion matrix, after applying the training set or cross-validation or Percentage split method. A total of ten classification algorithms have been used in this performance study. These classifiers in WEKA have been categorized into different groups such as rule based (OneR, PART and JRip), tree based (RandomTree, J48), function (SMO, LMT), bayes (NaiveBayes) and Lazy (KStar, AdaboostM1) , are evaluated on three different datasets such as (Vote) have nominal class value,(German-g) have string class value and (Ionosphere) have numeric class value.

There's a few more variables to considered before making the final decision, but from the performance seen in earlier chapters, the proposed solution for how researchers should tackle the problem of classifying structured data in there data sets is to implement a solution. The reason why RandomTree is proposed instead of the other two candidates AdaBoostM1 and LMT that also managed to reach the goal of a positive classification 100% percentage three times , whereas LMT classification percentage perform 75.90 % and 77.06 %.

Also LMT classifier suffers from taking a long time when constructing a classifier model for the data sets, so the running time to classify data is related to the size of the data set used as model, so a big data set will make this classifier unusable if a user is actively awaiting the result from the classification due to the time complexity involved ,whereas RandomTree being the total opposite of since it requires close to no time at all to set up the classifier model to store the whole data set in the memory. For future works, it would be interesting to see if the tree based algorithms such as RandomTree could keep the high performance seen in this thesis if the data set was extended with a high number of different classes. Extending this work by including other classification algorithms like SGD, M5Rules, Bagging, etc., and prediction through classification using other large data sets can be interesting. Another future direction can be testing with data sets of different domains other than standard UCI repository that can be from a new real life data or obtained from survey on different domains.

ACKNOWLEDGMENT

I would like to thank all my supervisors who helped me out a lot in this paper to complete in a good way.

REFERENCES

[1] Ian H. Witten, Eibe Frank , Data Mining: Practical Machine Learning Tools and Techniques, 2005 by Elsevier Inc..
[2] Deepali Kharche, K. Rajeswari, Deepa Abin, SASTRA University, Comparison of different datasets using various classification techniques with WEKA, Vol. 3, Issue. 4, April 2014.
[3] *http://www.tutorialspoint.com/data_mining/dm_classification_prediction.htm*
[4] http://weka.sourceforge.net.
[5] *http://en.wikipedia.org/wiki/Weka_(machine_learning)*

