Performance Evaluation and Comparison of Classification Techniques for Outcome Estimation in Strategic Board Games

Maryam Gulzar†, Arshad Ali and Basharat Naqvi††,

The Department of Software Engineering, The University of Lahore, 55150, Pakistan
The Department of Computer Science & Information Technology, The University of Lahore, Lahore, 55150, Pakistan
Education Department, The Government of Punjab, Pakistan

Summary
Supervised learning aims to construct a distribution model for class labels with respect to features of prediction. Various machine learning approaches have been developed to analyze classification technique on different kinds of data.
The objective of this work is to evaluate and compare the prediction performance of various classification techniques on 3 datasets belonging to strategic board games. This comparison analysis is done by using WEKA, open source software, which is responsible for implementing variety of machine learning algorithms for data-mining application i.e. classification.
This work provides basic overview of selected machine learning classification models alongwith a brief description of datasets of three strategic board games. Then, it evaluates and compares the prediction performance of various classifiers using K-fold cross validation test mode. The results are based on several evaluation metrics like accuracy, precision, recall, kappa statistics, mean absolute error, and root mean squared error. Finally, it provides the best classification method for outcome prediction in strategic board games. Tree based LMT, SVM based SMO and K-NN based LBk are observed as the most suitable models for outcome prediction of strategic board games, LMT being the most influential one.

Key words:
Classification, evaluation metric, machine learning, prediction, strategic board games

1. Introduction

Data mining is defined as the procedure of extracting useful information from a dataset [1]. It refers to extracting or mining knowledge from large amounts of data [2]. Many disciplines including biology, engineering, commerce, physics, health, communication networks and so on can benefit from data mining solutions which can be applied to these areas effectively.
Classification is one of the two forms of data analysis for the purpose of extracting models which describe important classes or predict future data trends. It is supervised version of machine learning. Classification models predict categorical class labels, for example, a classification model can be built for bank loan application when a bank loan officer wishes to analyze the data for the purpose of knowing customer credibility as either safe or risky. Similarly, a classification models can be applied to predict the outcome of a game between two players in terms of a win, a loss or a draw for a player.
Classification approach has been applied in many fields like engineering, management, medicine and so on. Classification algorithms attempt to analyze problems of diverse nature such as diagnosis of diseases, image recognition and so on. Bayes network classifiers (i.e. Bayes Net and Naïve Bayes), neural network classifier (i.e. multilayer perceptron), support vector machine (i.e. SMO), K-nearest neighbor (i.e. LBk), tree-based classifier (i.e. LMT), and ensemble model (i.e. random forest) are most influential and well-known classification techniques [2].
This work provides the comparison results of classification models by using various performance evaluation criteria. The experiments are done by using WEKA, open source software tool, with three binary classification datasets of strategic board games. Any dataset is structured as M x N data matrix, say X. In this structure, each row, M, shows an instance or record whereas each column, N, consists of the value of attribute for each record or instance.
In a board game, pieces are placed on an already marked board under set of rules. Some of the board games are purely based on strategy and skill of a player i.e., chess, many involve some probability of chance, and children's games are totally based on chance without any involvement of skill or decision.
Prediction performance of classifiers is done by using K-fold cross validation evaluation test mode [3]. The training set and holdout are other evaluation test modes. We have chosen K-fold cross validation mode due to its ability of overcoming the issue of other two modes. It splits observations randomly into a number of n folds. The first n-1 folds are used for training purpose while the remaining one is used for testing purpose. Then, it works by considering a different fold for testing in rotation. Some theoretical evidence back up that 10 is the right choice for number of folds in order to obtain best estimates [4].
Therefore, in this work, we have used 10 folds, so 10 different evaluations results are obtained and the overall performance is the average of the 10 evaluation results.
Focus: This work considers datasets ranging from small to large size belonging to strategic board games. Various

Manuscript received July 5, 2018
Manuscript revised July 20, 2018

classification techniques have been applied for evaluation by using evaluation metrics such as accuracy, recall, precision, kappa statistics, mean absolute error (MAE) and root mean squared error (RMSE). The performance comparison of classifiers is made in terms of evaluation metrics. This study focuses on the following research questions:

- Can various machine learning classification models successfully classify outcome in strategy games?
- Which classifier is better option in classifying outcome of strategic board games?

Organization of the Paper: The rest of the paper is organized as follows. Section 2 describes selected datasets and provides an overview of classification techniques. Section 3 describes evaluation metrics used and results as well. Section 4 discusses the comparison of various classifiers. Finally, Section 5concludes the work.

2. Datasets and Classification Models

This section, presents description about three datasets which are selected for the purpose of evaluation and comparison of classification models. Then, a brief overview of various classification techniques used to analyze the selected datasets is provided.

Evaluation of classifiers is done on chosen datasets by using WEKA, an open software tool. This tool was developed in New Zealand by the University of Waikato team, implements data mining techniques in JAVA language. This open source tool is issued under General Public License [5]. It possesses not only the ability for developing machine learning algorithms but also provides their application in real data mining issues. This tool includes algorithms for preprocessing classification of data, regression, clustering, association rules and it includes tools for the purpose of visualization as well. The format of data file processed by WEKA is ARFF consisting of distinct tags which specify different stuffs such as attribute names, attribute types, attribute values and data in the data file. Explorer, consisting of a set of panels, is the main interface of WEKA. These panels are used for doing a certain task. Other panels in the Explorer are used for the purpose of further analysis of data provided dataset has been loaded. A detailed description of this software tool is provided in [6].

2.1 Datasets

This work considers three datasets of strategic board games for performance evaluation and comparison of various classification methods. The datasets are categorized as small, medium and large on the basis of number of instances in each dataset. For example, dataset with instances less than or equal to 1000 is put in small category. Dataset having number of instances more than 1000 and upto 5000 is categorized as medium whereas dataset with more than 10000 instances is considered in large category. Table I below depicts the major details about datasets which are taken from UCI Machine Learning Repository [7] for the purpose of comparison of classification techniques. Table 1 provides information about chosen dataset. In Table 2, we provide abbreviations used more frequently throughout the paper.

Table 1: Datasets of Strategy games for classification comparison

Datasets	***Attributes***	***instances***	***Scale***
Tic-Tac-Toe	10	958	Small
Chess	37	3196	Medium
Connect4	43	67557	Large

Table 2: Abbreviations

Description	***Abbreviation***
Multilayer perceptron	MLP
Rando forest	RF
Sequential minimal optimization	SMO
Logistic Model Trees	LMT
Support vector machine	SVM
Mean Absolute Error	MAE
Root Mean Squared Error	RMSE

Tic-Tac-Toe is a well-known online game which is played on a 3x3 grid between two players. A player who occupies three grids in a row, a column or a diagonal is the winner. In this game, one player is not allowed to win all the time as many games results in a draw as well. So the best strategy for a player is not to lose the game [8]. Tic-Tac-Toe, a small scale dataset, has 9 attributes other than prediction attribute and contains 958 instances which encode all possible board configurations at the end of tic-tac-toe. It is an artificial dataset without any redundant instances. Prediction attribute for player "x" is win or no win. Win for player X means "X" has acquired three-in-a-row places (i.e. horizontally, vertically or diagonally). Figure 1 depicts a "win" for player X on account of three X diagonally. In this game, there are possibly 255,168 games in total, however; only 958 terminal configurations are achieved where a winner is observed. [9].

X	O	X
O	X	O
	O	X

Fig. 1 A Tic-Tac-Toe game won by X

Chess is supposed to be the kind of all strategic board games. Chess, a medium scale dataset, consists of 37 attributes including decision attribute and 3196 instances. Each instance is a board-description for chess endgame. Attributes 1-36 describe the board while the last attribute is the classification "win" or "nowin". In this dataset, prediction job is determined in order to find that weather white earns a win or not.

Connect4, a large scale dataset, comprises of 67557 instances and 43 attributes. First 42 attributes correspond to one connect4 square. These attributes contain the information as {X ,O , B}. When player “x” takes any position, attribute information is marked as x. Similarly, when player O takes any position, attribute information is marked as O. In case a square is blank i.e. no player has taken that position, then attribute information is marked as B. Prediction attribute is taken as “win” or “nowin” for player “X”. This dataset contains all legal 8-ply positions in Connect4 game.

For each of these datasets, various classification techniques are applied, the details of which are provided in sub-section 1.2.

2.2 Classification Techniques

Bayes Network Classifiers

There are many Bayesian network classifier learning algorithms which are implemented in WEKA [10]. Research community paid a considerable attention to Bayesian networks use for classification. For a given vector of attributes, the aim of Bayesian classifier is to correctly estimate the value of a chosen discreet class variable [11]. In present work, BayesNet and Naïve Bayes classifiers have been considered for evaluation purpose on selected datasets. In this work, we experimented with both Bayes network classifier and found that they offer similar results. Therefore, the results are shown only for Naïve Bayes approach. The authors of [12] provide detailed working of this approach.

Neural Network Classifier s

A neural network comprises of neurons which are responsible for converting an input vector into some output. These neuros are arranged in layers. Each neuron is gets an input and often applies a nonlinear function on input in order to produce the output which is passed to the next layer. In these networks, a neuron provides its output to all next layer neurons. There is no mechanism of provision of feedback to the previous layer. Thus, these networks are usually considered as feed-forward networks. Some weights are applied to the signals upon going through from one neuron to another and these weights are adjusted in the training period in order to opt the neural network to a specific problem which is its learning phase. Neural network are turned out to be very important classification tool. [13].

In this work, multilayer perceptron (MLP), a neural network classifier, is applied on chosen datasets. MLP is implemented in WEKA under function classifiers [14]. MLP classifies instances by using back-propagation [18] and can be constructed by hand, produced through an algorithm or both. There can be one or more non-linear hidden layers which makes it different from logistic regression. It has the ability to learn non-linear models and online learning of models (real-time learning).

Support vector Machine Classifier

Support vector machine (SVM) have been successfully applied to many applications such as face detection, verification and recognition [15], object detection and recognition [16], prediction [17-19] etc. Platt [20] proposed a Sequential Minimal Optimization (SMO). SMO makes an effort to solve the smallest possible optimization problem at every step. SMO classifier is also applied on chosen datasets and is include in WEKA [21]. The authors of [22] proposed

K-Nearest Neighbor Classifiers

Local learning classifiers select more appropriate patterns for learning purpose from the complete examples set. For each new query test offered to the system, choice is made through similarity measurement of the pattern. Such local classifiers are normally called lazy or instance-based learning algorithms [23]. Examples of lazy classifier are LBk and KStar which were applied to selected datasets in this work.

Tree based Classifier

In this work, we experimented wite J48 and logistic model trees (LMT) models and found that LMT has better performance than J48, therefore, we provided results only for LMT in this category. The idea of LMT was provided by authors of [24].

Ensemble Classifiers

These learning techniques build a set of classifiers for the purpose of classifying new instances by considering their weights for prediction. Bayesian aver-aging, bagging, boosting and random forest are some examples of these algorithms. Random forest operates by constructing a multitude of decision trees at the time of training and providing a class as output that is mode of the classes. Random forest was introduced by Brieman [25].

3. Evaluation Metrics and Results

A binary classifier provides outcome with two labels, for example, in terms of Yes/No and 1/0 against provided data as input. In order to evaluate performance after classification, observed class values of test dataset are compared with those predicted by classifier. Normally, one class is shown as positive (P) and the other one as negative (N). Three strategy games dataset chosen for performance evaluation and comparison in this work consist of two observed labels i.e. “win” or “nowin”. Two players “x” and “o” are considered in all three datasets and target

variable is labeled as "win" when player "x" succeeds while loss or draw for a player "x" is labeled as "nowin".
The confusion matrix is an important metric which helps figure out the accuracy and correctness of the model. For binary observed labels, the table of confusion matrix has two dimensions, namely observed and predicted. The columns show the observed/actual classification while rows provide predicted ones. It is worth mentioning that confusion matrix is not a performance measure in itself, however, it offers foundation for all of the performance measures. Four major items linked with confusion matrix are True Positives (TP), True Negatives (TN), False Positives (FP) and False Negatives (FN). True positives are data points classified as positive by the model that actually are positive (correct classification), TN are data points classified as negative by the model that actually are negative (correct classification), FP are data points classified as positive that actually are negative, and FN are data points the model identifies as negative that actually are positive (i.e. incorrect).
For a model to be 100% accurate, it must provide 0 FPs and 0 FNs, but this kind of scenario does not exist in real life. Every model being used for prediction of true class of the target attribute has some errors associated with it. The authors of [26] discussed various performance evaluation metrics.

3.1 Accuracy

The accuracy of a classification model refers to the number of accurate predictions made divided by the number of total predictions. Accuracy of any model is actually the number (or %) of correctly classified instances (CCI). In all datasets, target class is nearly balanced; therefore, accuracy is a good measure to be used.

$$CCI = \frac{\mathrm{TP+TN}}{\mathrm{TP+TN+FP+FN}} \tag{1}$$

CCI (in percentage) of various classification models applied on three strategy games dataset is provided in Table 3.

3.2 Recall or sensitivity

It refers to the number of true positives divided by the number of true positives plus the number of false negatives. It is the TP rate and also referred to as sensitivity. Recall (REC) basically indicates what proportion of instances is identified as "win" for player "x" by the model that actually were "win". It is preferred to have recall as close to 100% as possible for the purpose of having more focus on minimizing FNs.

$$REC = \frac{\mathrm{TP}}{\mathrm{TP+FN}} \tag{2}$$

Value of Recall (in percentage) of various classification models applied on three strategy games dataset is provided in Table 4.

Table 3: % Values of Accuracy metric

Name of algorithm	***Name of Strategic board game***		
	Tic-Tac-Toe	***Chess***	***Connect4***
naïve-bayes	69.62	87.89	76.30
mlp	96.87	99.34	84.72
smo	98.33	95.43	81.92
LBk	98.75	96.28	85.58
lmt	98.23	99.75	87.94
random forest	97.39	99.09	87.43

Table 4: % Values of Recall metric

Name of algorithm	***Name of Strategic board game***		
	Tic-Tac-Toe	***Chess***	***Connect4***
naïve-bayes	83.90	89.50	90.80
mlp	97.60	99.40	87.50
smo	100	95.60	85.10
LBk	100	99.0	96.50
lmt	99.80	99.90	91.70
random forest	99.40	99.40	94.70

3.3 Precision

Precision is defined as the number of true positives divided by the number of true positives plus the number of false positives. Precision (PREC) is basically represents the proportion of the data points predicted as positive by the model which were actually positive, while recall expresses the ability to find all relevant instances in a dataset. Precision is also referred to as Positive Predictive Value (PPV). It is better to make precision as close to 100% as possible in order to be more focused towards minimizing FPs.

$$PREC = \frac{\mathrm{TP}}{\mathrm{TP+FP}} \tag{3}$$

Value of precision (in percentage) of various classification models applied on three strategy games dataset is provided in Table 5.

Table 5: % Value of precision

Name of algorithm	***name of Strategic board game***		
	Tic-Tac-Toe	***Chess***	***Connect4***
naïve-bayes	73.40	87.60	77.20
mlp	97.60	99.30	85.90
smo	97.50	95.60	83.50
LBk	98.10	94.10	85.10
lmt	97.50	99.60	90.80
random forest	96.70	98.90	87.30

3.4 Other metrics

Other significant measures are kappa statistics, MAE, and RMSE.
Kappa statistics provides statistical significance of the model i.e. prediction agreement with the actual label. Its value as 1.0 indicates complete agreement between

predicted and actual values. Table 6 provides values of kappa statistics.

Table 6: % Values of kappa statistics

Name of algorithm	***Name of Strategic board game***		
	Tic-Tac-Toe	***Chess***	***onnect4***
naïve-bayes	0.2843	0.7571	0.4265
mlp	0.9309	0.9868	0.6872
smo	0.9627	0.9085	0.630
LBk	0.9721	0.9252	0.6821
lmt	0.9604	0.995	0.7194
random forest	0.9417	0.9818	0.709

4. Comparative Analysis

This section provides comparison of various classifiers in terms of accuracy for three datasets. Figure 2 shows the comparative results w.r.t. accuracy, recall and precision for Naïve-Bayes, MLP, LBk, LMT and random forest on tic-tac-toe dataset. It is observed that support vector machine based SMO and K-NN based LBk perform better as compared to other classifiers. Moreover, SMO has a slight edge over LBk by a narrow margin of 0.42%. Naïve-Bayes performs worse than all other classifiers in classifying instances correctly. Again, performance of SMO and LBk in term of recall or sensitivity is 100% and Naïve-Bayes model is again has lowest performance among the six models. However, in terms of precision, neural network based MLP performs slightly better than close competitors i.e. SMO and tree based LMT.

Figure 3 presents performance of classification models in terms of kappa, statistics, MAE and RMSE. It is evident from kappa bar that LBk has best significant prediction agreement with the actual class label. Then, SMO has close performance to that of SMO. However, an interesting observation is that SMO outperforms all classifiers by lowering MAE and RMSE (refer to Figure 3). These evaluation measures strengthen the argument that the performance of Naïve-Bayes model is less than all other models.

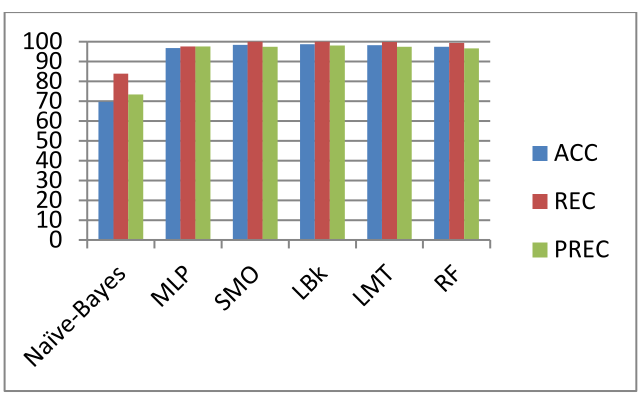

Fig. 2 Accuracy, recall and precisin results of classifiers for Tic-Tac-Toe

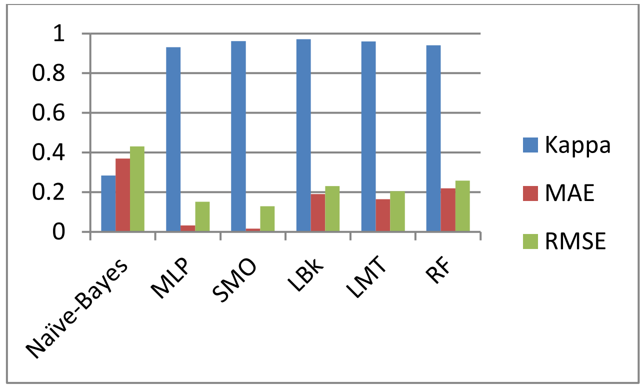

Fig. 3 Kappa, MAE and RMSE results of classifiers for Tic-Tac-Toe

Figure 4 shows the comparative results w.r.t. accuracy, recall and precision for Naïve-Bayes, MLP, LBk, LMT and random forest on chess dataset. It is observed that tree based classifier i.e. LMT performs better than all other models in terms of these three evaluation metrics i.e., ACC, REC and PREC. Figure 5 presents performance of classification models in terms of kappa, statistics, MAE and RMSE for chess dataset which confirms the superiority of LMT over other classifiers by showing that LMT has highest kappa statistics value while its MAE and RMSE are lower than all other models. The performance of MLP and RF is very close to LMT as evident from Figures 4-5. Again, a close look at these figures shows that Naïve-Bayes model's performance is lowest in terms of selected evaluation metrics.

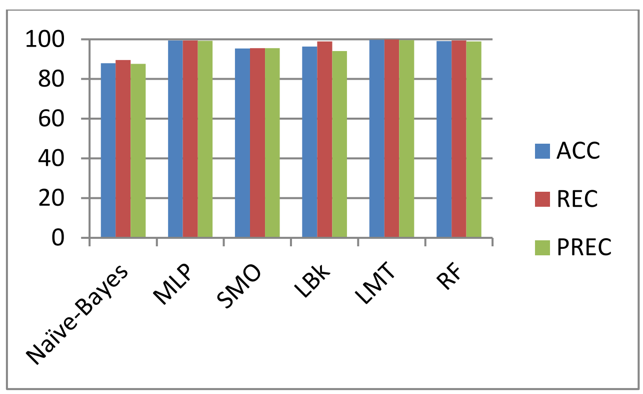

Fig. 4 Accuracy, recall and precisin results of classifiers for Chess

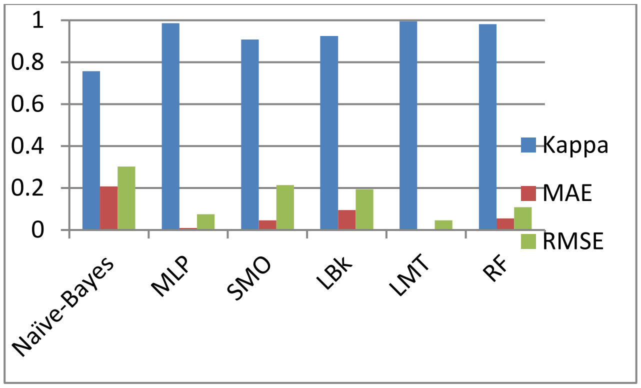

Fig. 5 Kappa, MAE and RMSE results of classifiers for Chess

Figure 6 shows the comparative results w.r.t. accuracy, recall and precision for Naïve-Bayes, MLP, LBk, LMT and random forest for Connect4 dataset. The figure shows that LMT has best performance in terms of accuracy and precision, while LBk has better performance in terms of recall. For this dataset, Naïve-Bayes model provides lowest performance in terms of accuracy and precision, but it performs better than SMO and LMP in terms of recall. Figure 7 presents performance of classification models in terms of kappa, statistics, MAE and RMSE for Connect4 dataset which confirms the superiority of LMT over other classifiers by showing that LMT has highest kappa statistics value while its MAE and RMSE are lower than all other models. Again, a close look at Figures 6-7 shows that Naïve-Bayes model's performance is lowest in terms of selected evaluation metrics.

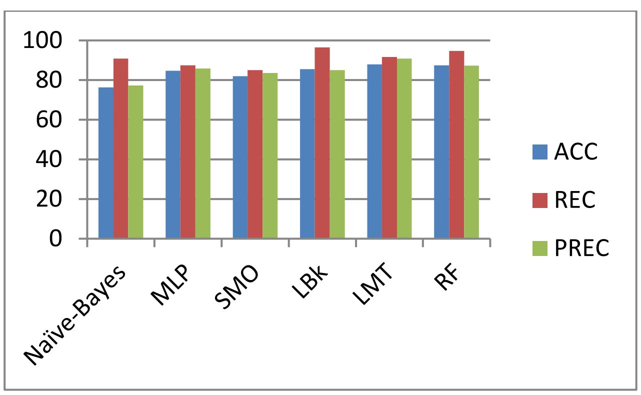

Fig. 6 Accuracy, recall and precisin results of classifiers for Connect4

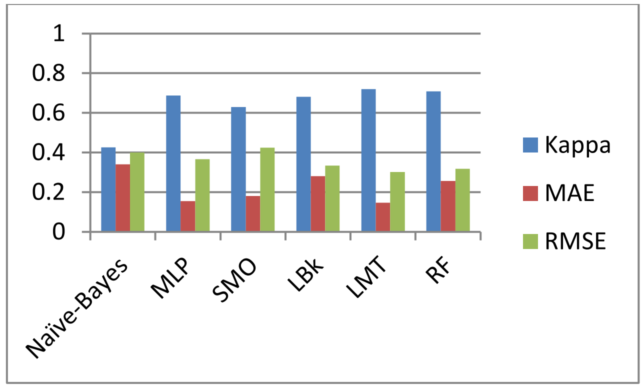

Fig. 7 Kappa, MAE and RMSE results of classifiers for Connect4

In summary, logistci model tree, a tree based model, is found to be the best model keeping in view all three datasets. For Tic-tac-toe, LBk was observed as the best model, however, perfomrnac e of LMT was very close to LBk. For medium and large scale datasets, LMT performed better than all other classifeirs by some margin in terms of all evaluation metrics.

5. Conclusions

This work evaluated and compared the prediction performance of various classification techniques by using datasets of strategic board games. These results were obtained by using WEKA.
Firstly, we provided details about datasets used for this study. Then, an overview of selected classifiers was provided. Several evaluation metrics like accuracy, recall, precision, kappa statistics, MAE and RMSE were considered for performance evaluation of classifiers and K-fold cross validation test mode was applied. Tree based LMT, SVM based SMO and K-NN based LBk are observed as the most suitable models for outcome prediction of strategic board games, LMT being the most influential one.
The birth of social media opens up new opportunities for researchers in the field of data mining and retrieval of information [27]. As a future work, we aim to investigate suitable classification models for prediction of sentiments based on social media contents.

Acknowledgments

This work is mainly based on datasets taken from UCI machine learning repository [5].

References

[1] M. Kantardzic, Data Mining: Concepts, Models, Methods, and Algorithms, IWiley, IEEE, 2011

[2] Han, J., Pei, J., & Kamber, M. (2011). Data mining: concepts and techniques. Elsevier.
[3] Picard, R. R., & Cook, R. D. (1984). Cross-validation of regression models. Journal of the American Statistical Association, 79(387), 575-583.
[4] Witten, I. H., Frank, E., Hall, M. A., & Pal, C. J. (2016). Data Mining: Practical machine learning tools and techniques. Morgan Kaufmann.
[5] WEKA at http://www.cs.waikato.ac.nz/~ml/WEKA
[6] Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., & Witten, I. H. (2009). The WEKA data mining software: an update. ACM SIGKDD explorations newsletter, 11(1), 10-18.
[7] Asuncion, A. & Newman, D.J. (2007). UCI Machine Learning Repository [http://www.ics.uci.edu/~mlearn/MLRepository.html]. Irvine, CA: University of California, School of Information and Computer Science
[8] Bhatt, A., Varshney, P., & Deb, K. Evolution of no-loss strategies for the game of Tic-Tac-Toe. IIT, Kanpur, Department of Mechanical Engineering, KanGAL Report Number 2007002.
[9] Schaefer, S. (2002). Tic-Tac-Toe (Naughts and Crosses, Cheese and Crackers, etc). Mathematical Recreations.
[10] I.H. Witten, E. Frank. Data mining: Practical machine learning tools and techniques with Java implementations. Morgan Kaufmann, 2000
[11] Grossman, D., & Domingos, P. (2004, July). Learning Bayesian network classifiers by maximizing conditional likelihood. In Proceedings of the twenty-first international conference on Machine learning (p. 46). ACM.
[12] Friedman, N., Geiger, D., & Goldszmidt, M. (1997). Bayesian network classifiers. Machine learning, 29(2-3), 131-163.
[13] Zhang, G. P. (2000). Neural networks for classification: a survey. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 30(4), 451-462.
[14] Pal, S. K., & Mitra, S. (1992). Multilayer Perceptron, Fuzzy Sets, Classifiaction.
[15] Jonsson, K., Kittler, J., Li, Y. P., & Matas, J. (2002). Support vector machines for face authentication. Image and Vision Computing, 20(5-6), 369-375.
[16] Gao, D., Zhou, J., & Xin, L. (2001). SVM-based detection of moving vehicles for automatic traffic monitoring. In Intelligent Transportation Systems, 2001. Proceedings. 2001 IEEE (pp. 745-749). IEEE.
[17] Van Gestel, T., Suykens, J. A., Baestaens, D. E., Lambrechts, A., Lanckriet, G., Vandaele, B., ... & Vandewalle, J. (2001). Financial time series prediction using least squares support vector machines within the evidence framework. IEEE Transactions on neural networks, 12(4), 809-821.
[18] McKay, D., & Fyfe, C. (2000). Probability prediction using support vector machines. In Knowledge-Based Intelligent Engineering Systems and Allied Technologies, 2000. Proceedings. Fourth International Conference on (Vol. 1, pp. 189-192). IEEE.
[19] Mitra, P., Murthy, C. A., & Pal, S. K. (2000). Data condensation in large databases by incremental learning with support vector machines. In Pattern Recognition, 2000. Proceedings. 15th International Conference on (Vol. 2, pp. 708-711). IEEE.
[20] Platt, J. C. (1999). 12 fast training of support vector machines using sequential minimal optimization. Advances in kernel methods, 185-208.
[21] Hameed, M., Tahir, F., & Shahzad, M. A. (2018). Empirical comparison of sentiment analysis techniques for social media. International Journal of Advanced and Applied Sciences, 5(4), 115-123.
[22] Gashti, M. Z. (2017). A novel hybrid support vector machine with decision tree for data classification. International Journal of Advanced and Applied Sciences, 4(9), 138-143.
[23] Galván, I. M., Valls, J. M., García, M., & Isasi, P. (2011). A lazy learning approach for building classification models. International journal of intelligent systems, 26(8), 773-786.
[24] Landwehr, N., Hall, M., & Frank, E. (2005). Logistic model trees. Machine learning, 59(1-2), 161-205.
[25] Breiman, L. (1994). Heuristics of instability in model selection. Technique Report. Statistics Department. University of California at Berkeley.
[26] Altaher, A. (2017). Hybrid approach for sentiment analysis of Arabic tweets based on deep learning model and features weighting. International Journal of Advanced and Applied Sciences, 4(8), 43-49.
[27] Cardoza, C., & Wagh, R. (2017). Text analysis framework for understanding cyber-crimes. International Journal of Advanced and Applied Sciences, 4(10), 58-63.

Maryam Gulzar received her BSCS degree from CS&IT Department of the University of Lahore (UoL), Pakistan in 2016. She recently completed her MS(SPM) study from CS Department of National University of Computer and Emerging Sciences (NUCES), Lahore campus, Pakistan in 2018. Currently, she is working as Junior Lecturer at Software Engineering Department of UoL since 2016 soon after completion of her BSCS degree. Her research interests include software metrics measurements and machine learning.

Arshad Ali received the BSc degree with Mathematics and Physics from Punjab University, Lahore, in 1997. In 2003, he earned the MSc degree in computer science from Punjab University, Lahore. He received a master diploma in information technology (Informatique) with speciality in mobile networks in 2009. He received the PhD degree in information technology (telecommunications and electronics) jointly from the Institute of Telecom SudParis and UPMC (Paris VI). Between 2005 and 2008, he served as an audit officer in the office of Auditor General of Pakistan. He was a post-doctoral researcher at Orange Labs, Paris. Currently, he is working as an assistant professor in the Computer Science Department at the University of Lahore, Pakistan. His research interests are in the areas of delay/disruption tolerant networks, wireless mobile ad hoc networks, network coding, Software metrics and supervised learning.

Basharat Naqvi received BSCS degree from Punjab University College of Information Technology (PUCIT), The Punjab University, Lahore, Pakistan in 2011. He obtained his MS (CS) degree from CS&IT Department of The University of Lahore (UoL), Lahore, Pakistan. He has been working as Secondary School Teacher at Schools Education Department, Govt. of Punjab, Pakistan since 2011. His research interests include social network analysis, data mining and machine learning.