Minority Resampling Based Ensemble Framework Using Enhanced Early Drift Detection Method For Imbalanced Data Streams

Priya S (priyas3@srmist.edu.in)
SRM Institute of Science and Technology https://orcid.org/0000-0003-2906-8124

Annie Uthra
SRM Institute of Science and Technology

Research

Keywords: Class Imbalance, Concept Drift, Resampling, Classification, Data streams.

Posted Date: January 12th, 2021

DOI: https://doi.org/10.21203/rs.3.rs-141880/v1

MINORITY RESAMPLING BASED ENSEMBLE FRAMEWORK USING ENHANCED EARLY DRIFT DETECTION METHOD FOR IMBALANCED DATA STREAMS

[1]S.Priya*, [2]R.Annie Uthra
[1]priyas3@srmist.edu.in,[2]annieuthra@gmail.com
[1,2] Department of Computer Science and Engineering, College of Engineering and Technology,
SRM Institute of Science and Technology, SRM Nagar,
Kattankulathur, Chennai

Abstract:

As the data mining applications are increasing popularly, large volumes of data streams are generated over the period of time. The main problem in data streams is that it exhibits a high degree of class imbalance and distribution of data changes over time. In this paper, Timely Drift Detection and Minority Resampling Technique (TDDMRT) based on K-nearest neighbor and Jaccard similarity is proposed to handle the class imbalance by finding the current ratio of class labels. The Enhanced Early Drift Detection Method (EEDDM) is proposed for detecting the concept drift and the Minority Resampling Method (KNN-JS) determines whether the current data stream should be regarded as imbalance and it resamples the minority instances in the drifting data stream. The K-Nearest Neighbors technique is used to resample the minority classes and the Jaccard similarity measure is established over the resampled data to generate the synthetic data similar to the original data and it is handled by ensemble classifiers. The proposed ensemble based classification model outperforms the existing over sampling and under sampling techniques with accuracy of 98.52%.

Keywords: Class Imbalance, Concept Drift, Resampling, Classification, Data streams.

Introduction:

The amount of data generated by smart devices, social media, and from all kinds of sensors has increased with the rapid growth of information technology. Different sources generate an ordered sequence of instances with high speed, named as streaming data. Stream data analysis innovates the business ideas and provides decision-making support using data mining approaches [1]. The streaming analysis is an extraction of sequential patterns from streams of data. The big data streaming analysis has to cope-up with the three features such as volume, velocity, and variety [2][3]. An integral dimension of streaming data features is time, and so the stream data must be analyzed promptly. Thus, the data mining algorithms are essential to capture and mine the stream data record by record in a realistic time. The streaming data analysis includes online and offline techniques. Even though most of the sources generate the data streams in the non-stationary environment, the traditional offline works build the data mining algorithms under the hypothesis that the data will be static. The offline data mining algorithms include data collection, data mining on static data sets, interpretation, and evaluation of result. However, the offline algorithms exert a lot of unresolved challenges in data storage and query processing due to huge volume and variety of data. The online learning algorithms are introduced to cope-up with this problem.

Data streams are massive, continuously changing, timely ordered, and possibly infinite in length. In recent years, mining and analyzing the data streams is very crucial and it has gained attention over the past decade. Some of the real-world applications for class imbalance data are in anomaly detection [4], diagnosis of fault [5], medical diagnosis [6], detections of credit card fraudulent transaction [7], oil spill detection from radar images [8], biological sequence detection [9], spam filtering [10] and many others. In the binary classification problem, the rare class is considered to be a positive class, which occurs very infrequently and the negative class is considered as majority. When more number of positive class examples is there in the dataset, it can significantly impair the learner capacity to learn the positive class. This makes the classifier to learn the class boundary in reality. In the classification problem, for a static dataset the instances will be generated from a specified generating function, which belong to some distribution and the learner tries to learn the concept from the static dataset and this is not valid in the streaming data model. The swift in data distribution is called concept drift and the imbalance distribution makes many conventional machine learning algorithms to be less efficient, particularly in predicting examples of minority groups. Hence, the data stream classification in the existence of concept drift and class imbalance is very challenging.

In a data stream application, the data arrives continuously and learning can be done either by using batch or online method. In batch learning, the data samples are processed in chunks at each time step whereas in online learning model, the data samples are processed incrementally and the classifier model will be updated once the samples are received. The online learning can build and update the classification model to accommodate the new instance of data sample and it can preserve its performance on the previous instance, giving the way for stability-plasticity problem [11]. The main task in imbalanced learning is that, completely under-represented class cannot gain equivalent interest to the classification algorithm. An online imbalance learning system needs to be developed to increase the accuracy of the classification in order to solve this issue. When we consider the problem of spam filtering and credit card application, the minority class is more difficult to collect than the majority class. Hence the misclassification of minority class will be very costly. The traditional models aim at maximizing the overall performance, which can lead to the greater chance of predicting the example as majority class, and there is less chance of recognizing the minority class correctly. In real time, it is seen that the accuracy of the majority class is close to 100% and the minority class will have less than 10% accuracy. The consequence of imbalance class was studied on classifiers such as Random Forest [12], Naive Bayes [13], Decision Tree [14] and KNN[15].

The problem of class imbalance can be tackled at data and algorithmic levels. In the data level, techniques such as oversampling the minority class example, under sampling the majority class or resampling technique is applied to make the dataset balanced. The over sampling and under sampling method is simple whereas the most popular method is resampling, which adds or removes the data samples randomly. In the algorithmic level, the training mechanism of the minority class is modified in order to improve the accuracy of the minority class. There are different types of learning methods called cost-sensitive learning, meta-learning and ensemble learning [16]. In the cost sensitive learning, the cost of misclassification with respect to the classes is determined from the cost matrix. The preprocessing mechanism of training data and post processing of the test data is integrated, in a way that the learner is not modified in the case of meta-learning. The ensemble method is the most popular method that is used for handling the

class imbalance. It tries to improve the accuracy of the single classifier by combining multiple classifiers as the base learner and the new classifier will outperform every one of them. In the recent year, the ensemble based classification has provided a better solution for class imbalance problems.

The rest of this paper is organized as follows. Section II describes the related work of class and imbalance and concept drift. Section III explains the proposed model for detecting and handling class imbalance in drifting datastream. Section IV discusses about the experimentation result and Section V describes the conclusions and discusses future possible directions.

2. Related Works:

The existing algorithms for class imbalance and concept drift mainly deals with online or offline learning is studied by the authors in [17]. Using batch or incremental learning, the data streams may be processed and ensemble learners are deemed successful for issues with data stream classification. The data instances are learned one by one without any prior knowledge in the online learning process. From the mining perspective, streaming data analysis has opened many challenges for online learning algorithms that include concept drifts, temporal dependencies, imbalanced class distribution, and limited data storage availability. The concept drift happens when an unforeseen change occurs on streaming data over time. The skewed class distribution over streaming data is known as a class imbalance. The concept drift creates significant challenges in the discovery of sequential patterns for online learning algorithms.

Due to the uneven data distribution, the data streaming applications face the issues of concept drift and imbalanced data distribution. Ensemble learning [18] is widely used for handling imbalanced class. The conventional ensemble member selection algorithms such as bagging, boosting, and random sampling techniques are cost-effective [19]. However, the heuristic based approaches lack proper focus on the performance of ensemble classification over skewed classes. Moreover, the conventional approaches lack in understanding the importance of diversity among ensemble members and impact of minority class on the performance of ensemble classifier. The ensemble member diversity should not be high which have almost no convergence to the new data pattern. Moreover, there is no clear idea on how to determine the optimal number of classifiers and combine those classifiers. It is essential to analyze the relationship between the nature of imbalanced classes and the ensemble size required to handle the concept drift on imbalanced class efficiently. Mostly, the ensemble classification algorithms exploit majority voting combination method. The majority voting concept tends to ensemble failure when the similar classifiers of incorrect results are combined. Therefore, this necessitates that the ensemble diversity must be exploited during the combination phase of ensemble algorithm.

The Uncorrelated Bagging [20] technique is used to learn the concept drifting imbalanced data stream. Dynamic Ensemble Selection-KNN model [21] ranks the accuracy of the base models of ensemble in the decreasing order and it uses the diverse member to form the ensemble. The random under-sampling [22] removes the majority class instances randomly and it removes the biased samples which will hinder the performance of the classifier. Therefore SMOTE [23] is proposed which generates the current minority class instances using the k-Nearest Neighbour. The support vectors are used to generate new samples based on SVM SMOTE [24]. In

ADASYN [25] the number of samples generated will be proportional to existing neighborhood instances. Based on accuracy and diversity, a heuristic replacement strategy is used in Streaming Ensemble Algorithm (SEA) [26]. The class distribution is balanced by the current stationary minority class and the Mahalanobis distance is used as measure in selectively recursive approach (SERA) for selecting the previous minority group that is most similar in the candidate region[27]. The resampling and time-delayed metrics in OOB an UOB [28] can cope with the imbalanced data stream. The combined issue of drift and class imbalance is addressed in two category namely chunk based and online ensemble [29].

3. Timely Drift Detection and Minority Resampling Technique (TDDMRT)

Many stream data mining strategies do not concurrently take into account the class imbalance and the issues of concept drift. However, the accuracy of the minority class is inferior. To deal with these challenges, conventional data and algorithm level methods are exploited. The data level method focuses on changing the training samples to improve the accuracy of ensemble learners. To balance the class distributions, the data level methods generate more samples for minority classes, more than that in minority classes.

Since most approaches exploit random sampling, this often leads to missing important samples and poor classification accuracy. By providing importance to the minority classes, the modified ensemble members include each of the considered groups of examples. This increases the cost of stream data classification. To minimize the cost of ensemble classification without reducing the accuracy, it is essential to fix the optimal size of the ensemble classifier as well as to select the diverse ensemble members. Even though the optimal number of members of an ensemble is determined, an insufficient amount of samples in a class has an adverse impact on the classification accuracy. Thus, the hybrid method is necessary for solving the issues of concept drift over imbalanced data in a cost-efficient manner.

Data imbalance is the most important factor that affects the performance of the model that are developed through the learning algorithms. The data are initially used to train and construct an effective model and its imbalance result in some flaw model that can be biased over the test and real time data [30]. Henceforth, the data imbalancing has to be addressed at the earliest to build a effective prediction model. In the proposed method, a novel scheme of Timely Drift Detection Minority Resampling Technique (TDDMRT) is employed to improve the performance of model.

The datasets are initially acquired from the data perception center and most commonly it was divided into train and test data. The data selected for training is initially fed into the TDDMRT scheme which contains the Enhanced Early Drift Detection Model (EEDDM) and the resampling model based on K-nearest neighbor and Jaccard similarity. The imbalanced data are initially identified along with the instances of misclassification. The extracted features are fed in to the EEDDM model to perform the drift analysis and the drifted imbalanced data from dataset are estimated over the NSL-KDD[31] dataset with labels of bad and normal conditions. The drifted imbalanced data are classified into minority and majority classes. The obtained minority classes are resampled using the KNN technique and the Jaccard similarity measure is established over the resampled data to generate the synthetic data similar to the original data. Thus, the

imbalanced data in the dataset will be balanced and the model for testing is constructed effectively.

3.1 Enhanced Early Drift Detection Model (EEDDM)

The early drift detection model was proposed in [32]. In that work, two different threshold conditions are established to analyze the drift between the points based on error rate and standard deviation. However, in the current model, three different conditions are formed with the threshold values. The data are streamed into the EEDDM through the data stream generators.

$$\text{Distance between errors} = m + 2s \ldots\ldots (1)$$

$$\text{Error = probability of misclassifying } P_m = \left(\frac{WC_i}{N_i} + \frac{WC_j}{N_j}\right) \ldots.. (2)$$

Where *wc* is the wrong or mistaken classification over the split data and N_i and N_j is the size of split dataset

$$\text{Mean distance, } m = \sqrt{{P_{m1}}^2 + {P_{m2}}^2} \ldots\ldots. (3)$$

$$\text{Standard deviation, } s = \sqrt{\frac{P_{mi}(1-P_{mi})}{i}} \ldots\ldots. (4)$$

In the proposed model, the mean (m) of the errors and standard deviation (s) is formulated as the tan h function for the points and are estimated with the equation (5):

$$\text{Distance between errors} = m + 2\tanh s \ldots\ldots\ldots (5)$$

The corresponding maximum distance is given in equation 6:

$$m' + 2\tanh s' \ldots\ldots\ldots (6)$$

Where m' and the s' are the maximum mean and standard deviation over the given points in space. The ratio between the point error and maximum error is estimated in equation 7:

$$\alpha = \frac{m + 2\tanh s}{m' + 2\tanh s'} \ldots\ldots\ldots (7)$$

Based on the value of α, three different drift levels are obtained over the split dataset. The first level in the in-control level for which the value of α is less than 0.5. The second level is the warning level for which the value of α is more than 0.5 and less than 0.90. The third and final condition is out of control level the for which the value of α is more than 0.9.

Algorithm 1: Enhanced Early Drift Detection Model (EEDDM)

1: get the sample y //y belong the data feature or point
2: if y ≤ n
3: Calculate the mean and standard deviation of error between x and y
4: Estimate the distance of error with equation (1)

5: Get the maximum distance of error with equation (2)
6: Estimate the threshold value α with equation (3)
7: else
8: if α more than 0.9
9: "Drift is in out of control level"
10: Add y to the dataset
11: else if α lie in between 0.5 to 0.9
12: "Drift is in warning level"
13: Replace the point y_i with the alternative point y_j
14: else
15: reset the dataset (No drift)
16: end

3.2 Minority Resampling (KNN-JS)

The drifted instances in the dataset are determined and corrected based on the proposed EEDDM. The corrected imbalanced data are classified into minority and majority classes. The minority classes are taken and are resampled through the K-nearest neighbor algorithm [33]. The similarity of the new instances and the existing instances are calculated using the jaccard similarity measure [34] which is given in equation 4:

$$J = \frac{A \cap B}{A \cup B}$$

Where A and B are the new and existing instances of the data. The threshold is set as 0.75 and hence the new instance that exceeds this value is taken for the synthetic data. The process for resampling is carried out until the dataset is completely balanced.

Algorithm 2 : Synthetic Data Construction

1: procedure [SYData;Class_SYData] = TDDMRT (DATA;Class)
2: Threshold = 0.75
3: C = Unique elements in Class
4: for i = 1 to |Class| do . //Number of Elements in Class
5: x = Positions(Class(i) == C)
6: SC{x} = [SC{x}; i] . //Segregate the class
7: end
8: for i = 1 to |C| do
9: y(i) = SC{i}. //To count the elements in each class SC
10: end
11: [a; b] = maximum(y) //a maximum value and b class number corresponding to a
12: Majority_Class = DATA(SC{x}); Majority_Count = a
13: R = {C1, C2,……,C| SC| - {Cb}
14: for i = 1 to |R| do
15: M(i, :) = Mean(DATA(SC{R (i)}, :))

```
16:     ct = 1
17:     while (True) do
18:     SD = Randomly Constructed vector of size (1 _X size(DATA; 2))
19: th = SIMLX(SD;M(i, :))      // Similarity measurement algorithm
20: if th ≥Threshold then
21: Synthetic_Data(ct, :) = SD   // Synthetic Data construction
22: ct = ct + 1
23: if ct > Majority_Count - y(R(i)) then
24: break
25: end
26: end
27: end
28: SYData{R(i)} = Synthetic_Data
29: Class_SYData{R(i)} = C(R(i)) X ones(ct - 1; 1)
30: end
```

3.3 Classifier model

In the present work, the NSL-KDD cup dataset is used for analyzing the effectiveness of proposed TDDMRT. The data in the NSL-KDD cup dataset are corrected for drift and balanced using the EEDDM and random sampling respectively and fed into the ensemble models of Decision tree, naïve bayes and random forest for classification.

4. Experimentation Results and Discussion

The proposed TDDMRT scheme with EEDDM and the resampling based on KNN-JS framework is compared with the existing model over the NSL-KDD cup dataset. The performance of the TDDMRT over data chunk is given in Table 1 and Figure 1.

Table 1. Performance of TDDMRT over data chunk

Split data	Accuracy (%)	Time (s)
Chunk 1	99.51	3200
Chunk 2	97.62	3420
Chunk 3	98.73	3250
Chunk 4	96.56	2810
Chunk 5	98.89	2856
Chunk 6	99.24	2845
Chunk 7	98.7	2690
Chunk 8	98.82	2548
Chunk 9	98.9	2728
Chunk 10	98.25	3613

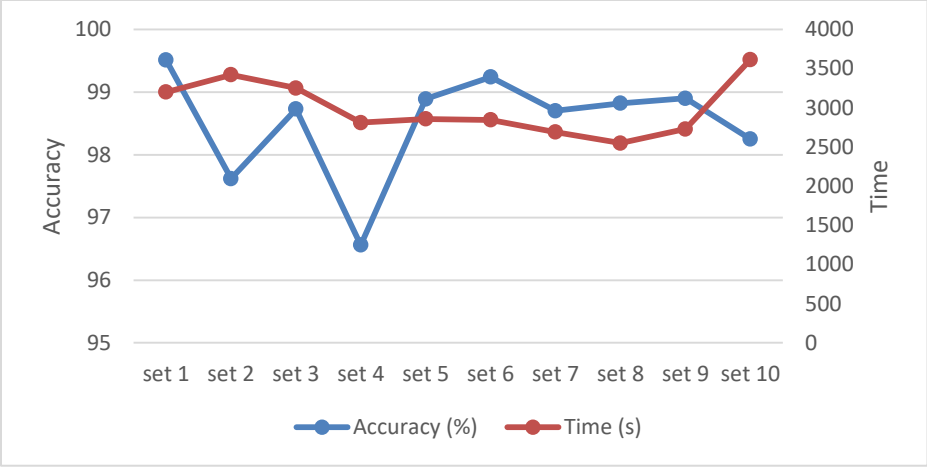

Figure 1. Performance Graph for TDDMRT over data chunk

The comparison for EEDDM is carried out with Dynamic classifier selection based low Accuracy DDM (DCS-LA-DDM) and DDM [35], Semi-Supervised Adaptive Novel Class Detection and Classification (SAND), Accurate Hoeffding trees (AHT) , Streaming classification with Emerging New Class emerging by class Matrix Sketching, EcsMiner, Semi-supervised Adaptive Classification Over data Stream (SACCOS) [36] . The accuracy of the proposed model is about 98.52% that was achieved over the data label of 7% which is better than the Novel DCS-LA-DDM and SACCOS (ANN). However, the SAND-D model showed better accuracy than AHT and SENC-MaS over lesser percentage of labels. The comparison on accuracy and % of label for drift detection over NSL-KDD dataset is given in Table 2 and Figure 2.

Table 2. Comparison of existing drift detection method with proposed EEDDM for NSL-KDD Dataset

Methods	**Accuracy**	**% of label**
DCS-LA-DDM [35]	95.57%	25.06%
DDM [35]	97.89%	100%
AHT [36]	94.74%	100%
ECS MINER [36]	96.77%	100%
SENC-MaS [36]	79.77%	58.26%
SAND-D [36]	95.47%	15.41%
SACCOS (ANN) [36]	97.27%	7.37%
EEDDM (Our Proposed)	98.52%	7%

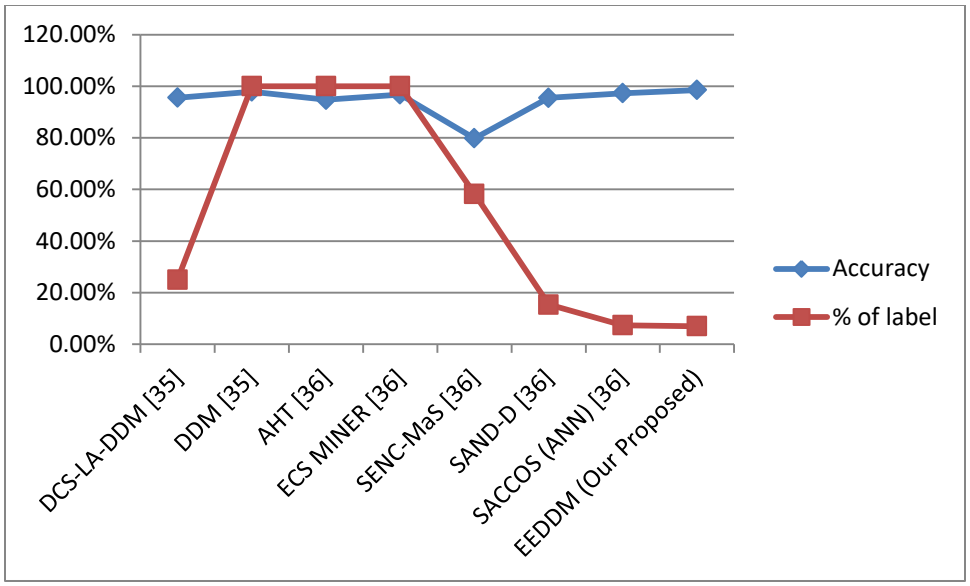

Figure 2. Performance of drift detectors

The Table 3 and Figure 3 showed the comparison on performance over the balanced data that are obtained from oversampling (OS), under sampling (US) and TSSMRT. The accuracy, precision, sensitivity, specificity, detection rate metrics are involved to analyze the classification over the KDD cup data.

The balanced data through the TDDMRT technique showed accuracy of 86.40% in Random forest classifier and it is better than the classification through Naïve bayes and decision tree classifier. Similarly, the performance on detecting the attack in the dataset is about 0.025 for the RF classifier and it is better than the NB and DT classifier over the TDDMRT approach. The sensitivity, precision and specificity of NB classifier is about 0.856, 0.782 and 0.075 respectively are lesser compared to both the RF and DT classifier over the given data.

Table 3. Performance of ML algorithm over the NSL-KDD cup data with resampling technique

ML algorithms	Random Forest			Naïve Bayes			Decision tree		
Parameters	OS	US	TSDD MRT	OS	US	TSDD MRT	OS	US	TSDD MRT
Sensitivity	0.78	0.957	0.968	0.671	0.848	0.856	0.01	0.773	0.982
Specificity	0.888	0.747	0.905	0.769	0.543	0.782	0.999	0.883	0.999
Precision	0.1394	0.08	0.2432	0.063	0.041	0.075	0.7	0.132	0.762
Negative predictive value	0.994	0.998	0.995	0.99	0.993	0.995	0.977	0.994	0.999
Detection rate	0.017	0.021	0.025	0.015	0.019	0.022	0.0002	0.0174	0.022
Balanced Accuracy	0.834	0.852	0.864	0.72	0.695	0.734	0.5051	0.828	0.832

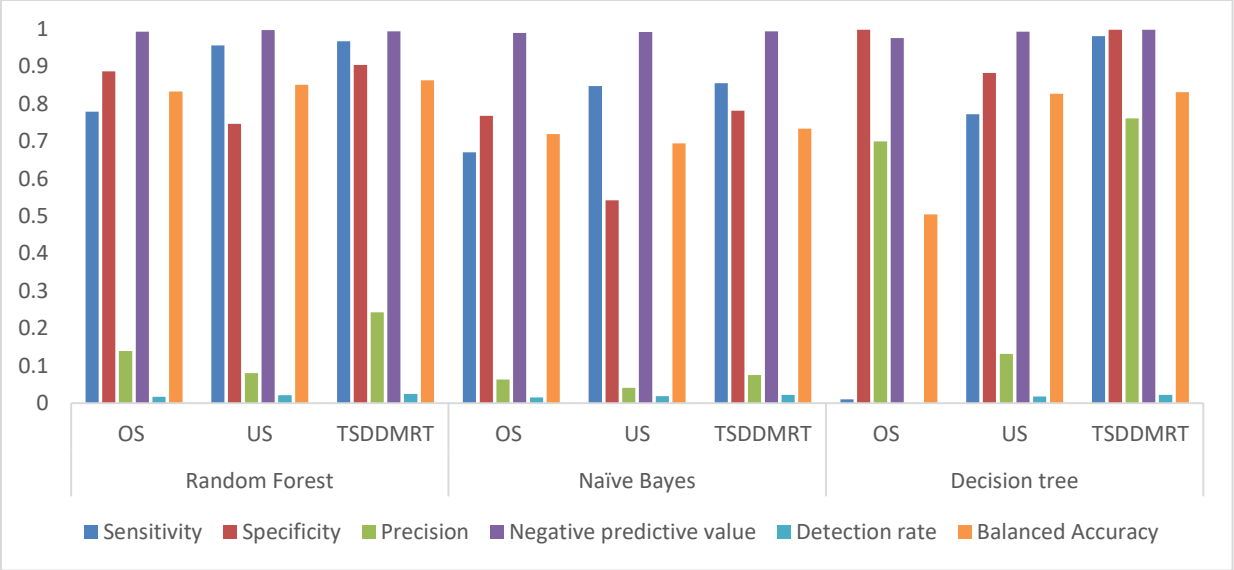

Figure 3: Performance of ML algorithm over the NSL-KDD cup data with resampling technique

Especially for DT classifier, the precision is about 0.762 along with sensitivity and specificity of about 0.982 and 0.999 which is maximum over the different classifiers. The Negative predictive value is about 0.999 for DT classifier and it is about 0.995 over the RF and NB classifier.

5. Conclusion and Future Work

In this paper, a Timely Drift Detection and Minority Resampling Technique (TDDMRT) based on K-nearest neighbor and Jaccard similarity is proposed to handle the combined problem of concept drift and class imbalance. A minority resampling method is applied to handle the class imbalance by finding the imbalance ratio. The Enhanced Early Drift Detection Model (EEDDM) handles the concept drift by analyzing the drift between points based on threshold values. The experimental results shows that the proposed method can handle both concept drift and class imbalance with the accuracy of 98.52% which is better compared with the results of the existing literature. For future work, we plan to apply the drift detection in imbalanced data stream for a multi-class problem.

Declarations

Ethics approval and consent to participate
Not Applicable

Consent for publication
Not Applicable

Availability of data and materials
The data that support the findings of this study are available from the corresponding author upon reasonable request

Competing interests
The authors declare that they have no competing interests.

Funding
This work was not funded

Authors' contributions

Priya was the main contributor of this work. She has done an initial literature review, data collection, experiments, prepare results, and drafted the manuscript. Annie has worked closely with Priya to review, analyze, and manuscript preparation and to improve the final paper. Both the authors have read and approved the final manuscript.

Acknowledgements
The authors would like to thank the SRM Institute of Science and Technology, Department of CSE for providing an excellent atmosphere for researching on this topic.

Authors' information
S.Priya is currently Assistant Professor in the Department of Computer Science and Engineering at SRM Institute of Science and Technology. Her research interests include class imbalance learning, online learning, machine learning and deep learning.

R.Annie Uthra is currently Associate Professor in the Department of Computer Science and Engineering at SRM Institute of Science and Technology. Additionally, she serves as the Adjunct Associate Teaching Professor in the Institute for Software Research in the School of Computer Science at Carnegie Mellon University, Pittsburgh, USA. A graduate of SRM University's Master of Engineering in Computer Science and Engineering program, and has received Ph.D Degree from SRM University. Her research interest includes machine learning, wireless sensor networks, Navigation, Energy Aware Routing Technique.

Reference :

[1] Chen, Min & Mao, Shiwen & Liu, Yunhao. (2014). Big data: A survey. Mobile Networks and Applications. 19. 10.1007/s11036-013-0489-0.
[2] Oguntimilehin, Abiodun & Ademola, Ojo. (2014). A Review of Big Data Management, Benefits and Challenges. Journal of Emerging Trends in Computing and Information Sciences. 5. 433-438.
[3] Fan, W. & Bifet, Albert. (2014). Mining big data: Current status, and forecast to the future. ACM SIGKDD Explorations Newsletter. 16. 1-5.
[4] Tavallaee, Mahbod & Stakhanova, Natalia & Ghorbani, Ali. (2010). Toward Credible Evaluation of Anomaly-Based Intrusion-Detection Methods. Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on. 40. 516 - 524. 10.1109/TSMCC.2010.2048428.
[5] Yang, Z. & Tang, W.H. & Shintemirov, Almas & Wu, Q.H.. (2009). Association Rule Mining-Based Dissolved Gas Analysis for Fault Diagnosis of Power Transformers. Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on. 39. 597 - 610. 10.1109/TSMCC.2009.2021989.
[6] Mazurowski, Maciej & Habas, Piotr & Zurada, Jacek & Lo, Joseph & Baker, Jay & Tourassi, Georgia. (2008). Training Neural Network Classifiers for Medical Decision Making: The Effects of Imbalanced

Datasets on Classification Performance. Neural networks : the official journal of the International Neural Network Society. 21. 427-36. 10.1016/j.neunet.2007.12.031.

[7] S.Priya, Siddharth Agarwal, Pankaj Thakur, Annie Uthra. (2020). Ensemble Based Classification for Class Imbalanced Credit Card Fraudulent Data. International Journal of Advanced Science and Technology, 29(06),2129-2141.

[8] Kubat, M.. (2000). Addressing the Curse of Imbalanced Training Sets: One-Sided Selection. Fourteenth International Conference on Machine Learning.

[9] Al-Shahib, Ali & Breitling, Rainer & Gilbert, David. (2005). Feature Selection and the Class Imbalance Problem in Predicting Protein Function from Sequence. Applied bioinformatics. 4. 195-203. 10.2165/00822942-200594030-00004.

[10] Ratadiya, Pratik & Moorthy, Rahul. (2019). Spam filtering on forums: A synthetic oversampling based approach for imbalanced data classification.

[11] Grossberg, Stephen. (1988). Nonlinear neural networks: Principle, mechanisms, and architectures. Neural Networks. 1. 17-61. 10.1016/0893-6080(88)90021-4.

[12] Gomes, Heitor Murilo & Bifet, Albert & Read, Jesse & Barddal, Jean Paul & Enembreck, Fabrício & Pfahringer, Bernhard & Holmes, Geoff & Abdessalem, Talel. (2017). Adaptive random forests for evolving data stream classification. Machine Learning. 106. 1-27. 10.1007/s10994-017-5642-8.

[13] Grobelnik, Marko. (1999). Feature selection for unbalanced class distribution and Naive Bayes.

[14] Japkowicz, Nathalie & Stephen, Shaju. (2002). The Class Imbalance Problem: A Systematic Study. Intell. Data Anal.. 6. 429-449. 10.3233/IDA-2002-6504.

[15] Zhang, J.P. and Mani, I. (2003) KNN Approach to Unbalanced Data Distributions: A Case Study Involving Information Extraction. Proceeding of International Conference on Machine Learning (ICML 2003), Workshop on Learning from Imbalanced Data Sets, Washington DC, 21 August 2003.

[16] López, Victoria & Fernández, Alberto & García, Salvador & Palade, Vasile & Herrera, Francisco. (2013). An insight into classification with imbalanced data: Empirical results and current trends on using data intrinsic characteristics. Information Sciences. 250. 113–141. 10.1016/j.ins.2013.07.007.

[17] Priya, S., Uthra, R.A. Comprehensive analysis for class imbalance data with concept drift using ensemble based classification. J Ambient Intell Human Comput (2020). https://doi.org/10.1007/s12652-020-01934-y

[18] Krawczyk, Bartosz & Minku, Leandro & Gama, João & Stefanowski, Jerzy & Wozniak, Michal. (2017). Ensemble learning for data stream analysis: A survey. Information Fusion. 37. 132–156. 10.1016/j.inffus.2017.02.004.

[19] Gomes, Heitor Murilo & Barddal, Jean Paul & Enembreck, Fabrício & Bifet, Albert. (2017). A Survey on Ensemble Learning for Data Stream Classification. ACM Comput. Surv.. 50. 23:1-23:36. 10.1145/3054925.

[20] Fan, Wei & Han, Jiawei & Yu, Philip. (2007). A General Framework for Mining Concept-Drifting Data Streams with Skewed Distributions. 10.1137/1.9781611972771.1.

[21] Soares, R.G.F. & Santana, Alixandre & Canuto, Anne & de Souto, Marcilio. (2006). Using Accuracy and Diversity to Select Classifiers to Build Ensembles. 1310 - 1316. 10.1109/IJCNN.2006.246844.

[22] Barandela, R. & Valdovinos, Rosa & Sánchez, Josep. (2003). New Applications of Ensembles of Classifiers. Pattern Analysis Applications. 6. 245-256. 10.1007/s10044-003-0192-z.

[23] Chawla, Nitesh & Bowyer, Kevin & Hall, Lawrence & Kegelmeyer, W.. (2002). SMOTE: Synthetic Minority Over-sampling Technique. J. Artif. Intell. Res. (JAIR). 16. 321-357. 10.1613/jair.953.

[24] Nguyen, Hien & Cooper, Eric & Kamei, Katsuari. (2011). Borderline over-sampling for imbalanced data classification. International Journal of Knowledge Engineering and Soft Data Paradigms. 3. 4-21. 10.1504/IJKESDP.2011.039875.

[25] He, Haibo & Bai, Yang & Garcia, Edwardo & Li, Shutao. (2008). ADASYN: Adaptive Synthetic Sampling Approach for Imbalanced Learning. Proceedings of the International Joint Conference on Neural Networks. 1322 - 1328. 10.1109/IJCNN.2008.4633969.

[26] Street, Nick & Kim, YongSeog. (2001). A Streaming Ensemble Algorithm (SEA) for Large-Scale Classification. 377-382. 10.1145/502512.502568.

[27] Chen, Sheng & He, Haibo. (2009). SERA: Selectively recursive approach towards nonstationary imbalanced stream data mining. Proceedings of the International Joint Conference on Neural Networks. 522-529. 10.1109/IJCNN.2009.5178874.

[28] Wang, Shuo & Minku, Leandro. (2015). Resampling-Based Ensemble Methods for Online Class Imbalance Learning. Knowledge and Data Engineering, IEEE Transactions on. 27. 1356-1368. 10.1109/TKDE.2014.2345380.

[29] Ren, Siqi & Zhu, Wen & Liao, Bo & Li, Zeng & Wang, Peng & Li, Keqin & Chen, Min & Li, Zejun. (2018). Selection-based resampling ensemble algorithm for nonstationary imbalanced stream data learning. Knowledge-Based Systems. 163. 10.1016/j.knosys.2018.09.032.

[30] Thabtah, Fadi & Hammoud, Suhel & Kamalov, Firuz & Gonsalvesv, Amanda. (2019). Data Imbalance in Classification: Experimental Evaluation. Information Sciences. 513. 10.1016/j.ins.2019.11.004.

[31] Nsl-kdd data set for network-based intrusion detection systems. Available on: http://nsl.cs.unb.ca/KDD/NSLKDD.html, March 2009.

[32] Baena-García, Manuel & Campo-Ávila, José & Fidalgo-Merino, Raúl & Bifet, Albert & Gavald, Ricard & Morales-Bueno, Rafael. (2006). Early Drift Detection Method. In Fourth international workshop on knowledge discovery from data streams, vol. 6, pp. 77-86.

[33] Lee, Taesam & Singh, Vijay. (2019). Discrete k -nearest neighbor resampling for simulating multisite precipitation occurrence and model adaption to climate change. Geoscientific Model Development. 12. 1189-1207. 10.5194/gmd-12-1189-2019.

[34] Verma, Vijay & Aggarwal, Rajesh. (2020). A comparative analysis of similarity measures akin to the Jaccard index in collaborative recommendations: empirical and theoretical perspective. Social Network Analysis and Mining. 10. 10.1007/s13278-020-00660-9.

[35] Pinagé, Felipe & Santos, Eulanda & Gama, João. (2019). A drift detection method based on dynamic classifier selection. Data Mining and Knowledge Discovery. 34. 10.1007/s10618-019-00656-w.

[36] Gao, Yang & Chandra, Swarup & Li, Yifan & Khan, Latifur & Thuraisingham, Bhavani. (2020). SACCOS: A Semi-Supervised Framework for Emerging Class Detection and Concept Drift Adaption over Data Streams. IEEE Transactions on Knowledge and Data Engineering. PP. 1-1. 10.1109/TKDE.2020.2993193.

Figures

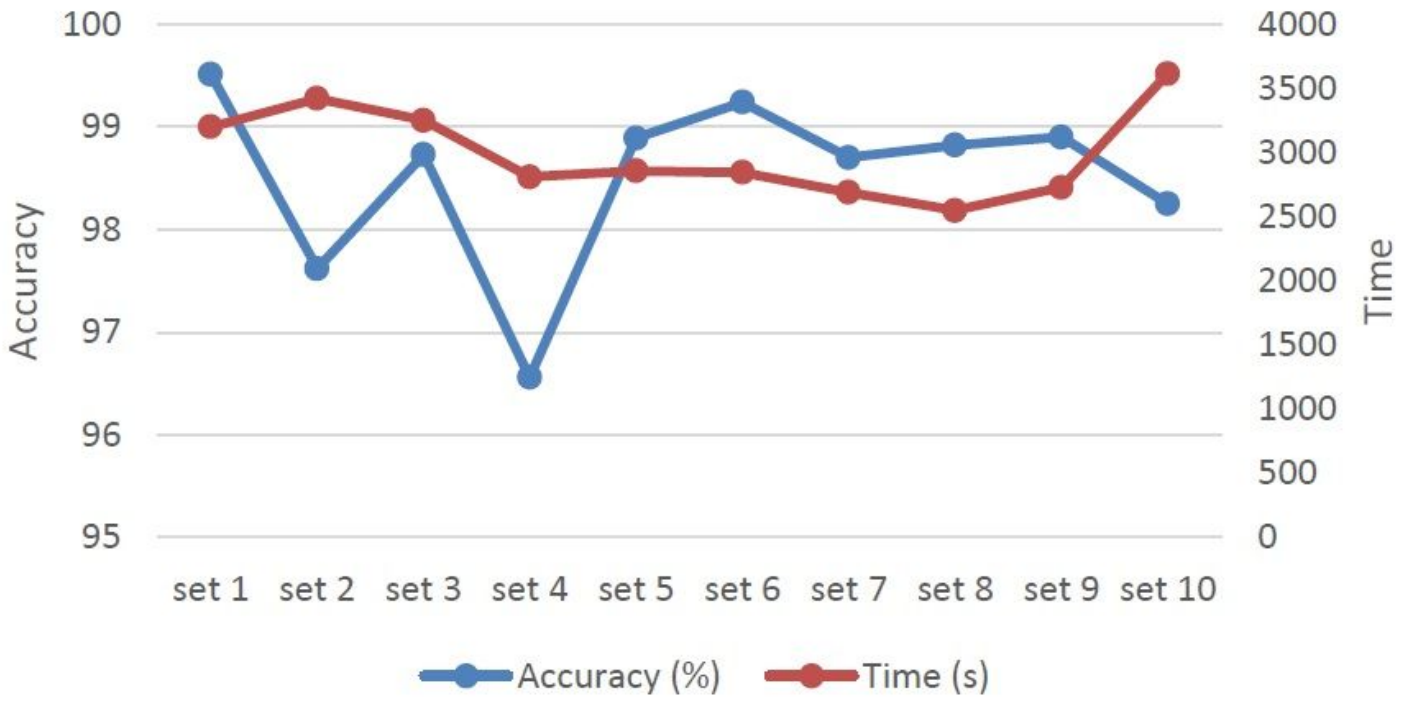

Figure 1

Performance Graph for TDDMRT over data chunk

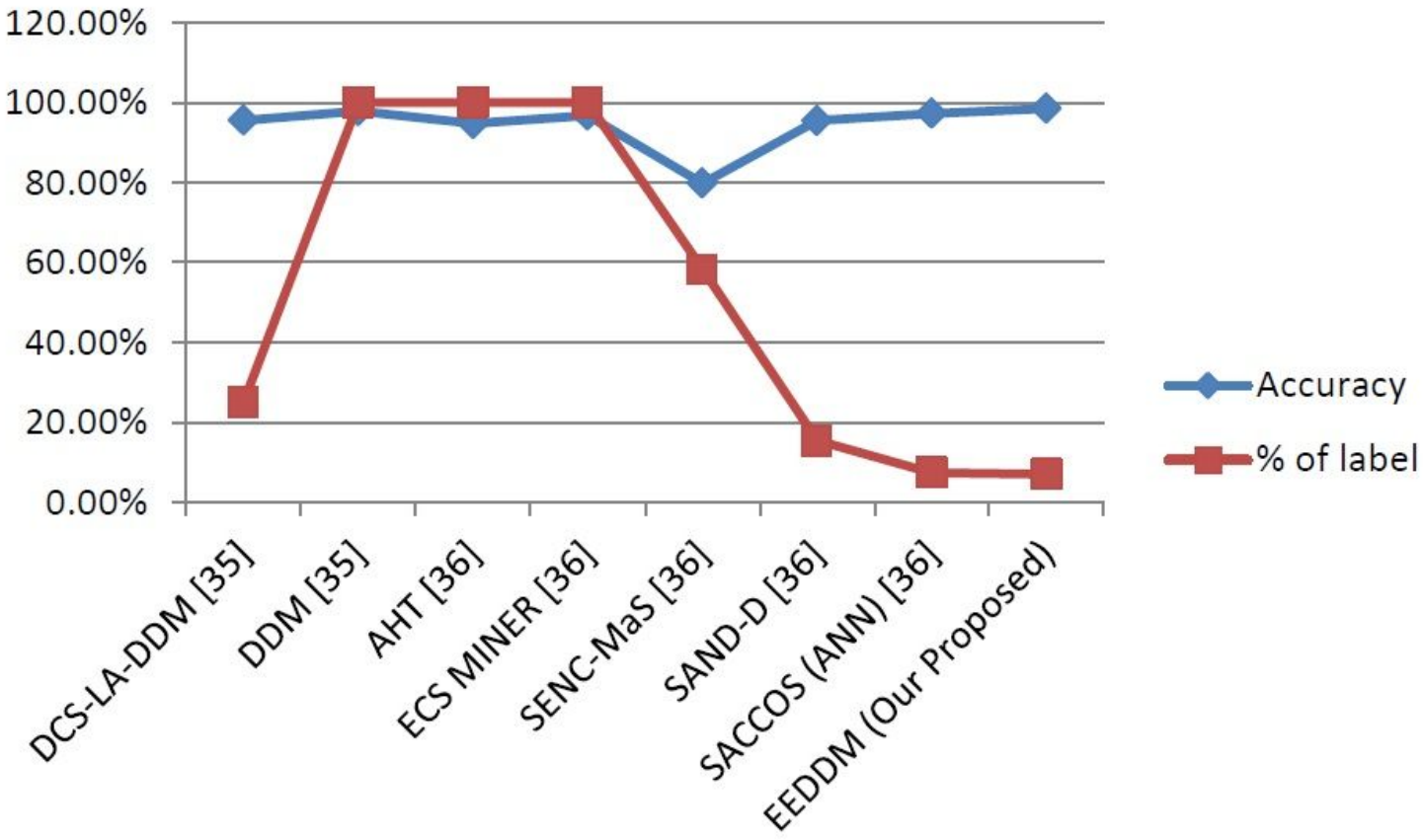

Figure 2

Performance of drift detectors

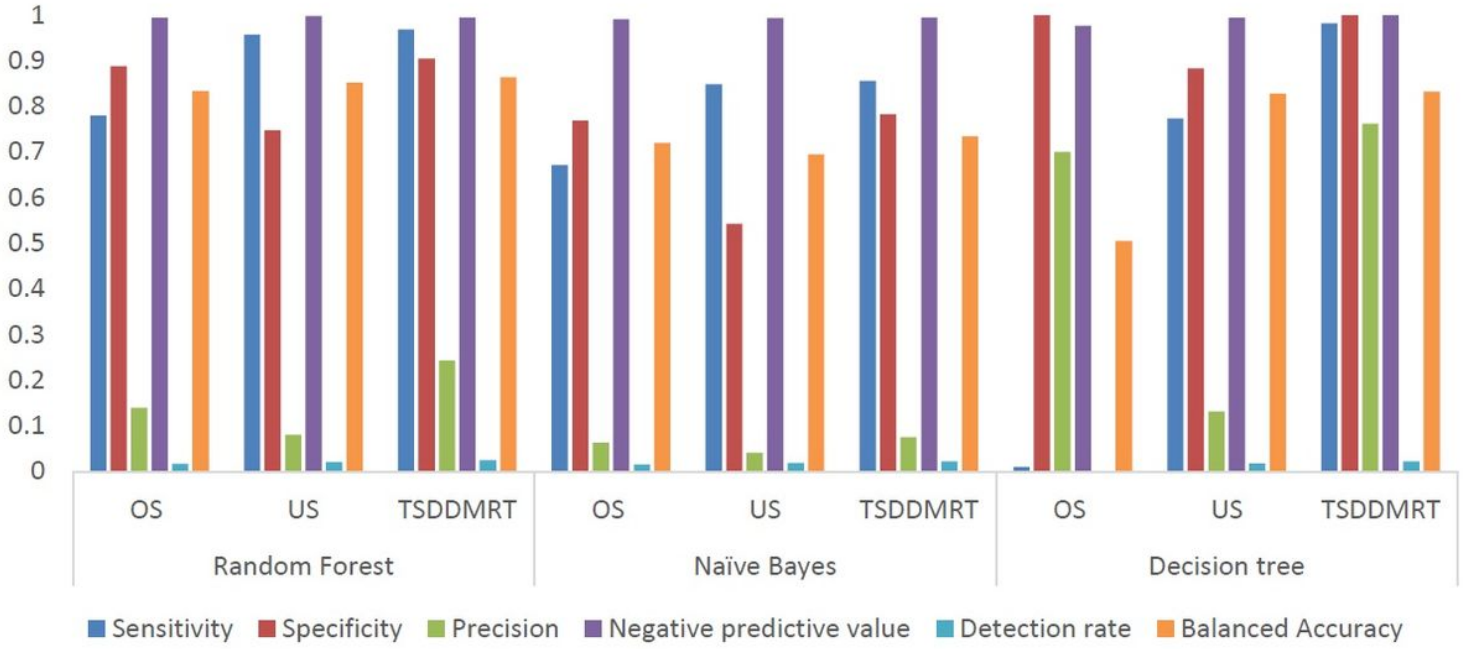

Figure 3

Performance of ML algorithm over the NSL-KDD cup data with resampling technique