

Hybrid dynamic chunk ensemble model for multi-class data streams

Varsha Sachin Khandekar, Pravin Shrinath

Department of Computer Engineering, Mukesh Patel School of Technology Management and Engineering, NMIMS University, Mumbai, India

Article Info

Article history:

Received Aug 19, 2021

Revised Nov 18, 2021

Accepted Dec 1, 2021

Keywords:

Concept drift

Data stream

Dynamic chunk

Ensemble learning

Imbalance data

ABSTRACT

In the analysis more specifically in the classification of continuous data stream using machine learning algorithms joint occurrence of concept drift and imbalanced issue becomes more provocative. Also, imbalance issue is again more challenging when the data stream is multi-class with minority class and that is too with data-difficulty factors. Incremental learning with ensemble models found more promising in handling these issues. But most of the approaches are for two-class data streams which can't be utilized for multi-class data streams. In this paper we have designed hybrid dynamic chunk ensemble model (HDCEM) for the classification of multi-class insect-data stream for handling imbalance and concept drift issue. To deal with imbalance issue we have proposed effective split bagging algorithm which has achieved better performance on minority class recall and F-measure on arriving dynamic chunks of data from multi-class data stream. HDCEM model can adapt to abrupt and gradual drift because it has combined features of both online and chunk-based learning together. It has achieved average 78% minority class recall in abrupt insect data stream and 71% in gradual drift insect stream.

This is an open access article under the [CC BY-SA](#) license.



Corresponding Author:

Varsha Sachin Khandekar

Department of Computer Engineering, Mukesh Patel School of Technology Management and Engineering
NMIMS University, V. L, Pherozeshah Mehta Rd, Vile Parle West, Mumbai
Maharashtra 400056, India

Email: varsha.khandekar@gmail.com

1. INTRODUCTION

In today's digital world due to evolution in technology voluminous data is increasing at rapid speed. The data generated on fly is called as a streaming data. The streaming data analysis using machine learning techniques endure many challenges like high data velocity, high volume of data, change in the underlying distribution over the time. For example, the properties of malicious uniform resource locator (URLs) and fraudulent transactions as well as spam tweets posted by spammers are changing continuously [1], [2]. In the analysis classification of instances from data stream, model have a hypothesis which finds mapping between feature variables (X) and target variables (Y) which are called as labels of the instances. There is a need of adaptive machine learning models which are able to adapt themselves to new underlying distribution which is known as a concept drift. The concept drifts are mainly further categorized as virtual and real drift.

Virtual drift: There is a change in the distribution of features $p(x)$ of instances or change in the distribution of concepts or target variables $p(y)$. Real drift: The relationship between input variables and target concept is changing. This change is change in the likelihood $p(x|y)$ thus $pt(x|y) \neq pt + 1(x|y)$ or change in posterior probability distribution $p(y|x)$ thus $pt(y|x) \neq pt + 1(y|x)$. The real drift affects on the decision boundary. In many real-world problems, most of the time they occur at same time. Dealing with concept drift

issue in the data stream classification task has become challenging and has become attentive in research community.

Earlier approaches have used some statistical change detection tests to monitor concept drift. Instead of using accuracy, drift detection method for online class imbalance (DDM-OCI) [3] monitors the class recall to deal with imbalance issue. For detecting drift on positive and negative class, linear four rates [4] have used true positive rate, true negative rate, positive predicted value and negative predicted value. Page-hinkley (PAUC-PH) [5] uses PH-test [6] to detect drift. Another important issue which affects the classification model's performance is class imbalance where number of instances of one of the class is dominant over the other. When imbalance and concept drift both these problems occur at same time in data stream, they will tend to exasperate each other. When class imbalance occurs, it becomes difficult to detect the concept drift and conform the model to new distribution. Active and Passive approaches have been used for handling the concept drifts. Active approach involves explicit detection technique while passive approach is based on adaption of model. Passive approach is more successful as compared to active which overcomes the limitations in an active approach. Class imbalance in stationary or in static environment is most addressed problem using various techniques. But there are only few models have been found which are dealing with both concept drift and class imbalance simultaneously. These models are categorized as online and chunk-based models. Chunk based models mostly have used ensemble learning approach. Online learning models [6], [7] adapt themselves for every instance arriving in the stream. These are more effective in handling abrupt kind of drift. In chunk-based learning, model is not adapting itself until certain number of instances are not collected in a buffer, whose size is mostly pre-decided and it is fixed throughout the analysis of data stream. Some chunk-based methods have used assignment of dynamic weights to component classifiers in ensemble model based on the accuracy measure [8].

There were some fixed size chunk based methods proposed which were used for classification imbalanced non-stationary data streams [9], [10]. In uncorrelated bagging [11] current chunk is balanced by preserving the minority class examples from previous chunks. But here the limitation is usage of memory for storing past data instances and also this can't adapt to new concept rapidly. Improvement in this technique is observed in selectively recursive (SERA) [12] and in recursive ensemble approach (REA) [13] by selecting only most similar past minority instances. Ditzler and Polikar [14] proposed two chunk-based ensembles called learn++. CDS that is concept drift with smote and learn++. NIE which is non-stationary imbalanced environment. Both are inspired from learn++. NSE to handle imbalanced data streams with concept drift [15] where learn++. NSE deals with concept drift using a dynamic weighting strategy and SMOTE for balancing the minority class instances. An ensemble of subset of online sequential extreme learning machine (ESOS-ELM) [16] have constructed and stored weight matrices for every chunk. Gradual resampling ensemble (GRE) [17] used clustering technique for selecting the minority class samples from previous chunk. To generate training dataset, they have used density based spatial of applications with noise (DBSCAN) clustering with minority class and tried to minimize overlapping with majority class.

Also, in few chunk-based methods preserve the minority samples from previous chunk which are merged with the minority samples in the succeeding chunk to get enough number of minority samples, however, this assumption may fail as imbalance ratio may not be fixed and may be changing over the time. Review shows that bagging based ensembles are useful for improving the performance of classifier for dealing with imbalance issue. Proposed method is based on bagging approach and compared with following state-of-the-art bagging methods. i) over bagging [18], this technique relies on a random over sampling of minority class to acquire each subset of dataset. Here every subset will include all the original examples and duplicate samples of randomly selected instances of the minority class; ii) synthetic minority oversampling technique (SMOTE) bagging [18], this approach makes use of SMOTE algorithm for creating new instances from minority class. To increase diversity in subset majority class instances are selected randomly; iii) under bagging [19], here instead of using under sampling, it uses oversampling technique for generating subsets from original dataset. Because of undersampling size of subset gets reduced; and iv) under over bagging [19], this approach uses both undersampling and oversampling along-with SMOTE bagging.

Most of the methods reviewed designed for two class classification in data stream, so there was one minority while another one was majority class. But these methods were failed to handle multiple minority classes and the dynamic imbalance ratio in multi-class data streams. Also, the size of chunk considered affect the performance of a model, explicitly when the data stream is imbalanced. In this paper, we have proposed a hybrid dynamic chunk ensemble model (HDCEM) for classification of multi-class insect imbalanced data streams. In proposed ensemble model decision tree is used as a candidate classifier. For test data classification dynamic ensemble selection is used. The proposed model has following advantages: i) It is able to perform the multi-class classification in non-stationary data streams, ii) Imbalance issue is resolved using novel split based resampling ensemble algorithm, iii) It can handle abrupt and gradual concept drifts, as features of both online and chunk-based learning are combined, and iv) For test data dynamic ensemble selection is applied.

2. RESEARCH METHOD

2.1. Dataset

Proposed model has been evaluated using mosquito insect stream dataset released by authors [20]. This dataset has 33 features and six class labels. The class labels are the species of three types of mosquitoes from both sexes. Details of these mosquito species are given as shown.

- *Aedes aegypti*. This mosquito species is commonly known as yellow fever mosquito. It is involved in spreading dengue fever, zika fever, chikungunya, mayaro, yellow fever viruses, and other disease agents [21].
- *Aedes albopictus*. This mosquito species called as Asian tiger mosquito or forest mosquito. It can spread diseases including yellow fever, dengue fever, and chikungunya fever [21].
- *Culex quinquefasciatus*. This is known as the Southern house mosquito. It is a medium-sized mosquito found in tropical and subtropical regions of the world. It is important in transmission of wuchereria bancrofti, avian malaria, and arboviruses including vSt, and louis encephalitis virus [22].

Features of this dataset are extracted by processing optical signal by using signal processing techniques. These features include wing beat frequency, various statistics from temporal representation, complexity measures of signal spectrum and so on. There are three variations of these datasets with abrupt, gradual and recurring concept drifts. Authors have generated this insect stream datasets with concept drifts using optical sensor based smart trap for catching the insects. The dataset with different concept drifts is generated by doing the variations in the temperature which may affect on the distribution of features of the insects.

2.2. Description of proposed model

In this section we describe our proposed model with pseudo-code. The proposed model is shown in Figure 1. Consider a data stream $S = \{xi, yi\}$, where xi is an input feature vector and $yi \in \{c1, c2, \dots, cm\}$ is output variable or class label of xi . Overall proposed model is described in pseudo-code under Figure 2. In the first phase, the model is trained on dynamic sized chunks. Instead of fixing the size of chunk initially, the chunks of dynamic size are formed by considering enough number of instances from all classes of the dataset (lines 5-6). These chunks are used for training the ensemble model. Stability of model is achieved by monitoring the error rate and applying statistical test on the variances in prediction error [23] (lines 7-14). Here, hybrid ensemble model is trained using the whole chunk, at the same time one special competent classifier is trained using every instance from the chunk, so abrupt or gradual drift, if exists, can be handled effectively. For handling imbalanced issue in chunks, split based resampling algorithm is used which is described in pseudo-code under Figure 3. For test data k-nearest neighbors (KNN) based dynamic ensemble selection is used (lines 14-21).

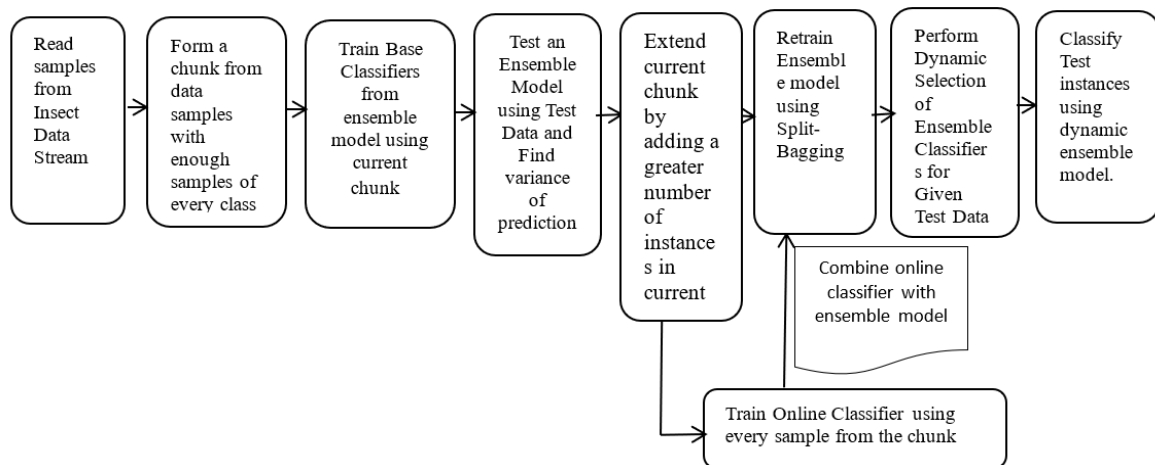


Figure 1. Framework for proposed model

To deal with imbalance issue in data stream we have implemented split bagging technique which is presented in pseudo-code under Figure 3. In multi-class imbalance data set, along with dynamic imbalance ratio of classes, there are other data difficulty factors like noisy minority instances and overlapping classes which deteriorates the performance of classifiers, although we generated the balanced dataset. In addition to

this diversity of training subsets is also crucial for improving the performance of classifier. Proposed algorithm has systematically created number of subsets by splitting the classes into partitions based upon the size of minority class and every partition is filled with the equal number of instances from majority class which are selected using negative binomial distribution [24] using as (1) (lines 1-13) and synthetic minority examples.

$$p(m|n) = \binom{m+n-1}{n} p^n q^m \quad (1)$$

Where, m = number failures for given n = number of successes and $p = q = 0.5$. While generating synthetic minority instances instead of generating randomly, synthetic examples are generated only from the safe minority samples and using the nearest neighbors for every class. So, the synthetic examples which will be generated will not contain any noisy or hard to learn instances (lines 14-15). Train the classifiers in ensemble model using balanced subsets (lines 16-18).

```

1:  Input: S: {xi, yi} Imbalanced Data Stream, yi = {c1, c2,..., cp} N: Number of
      component classifiers, Dk: Data chunk with dynamic size, Dt: Test data chunk for
      building ensemble model Mt: Initial ensemble model, Dl: Extended data chunk.
      chval: value returned by statistical test, th: predefined threshold value, Vt:
      Variance in prediction of test instances, Vl: Variance in prediction of test
      instances when ensemble classifier trained on extended chunk, Dtest: Test data
      for assessing the performance of Ensemble model, Ch: Ensemble Classifier with N
      number of component classifiers.
2:  Mt = φ
3:  Output: Ensemble model Ch
4:  for all xi in S
5:    Dk={x1, x2,..., xt } # collect samples from stream with enough number of samples
      of each class
6:    Dt =(Select random number of samples as a test data)
7:    Mt, Dk, Ch = SplitBagging (Dk, N)
8:    Ch = Ch U Mt
9:    Vt = CalculateVariance(Mt,Dt)
10:   Dl = {xt...xtd } # Add more samples in previous chunk ,this is an extended
      chunk.
11:   Ml, Dl, Ch= SplitBagging(Dl,N)
12:   Vl =CalculateVariance(Ml,Dt)
13:   chval = checkStability(Vt,Vl)
14:   if chval > th
15:     Cl = SplitBagging(Dl)
16:     Ch=Ch U Cl
17:     for i <-- 1 to |Dt|
18:       Yp <- Knn-DES(Ch,Dl,Dt) # dynamic ensemble selection
19:     end for
20:   end if
21: end for

```

Figure 2. Pseudo code for HDCEM

```

1:  Input: Dt:{xi,yi} imbalanced data chunk, C={c1,c2,...ck}, k: number of classes,
      L:No.candidate classifiers in Ensemble model, Qmaj = Size of Majority
      class, Qmin = Size of minority class, p : Number of classifiers in
      ensemble
2:  Output: bs = Balanced Data, Ch = Trained Ensemble model
3:  for i = 1 to p-1
4:    csi = |size of class ci|
5:    if csi % Qmin =0
6:      Npci = csi/Qmin
7:    else
8:      Npci =csi/Qmin +1
9:  for j=1 to k-1
10:   for l = 1 to Npcj
11:     Select Qmin nearest neighbors from minority class from class j
      using Negative Binomial Distribution using equation no (1) and assign to
      partition partlj
12:     Add partlj for class j and add to bs
13:   end for
14:   Select k number of nearest neighbors from minority class Qmin for class j
      and Take only safe minority class examples and generate samples using SMOTE
      and add this in bs
15: end for
16: for j = 1 to L
17:   Select partition from bs for every class and create a subset and add to Btr
      Cl <- L(Btr)
18:   Ch <- Ch U Cl
19:   Return ensemble model Ch, Btr

```

Figure 3. Pseudo code for split bagging

Proposed model uses dynamic ensemble selection for classification of test data. Pseudo-code for KNN-dynamic ensemble selection (KNN-DES) is shown in Figure 4. Here for every instance from test data, region of competence is computed based on k-nearest neighbors from training dataset. Only those k classifiers of selected k-nearest neighbors are used for deciding the label of test instance by maximizing the probability of prediction of that respective class label.

```

1:  Input :Btr : Balanced Training Chunk, Bte:Test Data , Ch: Ensemble model .
2:  Output : Output label for every test instance, xj ∈ Bte
3:  for every instance xj ∈ Bte
4:      find Xt= N k-nearest neighbors from training chunk Btr for instance, xj
5:      for t = 1 to N
6:          Wt = 1/dt # dt is an Euclidean distance between xj and xt, xt ∈ Xt
7:      end for
8:      Normalize weight wt =  $\frac{wt}{\sum_{t=1}^N wt}$ 
9:      for each cl ∈ Ch
10:         yj=C(xj) = argmax (  $\sum_{i=1}^N Pr( yk | xj \in Btr ,cl ) * wt$  )
11:     end for
12: Return yj

```

Figure 4. Pseudo code for KNN-DES

3. RESULTS AND DISCUSSION

In this section, we have done comparative analysis of proposed model HDCEM- split bagging with HDCEM-SMOTE bagging, HDCEM-over bagging and HDCEM-under over bagging. We intent to verify the potency of the proposed split bagging in multi-class imbalanced data streams. When data is imbalanced for assessing the performance of model rather than using accuracy F-measure, precision minority class recall performance measures are used. The precision, also called as True positive rate or specificity, is the ratio of correctly predicted positive instances to total predicted positive instances. the recall, also called as sensitivity, is the ratio of true positive instances to actual positive instances. F1 score or F-measure, is weighted average of precision and recall. These performance measures are defined as (2), (3), and (4).

$$\text{Precision} = \frac{TP}{TP+FP} \tag{2}$$

$$\text{Recall} = \frac{TP}{TP+FN} \tag{3}$$

$$F - \text{Measure} = \frac{2*(\text{Precision}*\text{Recall})}{(\text{Precision}+\text{Recall})} \tag{4}$$

3.1. Experimental results

The F-measure and minority class recall for imbalanced insect data streams with abrupt, gradual concept drifts at different chunk size are shown using following graphs. From Figures 5(a) and (b) it is clear that the proposed HDCEM with split bagging model has given stable performance for both abrupt and gradual drift insect data streams and outperformed over SMOTE bagging, over bagging and under over bagging for different chunk sizes. Minority class recall using proposed model is better than other techniques.

Average minority class recall in abrupt drift insect data stream is 78% and 71% in gradual drift insect data stream. In abrupt drift, the accuracy is more because there is one dedicated component classifier used which learns every instance from current chunk. While for gradual drift insect data stream, although chunk size is more as compared to abrupt drift data stream chunk size, minority class recall has not improved.

Reason behind this is that there might be increased number of hard to learn instances or increased imbalance ratio. Average overall accuracy achieved is 91%, but due to limited space graphical analysis is not shown here. To support this result, we have applied non parametric Mann-Whitney’s U statistical test [25]. Results of this statistical test in the form of average rank values (R+) and (R-) between proposed model with split bagging and with other bagging techniques are shown in Table 1. Table 1 depicts that the proposed model has outformed.

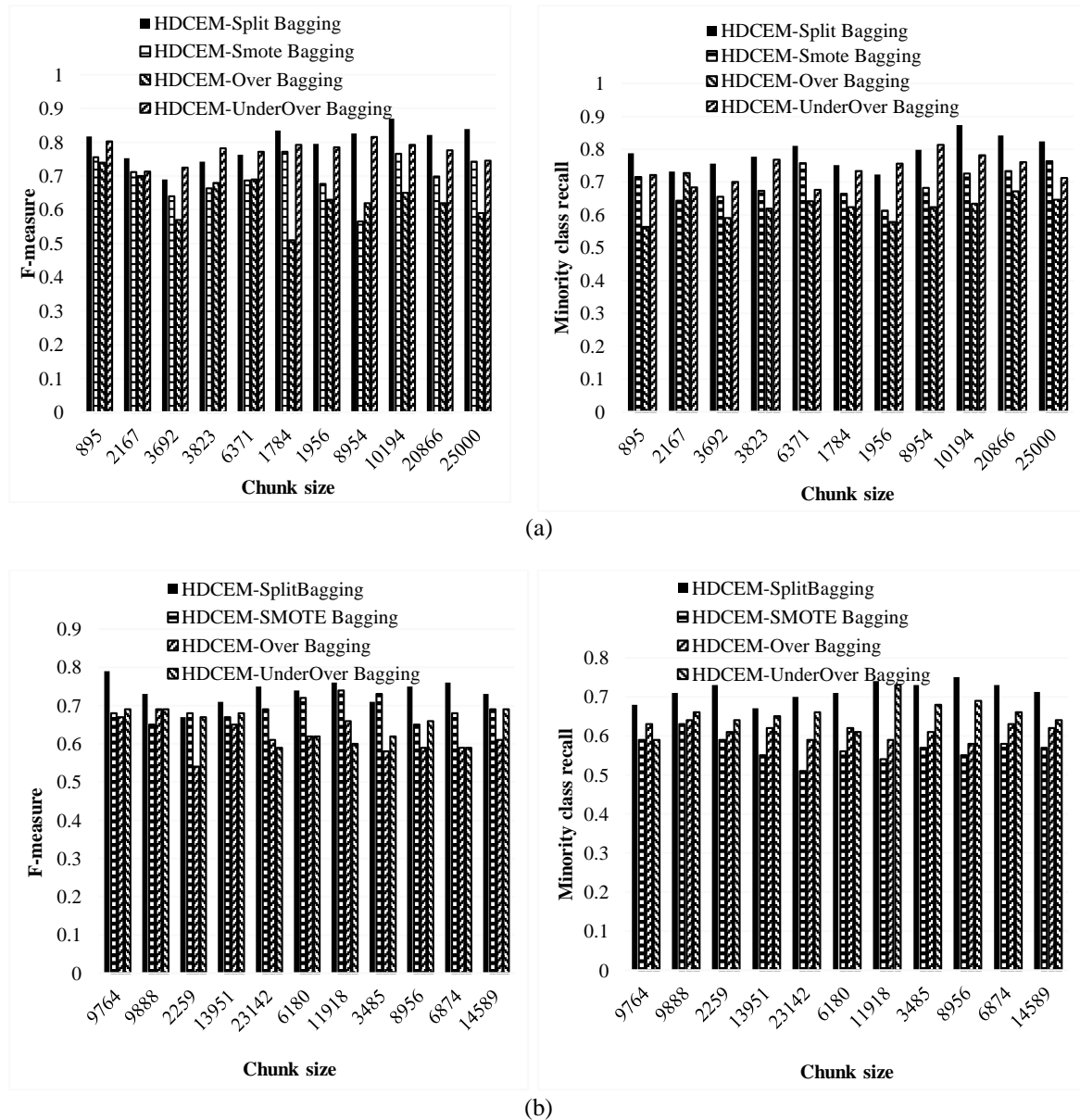


Figure 5. Comparative analysis using F-measure and minority class recall for (a) insect data stream with abrupt drift and (b) insect data stream with gradual drift

Table 1. Mann-Whitney’s U statistical test results

Comparison	F-Measure			Minority class recall		
	P-value	(R+)	(R-)	P-value	(R+)	(R-)
HDCEM-split bagging Vs HDCEM-SMOTE bagging	< 0.01*	166	87	< 0.01*	161	92
HDCEM-split bagging Vs HDCEM-over bagging	< 0.01*	181	72	< 0.01*	183	70
HDCEM-split bagging Vs HDCEM-under-over bagging	< 0.01*	143	110	< 0.01*	142	111

3.2. Computational time complexity

Figure 6 shows the time required for chunk processing. As the insect data stream is multi-class, computational cost of proposed model is high due to balancing technique used for generating diverse balanced subsets with handling of some of the data difficulty factors.

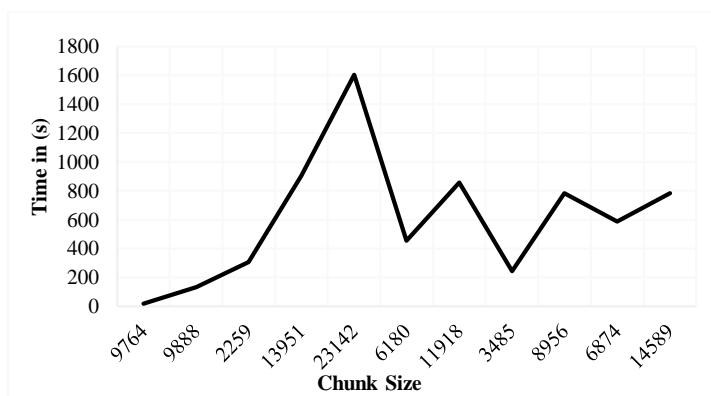


Figure 6. Computational time for chunk processing

4. CONCLUSION

For the classification of data streams, concept drift and imbalance issues are the major problems. In this paper we have proposed HDCEM for multi-class data stream to deal with these issues. HDCEM generates an ensemble model which is trained on data chunks whose size is decided dynamically rather than fixing it a priori. Also, for handling dynamic imbalance issue we have proposed Split based Bagging algorithm which can handle noisy, hard to learn minority and majority instances present in the dataset. In addition to this, instead of applying direct majority voting ensemble algorithm for test data prediction, k-nearest neighbor based dynamic ensemble selection is used. Experimental results showed that proposed model has outperformed, but it is computationally expensive. The time requirement for processing multiple classes in data stream is more so for future work we can implement proposed model using distributed environment platforms like Hadoop or Spark.

ACKNOWLEDGEMENTS

Authors thanks to Department of Computer Engineering, Mukesh Patel School of Technology Management and Engineering Mumbai, India, and Smt. Kashibai Navale College of Engineering, Pune for providing an infrastructure and support to carry research on above mentioned topic.




REFERENCES

- [1] H. M. Gomes, J. P. Barddal, A. F. Enembreck, and A. Bifet, "A survey on ensemble learning for data stream classification," *ACM Computing Surveys*, vol. 50, no. 2, pp. 1–36, Mar. 2017, doi: 10.1145/3054925.
- [2] B. Krawczyk, L. L. Minku, J. Gama, J. Stefanowski, and M. Woźniak, "Ensemble learning for data stream analysis: A survey," *Information Fusion*, vol. 37, pp. 132–156, Sep. 2017, doi: 10.1016/j.inffus.2017.02.004.
- [3] S. Wang, L. L. Minku, and X. Yao, "A Systematic Study of Online Class Imbalance Learning with Concept Drift," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 10, pp. 4802–4821, Oct. 2018, doi: 10.1109/TNNLS.2017.2771290.
- [4] H. Wang and Z. Abraham, "Concept drift detection for streaming data," in *Proceedings of the International Joint Conference on Neural Networks*, Jul. 2015, vol. 2015-September, doi: 10.1109/IJCNN.2015.7280398.
- [5] D. Brzezinski and J. Stefanowski, "Prequential AUC for classifier evaluation and drift detection in evolving data streams," in *Lecture Notes in Artificial Intelligence (Subseries of Lecture Notes in Computer Science)*, vol. 8983, Springer International Publishing, 2015, pp. 87–101.
- [6] S. Wang, L. L. Minku, and X. Yao, "Resampling-based ensemble methods for online class imbalance learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 27, no. 5, pp. 1356–1368, May 2015, doi: 10.1109/TKDE.2014.2345380.
- [7] S. Wang, L. L. Minku, D. Ghezzi, D. Caltabiano, P. Tino, and X. Yao, "Concept drift detection for online class imbalance learning," Aug. 2013, doi: 10.1109/IJCNN.2013.6706768.
- [8] K. Wu, A. Edwards, W. Fan, J. Gao, and K. Zhang, "Classifying imbalanced data streams via dynamic feature group weighting with importance sampling," in *SIAM International Conference on Data Mining 2014, SDM 2014*, Apr. 2014, vol. 2, pp. 722–730, doi: 10.1137/1.9781611973440.83.
- [9] Y. Lu, Y. M. Cheung, and Y. Yan Tang, "Adaptive chunk-based dynamic weighted majority for imbalanced data streams with concept drift," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 31, no. 8, pp. 2764–2778, Aug. 2020, doi: 10.1109/TNNLS.2019.2951814.
- [10] G. Ditzler, M. Roveri, C. Alippi, and R. Polikar, "Learning in Nonstationary Environments: A Survey," *IEEE Computational Intelligence Magazine*, vol. 10, no. 4, pp. 12–25, Nov. 2015, doi: 10.1109/MCI.2015.2471196.
- [11] J. Gao, W. Fan, J. Han, and P. S. Yu, "A general framework for mining concept-drifting data streams with skewed distributions," in *Proceedings of the 7th SIAM International Conference on Data Mining*, Apr. 2007, pp. 3–14, doi: 10.1137/1.9781611972771.1.
- [12] S. Chen and H. He, "SERA: Selectively recursive approach towards nonstationary imbalanced stream data mining," in *Proceedings of the International Joint Conference on Neural Networks*, Jun. 2009, pp. 522–529, doi: 10.1109/IJCNN.2009.5178874.
- [13] H. He and S. Chen, "Towards incremental learning of nonstationary imbalanced data stream: A multiple selectively recursive approach," *Evolving Systems*, vol. 2, no. 1, pp. 35–50, Nov. 2011, doi: 10.1007/s12530-010-9021-y.




- [14] G. Ditzler and R. Polikar, "Incremental learning of concept drift from streaming imbalanced data," *IEEE Transactions on Knowledge and Data Engineering*, vol. 25, no. 10, pp. 2283–2301, Oct. 2013, doi: 10.1109/TKDE.2012.136.
- [15] R. Elwell and R. Polikar, "Incremental learning of concept drift in nonstationary environments," *IEEE Transactions on Neural Networks*, vol. 22, no. 10, pp. 1517–1531, Oct. 2011, doi: 10.1109/TNN.2011.2160459.
- [16] B. Mirza, Z. Lin, and N. Liu, "Ensemble of subset online sequential extreme learning machine for class imbalance and concept drift," *Neurocomputing*, vol. 149, no. Part A, pp. 316–329, Feb. 2015, doi: 10.1016/j.neucom.2014.03.075.
- [17] S. Ren, B. Liao, W. Zhu, Z. Li, W. Liu, and K. Li, "The Gradual Resampling Ensemble for mining imbalanced data streams with concept drift," *Neurocomputing*, vol. 286, pp. 150–166, Apr. 2018, doi: 10.1016/j.neucom.2018.01.063.
- [18] S. Wang and X. Yao, "Diversity analysis on imbalanced data sets by using ensemble models," in *2009 IEEE Symposium on Computational Intelligence and Data Mining, CIDM 2009 - Proceedings*, Mar. 2009, pp. 324–331, doi: 10.1109/CIDM.2009.4938667.
- [19] R. Barandela, J. S. Sánchez, and R. M. Valdovinos, "New Applications of Ensembles of Classifiers," *Pattern Analysis and Applications*, vol. 6, no. 3, pp. 245–256, Dec. 2003, doi: 10.1007/s10044-003-0192-z.
- [20] V. M. A. Souza, D. M. dos Reis, A. G. Maletzke, and G. E. A. P. A. Batista, "Challenges in benchmarking stream learning algorithms with real-world data," *Data Mining and Knowledge Discovery*, vol. 34, no. 6, pp. 1805–1858, Jul. 2020, doi: 10.1007/s10618-020-00698-5.
- [21] N. G. Gratz, "Critical review of the vector status of *Aedes albopictus*," *Medical and Veterinary Entomology*, vol. 18, no. 3, pp. 215–227, Sep. 2004, doi: 10.1111/j.0269-283X.2004.00513.x.
- [22] L. C. Bartholomay *et al.*, "Pathogenomics of *Culex quinquefasciatus* and meta-analysis of infection responses to diverse pathogens," *Science*, vol. 330, no. 6000, pp. 88–90, Oct. 2010, doi: 10.1126/science.1193162.
- [23] G. Snedecor and W. Cochran, "Statistical methods, 8thEdn," *Statistical Methods*, p. 503, 1989.
- [24] S. Hido and H. Kashima, "Roughly balanced bagging for imbalanced data," in *Society for Industrial and Applied Mathematics - 8th SIAM International Conference on Data Mining 2008, Proceedings in Applied Mathematics 130*, Apr. 2008, vol. 1, pp. 143–152, doi: 10.1137/1.9781611972788.13.
- [25] H. B. Mann and D. R. Whitney, "On a Test of Whether one of Two Random Variables is Stochastically Larger than the Other," *The Annals of Mathematical Statistics*, vol. 18, no. 1, pp. 50–60, Mar. 1947, doi: 10.1214/aoms/1177730491.

BIOGRAPHIES OF AUTHORS



Varsha Khandekar    received the M.E. degree in Computer Engineering from the Savitribai Phule Pune University, Maharashtra Pune, India in 2008. She is working toward the Ph.D. degree with the Department of Computer Engineering, NMIMS's Mukesh Pate School of Technology Management and Engineering, Mumbai. She is currently an Assistant Professor in the Department of Information Technology of Smt. Kashibai Navale College of Engineering Pune, India. Her research interests include Data-Mining, Ensemble Learning, Machine Learning, Data Science and Big Data Analytics. She can be contacted at email: varsha.khandekar@gmail.com.



Pravin Shrinath    received the M. Tech degree in Computer Engineering in 2008 and PhD degree in 2016 from NMIMS's Mukesh Patel School of Technology Management and Engineering, Mumbai. He is currently an Associate Professor & Head of Department Computer Engineering at NMIMS's MPSTME, Mumbai. His research interests include Image Processing, Artificial Intelligence, Machine Learning. He can be contacted at email: pravin.srinath@nmims.edu.