Ingénierie des Systèmes d'Information
Vol. 28, No. 1, February, 2023, pp. 161-167
Journal homepage: http://iieta.org/journals/isi

# Ensemble Model for Multiclass Imbalanced Data Using Cluster Computing of Spark

Varsha S. Khandekar*, Pravin Shrinath

Department of Computer Engineering, Mukesh Patel School of Technology Management and Engineering, NMIMS University, Mumbai 400056, India

Corresponding Author Email: khandekar.varsha010@nmims.edu.in

https://doi.org/10.18280/isi.280117

**Received:** 1 September 2022
**Accepted:** 20 January 2023

***Keywords:***
*imbalance, Particle Swarm Optimization, SMOTE, spark, cluster computing, RDD, ensemble model, sampling*

**ABSTRACT**

Big data analysis using machine learning has become a challenging problem today. Classification problems become more challenging when class distribution is imbalanced. In this paper, we propose a distributed ensemble model with an intelligence technique based on Particle Swarm Optimization to overcome the imbalanced problem. For compensating the class imbalance, first SMOTE is used to balance the minority class samples, and then sampling based on Particle Swarm Optimization is applied. Here, to perform fast processing, the whole model is implemented using spark-cluster computing, which uses the underlying concept of parallel programming of spark RDD. Results of the proposed system have shown consistent improvements on several evaluation metrics and overall processing time. Evaluation of the proposed system has been done using different performance metrices also comparison between sequential and distributed ensemble models. Most of the existing techniques show different performances for different datasets, while the proposed method has shown better generalization property, which improves the data-model dependency issue. The proposed model has been evaluated using KDD-CUP'99 intrusion detection and insect sensor datasets. For the datasets, it shows better improvement over traditional sampling techniques. F-Measure value is 99% for KDD'cup dataset and 92% for insect dataset.

## 1. INTRODUCTION

Today, an enormous volume of data in various forms is generated from various data sources and is growing at an exponential rate. According to big data growth statistics, data creation will be very huge and will grow exponentially in next few years. Although this data has become a rich source of information, the analysis of the big data is becoming more challenging day by day for the researchers. Traditional methods, for which the primary pre-requisite was to store the data in one place, are becoming obsolete to handle this complex data. Faster response after analysis of the data is today's important demand, which is increasing nowadays. To get a faster response, faster processing of data is necessary. To tackle this big data challenge, researchers' main focus is on improving the speed of data processing as well as parallelizing the multiple tasks in analysis. In analysis of data classification of data is one of the important supervised learning task which correctly classify the dataset and predict the class label for unseen or test instance. Literature shows that there is vast research work carried out on classification algorithms. According to the No Free Lunch Theorem, there is no single algorithm which works better for any classification problem [1]. The performance of classification algorithm varies depending on the dataset on which it is applied. Ensemble algorithms also outperformed in classification tasks. This ensemble algorithm is also recognized with different names like multiple classifiers or Muli-classifier System (MCS) or Classifier Fusion [2, 3], which contains multiple component or base classifiers. Predictions of component classifiers are combined efficiently to predict the class of a new instance. These ensemble classifiers typically operate in three stages: a. generation, b. selection, and c. integration [4]. In generation phase either homogeneous or heterogeneous approach is used, in selection phase different ways are used to select subset of classifiers which are further integrated to get the final prediction. Among them, Majority Voting and Meta-classifiers are some of the most popular strategies. Bagging, boosting, and stacking are the most popular ensemble techniques for solving classification problems.

To deal with big data classification problems literature shows ensemble methods are promising using popular methodologies like Map-Reduce programming paradigm on Hadoop or Spark Big Data platforms which supports the distributed environment. Apache Spark is the successor of Hadoop, which is popular for fast processing of voluminous data. It is supported by a rich set of libraries like SPARK SQL, MLlib, Graphx for SQL processing, Machine Learning algorithms, and for graph processing [5].

The classification imbalanced issue is one of the major issues which affects the performance of the classifier. In some application areas like medical-diagnosis, accuracy of minority classes is important, which most of the time is very low because generally the classifiers are biased towards the majority classes. To tackle the problem of imbalance issue, mainly there are three strategies used, first, cost-sensitive approach where the cost of miss-classified instances are considered, second, algorithmic level where parameter tuning and learning mechanisms are used, in third data-processing level distribution of data instances is updated to re-balance the

data. Some recent research has focused on the use of evolutionary algorithms to deal with imbalanced issues. In this paper, to handle imbalanced issues, the following two methods are proposed: first, to deal with imbalanced big data, a hybrid method combining the data-level approach (SMOTE) and the Particle Swarm Optimization Bagging algorithm has been proposed. Second, to increase the speed of an algorithm, it is implemented using cluster computing and SPARK.

Section II provides a review of the literature on state-of-the-art methods for Big Data classification using various methodologies; Section III describes the proposed methodology and dataset; Section IV depicts experimental settings, results, and discussion; and Section V is the conclusion.

## 2. RELATED WORK

To overcome data imbalance problem many works have been explored under data-level and algorithmic-level approach [6]. Sampling techniques like Undersampling and Oversampling of majority class and minority class are the two common data -level approaches. Bagging based methods which are combined with these sampling techniques for imbalance data classification are categorized as Over Bagging and Under Bagging that is Bagging with Over Sampling and Bagging with Under Sampling, respectively. In over-bagging important step is oversampling of minority class instances so that the data becomes balanced i.e. equal number of instances for all classes. In oversampling either random sampling of minority instances with replacement and then these randomly sampled instances are duplicated or SMOTE is applied where synthetic samples are generated. In over-bagging bagging based on RandomOverSampling known as ROS-Bag and bagging based on SMOTE known as SMOTE-Bag are the commonly used approaches.

In under-bagging under-sampling of majority class instances are performed. Here subsets of majority class are generated which are roughly equal in size of minority class. These subsets are combined with minority class instances to generate balanced dataset for training the classifier. In literature, based on techniques used for undersampling different underbagging approaches have been proposed. In the research [7], majority class subset size is decided using negative binomial distribution while performing undersampling. This approach is known as Roughly Balanced Bagging. Exactly Balanced Bagging (abbreviated as EB-Bag) [7, 8] divides the original majority class into N disjoint subsets, where size of each subset is exactly similar to the minority class. Then each subset of majority class is combined with minority class instances to generate balanced dataset. In RandomUnderSampling (abbreviated as RUS-Bag) majority class instances are undersampled without replacement [9, 10]. Easy Ensemble [8, 9] generates majority class subset using random under-sampling with or without replacement. It makes use of AdaBoost algorithm to generate base classifiers in ensemble model. Similar to Easy-Ens, Bal-Cad algorithm also generates base classifiers using AdaBoost. Difference between these two algorithms is that it reduces the majority class instances in every iteration. It terminates when all base classifiers are generated or there are available instances from the training dataset [8, 11].

There are also PYTHON specific packages like 'imbalanced-learn' [12] which is combined with undersampling followed by oversampling to tackle this imbalance issue. In R language there are packages, like 'ROSE' [13] and 'CARET' [14] where synthetic samples are generated using smoothed bootstrap approach which handles imbalance as well as performs model estimation and accuracy evaluation. Also, recent work is towards combining optimization techniques and sampling techniques in classification problem. Like in the research [15], samples from majority class are selected using optimized undersampling of majority class instances. Authors of [16] have introduced intelligent oversampling SMOTE technique in combination with Support Vector Machine (SVM). Recently there are some other hybrid approaches like Neighbors Progressive Competition (NPC) algorithm, Random Hybrid Sampling [17] and Bagging of Extrapolation SMOTE SVM have been introduced [18]. Recently, there are intelligent techniques have been introduced to tackle imbalance problem. Literature shows that in all popular nature inspired algorithms, Particle Swarm Optimization is more efficient and robust for data classification task, but posed with limitations like more computational time and memory. Although, many approaches have been able to handle imbalance issue in data classification originally they have been proposed for smaller datasets. These limitations can be overcome by combining them with modern big data distributed solutions like Hadoop, Spark etc. Apache Hadoop is distributed framework for handling big data has become very popular which uses Hadoop Distributed File System (HDFS) as a main storage component. For processing big datasets, MapReduce is a parallel programming model which has increased the productivity in the form of big data processing. But this MapReduce is computationally expensive because it performs high disk reads and writes which degrades the performance of the model. To overcome these limitations many advanced technologies like Apche spark, Kafka took place of it [19].

From last decade, Apache Spark has become more popular because of its scalable and efficient data processing features as compared with Hadoop. It is one of the fast cluster computing engine that gives accurate measurability and fault detecting features as compared to Map-Reduce, but not similar to Hadoop's two stage disks based MapReduce paradigm [20]. In the research [21], for security analysis Apache spark has been used for big data processing. MapReduce and Spark have also used for implementing many algorithms like Genetic algorithms and Particle Swarm Optimization [22, 23].

Resilient Distributed Dataframe (RDD) is primitive component of Spark which makes Spark faster in processing the data. It is not using expensive disk access as all computations are in-memory which increases the performance of model in data processing. It consists of components like Spark core, Spark SWL and MLlib. Scheduling, memory management, disaster recovery and interaction with storage system are included in SPARK core [24].

Recently [25], assessed the performance of ensemble model and single classifier model using Map-Reduce technique on Hadoop and Apache spark. Authors in the research [26] have proposed model for predicting highway traffic accidents using SMOTE for handling imbalanced issue. For bio-informatics applications sample subset optimization techniques for imbalanced and ensemble learning problems have been proposed in the research [27]. Sampling techniques are used in combination with ensemble learning in data classification problems [28, 29]. Some improved under-sampling techniques have been proposed in researches [30-32] to deal with

imbalanced issue. In the research [33], a framework to tackle imbalanced issue in multi class datasets with novel version of SMOTE have been introduced. A Spark Based Mining Framework (SBMF) is proposed to address the imbalanced data problem in the research [34]. Bangare et al. [35, 36] worked in the disciplines of ML and IoT. The LRA-DNN approaches have been proposed by Shelke et al. [37]. Gupta et al. [38] demonstrated effective extraction techniques. CNN approaches were used by Awate et al. [39]. The network security work was proposed by Wu et al. [40]. Deep neural networks were employed well for brain tumor research by Ladkat et al. [41]. The authors of [42-44] assessed, employed and configured fresh version of CNN termed as Capsule Network for medicinal plant retrieval.

To minimize the impact of data level difficulty factors like border instances, two modules like Border Handling Module and Selective Border Instances sampling have been proposed.

Although, many researchers have worked on imbalance issue, main focus was on binary classification problems. This problem becomes more extensive when classification is for multi-class domain. It is more challenging because of certain difficulties such as complex relationship among the classes, data-level difficulty factors such as overlapping or small-disjunct classes and also huge volume of data. All these problems in the literature are less addressed.

## 3. PROPOSED APPROACH

In this paper, we have proposed SMOTE-PSO Ensemble model to deal with imbalanced big datasets.

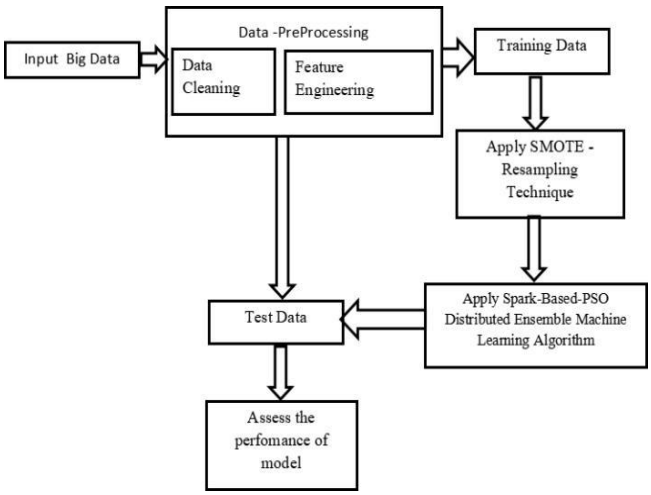


**Figure 1.** Overall architecture of proposed model

Figure 1 shows overall architecture of proposed method.

Proposed model works in three phases 1. Data Pre-Processing 2. Apply SMOTE Resampling technique 3. Apply Spark based PSO Distributed Ensemble Machine Learning algorithm. First step is very commonly used by almost every machine learning model. In this phase, data cleaning, where missing or null values and outliers are handled. Also feature engineering like transformations are applied. For example, categorical features are converted into numerical features using methods like Label Encoding. After data pre-processing, in second step, oversampling of minority class is performed using SMOTE technique. Limitation of SMOTE algorithm is that, it may lead to over-fitting of the model. In the third step, intelligent selection of subsets is performed using Particle Swarm Optimization (PSO). Overall pseudo-code for proposed model is given below:

Input: Imbalanced big dataset D.

Output: Ensemble model, Accuracy, Precision, F1-score, Recall and balanced dataset.

1. Apply data -pre-processing techniques.
2. Apply transformation techniques over categorical features.
3. Split data-set into training and test dataset.
4. Identify Majority and Minority classes from the training dataset.
5. Apply SMOTE on minority class and generate balanced dataset.
6. Call driver program on master node which splits the data on n number of partitions and send this to slaves in cluster.
7. Slave runs Spark-PSO module on partition which maximizes the value of fitness function. Here F1-score is used as fitness function.
8. All the optimal subsets are selected locally by slave or worker nodes are collectively selected by master node.
9. Model trained on optimal subset is used for testing the data.
10. Model is evaluated using performance metrices like precision, recall, accuracy and F1-score.

### 3.1 Spark-PSO module

This module works according to following steps:

1. Different sample subsets applying K-fold stratified cross validation are created from the dataset generated in step 5 which are considered as a particle and stored as an particle array. Here each particle in swarm is described as:

$$\mathrm{Xi} = \{s_1, s_{2,} s_3, \ldots s_n\} \quad (1)$$

$$\mathrm{Vi} = \{v_1, v_{2,} v_3, \ldots v_n\} \quad (2)$$

$$\mathrm{gbest} = \{\mathrm{gbest}_1, \mathrm{gbest}_{2,} \mathrm{gbest}_3, \ldots \mathrm{gbest}_n\} \quad (3)$$

2. Particle array is converted into RDD.
3. Initialize the PSO parameters like particle's current position, velocity, best known position, current fitness function value, particle's best fitness value in a driver program on master node.
4. All the particle's parameters like position, velocity are updated using fitness function at slave nodes in certain number of runs or iterations and global best values are sent back to the master node.

Here the fitness function is F-measure which is used to evaluate the velocity of particle which is updated using the following equation:

$$v_{i,d}(k+1)=\omega v_{i,d}(k)+\varphi_p r_p(l_{best}-x_{i,d}(k))+\varphi_g r_g(g_{best}-x_{i,d}(k)) \quad (4)$$

where, k is number of iterations, d is number of dimensions for each particle, ω, $\varphi_p$, $r_p$, $\varphi_g$, $r_g$ are the PSO parameters. These parameters ω is a constant called the inertia weight, $\varphi_p$ and $\varphi_g$ are the cognitive and social coefficients, respectively, and $r_p$ and $r_g$ are random numbers in the range [0, 1] and preset according to guidelines given in the research [27]. While $l_{best}$ and $g_{best}$ are local and global best positions, respectively.

The parameter values for PSO are shown in Table 1.

**Table 1.** PSO parameters

| Parameter | Value |
|---|---|
| No.of iterations | 120 |
| φp | 1.45 |
| φg | 1.45 |
| ω | 0.679 |
| rp ,rg | 0.018-0.982 |

Position of each particle is updated using following equation:

$$x_{i,d}(k+1)=\begin{cases}0 \text{ if random() } >= v_{i,d}(k+1)\\ 1 \text{ if random() } < v_{i,d}(k+1)\end{cases} \quad (5)$$

5. At master node the best-global values are determined and finally the best subset selected which is given the best fitness value and corresponding model is used as final classifier model.

6. Apply the models on test dataset generated through step 5.

### 3.2 Experimental settings

3.2.1 Datasets

Proposed model is applied on two big datasets. First dataset is KDD-CUP99 dataset and the second dataset is insect dataset. Details of this datasets are given in Table 2.

**Table 2.** KDD-CUP'99 dataset

| Sr. No | Dataset name | Dataset size |
|---|---|---|
| 1 | KDD-CUP'99 | 4,94,021 |
| 2 | Insect-Dataset | 3,55,275 |

A) KDD-CUP'99 is widely used publicly available dataset used for intrusion detection by classifying the network attacks. This dataset describes Denial-of-Service (DOS), Remote 2 Local (R2L), User to root and Probing attacks. Dataset contains 24 attack types in training and 14 more attack types in testing for total of 38 attacks. Detailed distribution of these classes is as shown in Table 3.

**Table 3.** Type of attack KDD-CUP'99 dataset

| Type of attack/connection | Number of instances | % |
|---|---|---|
| Normal | 972781 | 19.85 |
| DoS | 3883390 | 79.27 |
| R2L | 1106 | 00.02 |
| U2R | 52 | 00.001 |
| Probe | 41102 | 0.83 |
| Normal | 972781 | 19.85 |

**Table 4.** Number of mosquito species

| Type of mosquito species | Number of instances | % |
|---|---|---|
| Ae-aegypti-male | 67237 | 18.9 |
| Ae-aegypti-female | 101256 | 28.5 |
| Ae-albopictus-male | 11701 | 3.29 |
| Ae-albopictus-female | 21204 | 5.96 |
| Cx-quinq-male | 99557 | 28.02 |
| Cx-quinq-female | 54320 | 15.28 |

B) Insect-Dataset: This dataset is generated using signal processing of optical signals from smart trap used for catching the mosquitoes. This dataset is used for classifying three types of mosquitoes species for both male and female sex. These species are Aedes aegypti, Aedes albopictus and Culex quinquefasciatus which can spread various types of diseases. Distribution of these classes shown in Table 4.

3.2.2 Spark cluster configuration

Proposed model is tested on Spark cluster which consists of 6 nodes including one master node and 5 slave or worker nodes. Each node has one Intel Core™ i5 4 core CPU, 8GB RAM and 80 GB hard disk. All nodes are connected with each other through Ethernet network. As there are 4 cores for each node total 24 cores are available. Figure 2 shows the actual cluster configuration and Table 5 shows details of node configurations in a cluster.

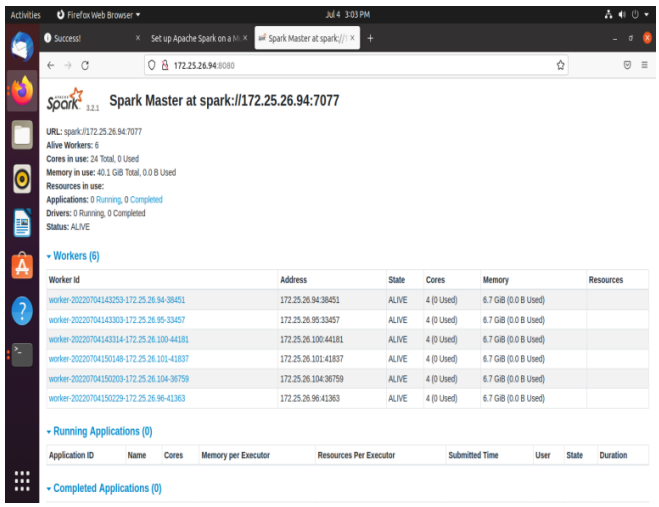


**Figure 2.** Spark cluster configuration

3.2.3 Parameters for comparison

For handling imbalanced big-data classification, to the best of our knowledge, no other Spark based approach using Particle Swarm Optimization has been found in the literature. Therefore, the results of the proposed model is compared with most popular traditional random under-sampling and random-oversampling based ensemble models.

**Table 5.** Cluster nodes configuration

| Sr. No | Item | Particulars |
|---|---|---|
| 1 | Node-Operating System | Ubuntu version= 20.04 |
| 2 | Spark | Spark version = 3.2.1<br>Hadoop Version = 2.7<br>Scala version = 2.11<br>HDFS blocksize = 128MB |

3.2.4 Evaluation metrics

To evaluate proposed model following evaluation metrices Eqns. (6), (7), and (8) are used.

$$F-measure = \frac{2*precision*recall}{precision+recall} \quad (6)$$

$$Precision = \frac{Tp}{TP+FP} \quad (7)$$

$$Recall = \frac{TP}{TP+FN} \quad (8)$$

where, TP is True Positive, FP is False Positive and FN is False negative values in prediction.

## 4. RESULTS AND DISCUSSION

Here, effectiveness of proposed model is verified using two multi-class imbalanced datasets. For KDD-Cup'99 dataset proposed model shows better performance over traditional sampling techniques. Ensemble model has obtained good results when it is combined with intelligent technique like Particle Swarm Optimization. Moreover, PSO-Splitbagging has reached the highest value on F1-measure 99% which is 3% higher for KDD-CUP'99 dataset and for insect dataset it is 92% which 4% higher than traditional RandomOverSampling (Ensemble-ROS-Bagging) technique as shown in Figures 3 and 4, respectively. Results of proposed model are satisfactory for the various evaluation measures than traditional technique. Model is implemented in distributed environment which shows overall processing time gets reduced as we increase the number of slave nodes as shown in Figure 5.

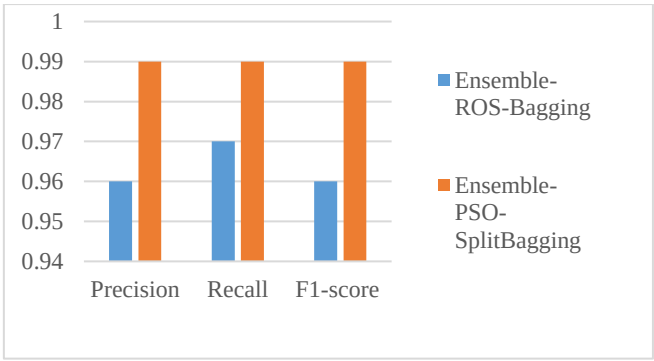


**Figure 3.** Performance evaluation for KDD-CUP dataset

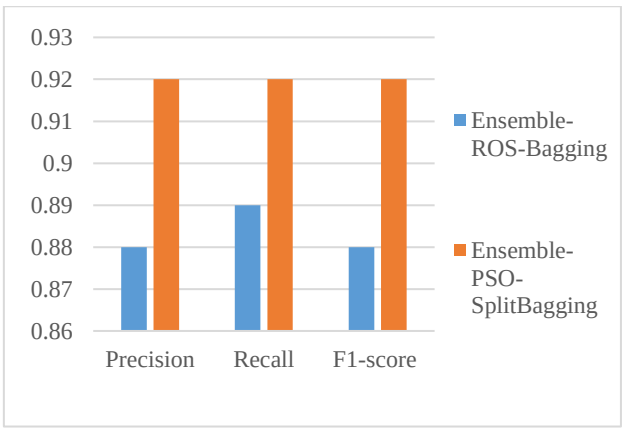


**Figure 4.** Performance evaluation insect dataset

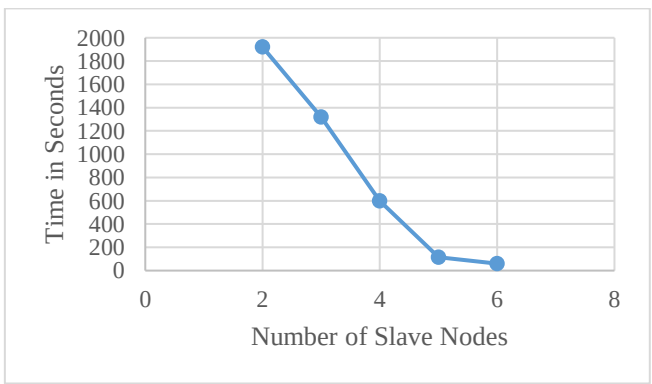


**Figure 5.** Computational time in spark-cluster

## 5. CONCLUSIONS

In this work, an Ensemble model and intelligent technique Particle Swarm Optimization is proposed for imbalanced multi-class imbalanced big dataset. This method is proposed to improve the overall classification performance as well as to reduce the computational time.

The experimental results on two multi-class datasets proves the effectiveness of the model. Different evaluation matrices are used to quantify performance of model. This method is implemented in distributed environment using popular Spark clustering which increases the computational efficiency with respect to time. This study has taken advantage of Ensemble learning and intelligent optimization technique for sampling the data to overcome the imbalanced issue.

## ACKNOWLEDGMENT

Authors thank to the 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 out research on above mentioned topic.

## REFERENCES

[1] Wolpert, D.H., Macready, W.G. (1997). No free lunch theorems for optimization. IEEE Transactions on Evolutionary Computation, 1(1): 67-82. https://doi.org/10.1109/4235.585893

[2] Woźniak, M., Grana, M., Corchado, E. (2014). A survey of multiple classifier systems as hybrid systems. Information Fusion, 16: 3-17. https://doi.org/10.1016/j.inffus.2013.04.006

[3] Kuncheva, L.I., Bezdek, J.C., Duin, R.P. (2001). Decision templates for multiple classifier fusion: An experimental comparison. Pattern Recognition, 34(2): 299-314. https://doi.org/10.1016/S0031-3203(99)00223-X

[4] Cruz, R.M., Sabourin, R., Cavalcanti, G.D., Ren, T.I. (2015). META-DES: A dynamic ensemble selection framework using meta-learning. Pattern Recognition, 48(5): 1925-1935. https://doi.org/10.1016/j.patcog.2014.12.003

[5] Harifi, S., Byagowi, E., Khalilian, M. (2017). Comparative study of apache spark MLlib clustering algorithms. In: Tan, Y., Takagi, H., Shi, Y. (eds) Data Mining and Big Data. DMBD 2017. Lecture Notes in Computer Science(), vol. 10387. Springer, Cham. https://doi.org/10.1007/978-3-319-61845-6_7

[6] Ali, A., Shamsuddin, S.M., Ralescu, A.L. (2013). Classification with class imbalance problem. Int. J. Advance Soft Compu. Appl, 5(3): 176-204.

[7] Liu, X.Y., Wu, J., Zhou, Z.H. (2008). Exploratory undersampling for class-imbalance learning. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 39(2): 539-550. https://doi.org/10.1109/TSMCB.2008.2007853

[8] Hido, S., Kashima, H., Takahashi, Y. (2009). Roughly balanced bagging for imbalanced data. Statistical Analysis and Data Mining: The ASA Data Science Journal, 2(5-6): 412-426.

https://doi.org/10.1002/sam.10061
[9] Galar, M., Fernandez, A., Barrenechea, E., Bustince, H., Herrera, F. (2011). A review on ensembles for the class imbalance problem: Bagging-, boosting-, and hybrid-based approaches. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 42(4): 463-484. https://doi.org/10.1109/TSMCC.2011.2161285
[10] Khoshgoftaar, T.M., Van Hulse, J., Napolitano, A. (2010). Comparing boosting and bagging techniques with noisy and imbalanced data. IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans, 41(3): 552-568. https://doi.org/10.1109/TSMCA.2010.2084081
[11] Fernández, A., Garcia, S., Herrera, F., Chawla, N.V. (2018). SMOTE for learning from imbalanced data: Progress and challenges, marking the 15-year anniversary. Journal of Artificial Intelligence Research, 61: 863-905.
[12] Lemaître, G., Nogueira, F., Aridas, C.K. (2017). Imbalanced-learn: A python toolbox to tackle the curse of imbalanced datasets in machine learning. The Journal of Machine Learning Research, 18(1): 559-563.
[13] Lunardon, N., Menardi, G., Torelli, N. (2013). R package ROSE: random over-sampling examples (version 0.0-3). Università di Trieste and Università di Padova, Italia.
[14] Kuhn, M. (2015). Caret: classification and regression training. Astrophysics Source Code Library, ascl-1505.
[15] Peng, C.Y.; Park, Y.J. (2022). A new hybrid under-sampling approach to imbalanced classification problems. Applied Artificial Intelligence, 36(1): 1975393. https://doi.org/10.1080/08839514.2021.1975393
[16] Piri, S., Delen, D., Liu, T. (2018). A synthetic informative minority over-sampling (SIMO) algorithm leveraging support vector machine to enhance learning from imbalanced datasets. Decision Support Systems, 106: 15-29. https://doi.org/10.1016/j.dss.2017.11.006
[17] Sidiq, S.J., Zaman, M., Butt, M. (2018). A framework for class imbalance problem using hybrid sampling. Artificial Intelligent Systems and Machine Learning, 10(4): 83-89.
[18] Wang, Q., Luo, Z., Huang, J., Feng, Y., Liu, Z. (2017). A novel ensemble method for imbalanced data learning: bagging of extrapolation-SMOTE SVM. Computational Intelligence and Neuroscience, 2017: 1827016. https://doi.org/10.1155/2017/1827016
[19] Sheshasaayee, A., Lakshmi, J.V.N. (2017). An insight into tree based machine learning techniques for big data analytics using Apache Spark. In 2017 International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICICT), Kerala, India, pp. 1740-1743. https://doi.org/10.1109/ICICICT1.2017.8342833
[20] Hadgu, A.T., Nigam, A., Diaz-Aviles, E. (2015). Large-scale learning with AdaGrad on Spark. In 2015 IEEE International Conference on Big Data (Big Data), Santa Clara, CA, USA, pp. 2828-2830. https://doi.org/10.1109/BigData.2015.7364091
[21] Lighari, S.N. (2017). Testing of algorithms for anomaly detection in big data using Apache spark. In 2017 9th International Conference on Computational Intelligence and Communication Networks (CICN), Girne, Northern Cyprus, pp. 97-100. https://doi.org/10.1109/CICN.2017.8319364
[22] Elsebakhi, E., Lee, F., Schendel, E., Haque, A., Kathireason, N., Pathare, T., Syed, N., Al-Ali, R. (2015). Large-scale machine learning based on functional networks for biomedical big data with high performance computing platforms. Journal of Computational Science, 11: 69-81. https://doi.org/10.1016/j.jocs.2015.09.008
[23] Ditzler, G., Hariri, S., Akoglu, A. (2017). High performance machine learning (HPML) framework to support DDDAS decision support systems: design overview. In 2017 IEEE 2nd International Workshops on Foundations and Applications of Self* Systems (FAS* W), Tucson, AZ, USA, pp. 360-362. https://doi.org/10.1109/FAS-W.2017.174
[24] Kato, K., Takefusa, A., Nakada, H., Oguchi, M. (2017). Consideration of parallel data processing over an Apache spark cluster. In 2017 IEEE International Conference on Big Data (Big Data), Boston, MA, USA, pp. 4757-4759. https://doi.org/10.1109/BigData.2017.8258533
[25] Meng, X., Bradley, J., Yavuz, B., et al. (2016). MLlib: Machine learning in Apache spark. The Journal of Machine Learning Research, 17(1): 1235-1241.
[26] Park, S.H., Kim, S.M., Ha, Y.G. (2016). Highway traffic accident prediction using VDS big data analysis. The Journal of Supercomputing, 72: 2815-2831. https://doi.org/10.1007/s11227-016-1624-z
[27] Yang, P., Yoo, P.D., Fernando, J., Zhou, B.B., Zhang, Z., Zomaya, A.Y. (2013). Sample subset optimization techniques for imbalanced and ensemble learning problems in bioinformatics applications. IEEE Transactions on Cybernetics, 44(3): 445-455. https://doi.org/10.1109/TCYB.2013.2257480
[28] Klikowski, J., Woźniak, M. (2020). Multi sampling random subspace ensemble for imbalanced data stream classification. In: Burduk, R., Kurzynski, M., Wozniak, M. (eds) Progress in Computer Recognition Systems. CORES 2019. Advances in Intelligent Systems and Computing, vol. 977. Springer, Cham. https://doi.org/10.1007/978-3-030-19738-4_36
[29] Zyblewski, P., Ksieniewicz, P., Woźniak, M. (2019). Classifier selection for highly imbalanced data streams with minority driven ensemble. In: Rutkowski, L., Scherer, R., Korytkowski, M., Pedrycz, W., Tadeusiewicz, R., Zurada, J. (eds) Artificial Intelligence and Soft Computing. ICAISC 2019. Lecture Notes in Computer Science(), vol. 11508. Springer, Cham. https://doi.org/10.1007/978-3-030-20912-4_57
[30] Koziarski, M. (2020). Radial-based undersampling for imbalanced data classification. Pattern Recognition, 102: 107262. https://doi.org/10.1016/j.patcog.2020.107262
[31] Vuttipittayamongkol, P., Elyan, E. (2020). Neighbourhood-based undersampling approach for handling imbalanced and overlapped data. Information Sciences, 509: 47-70. https://doi.org/10.1016/j.ins.2019.08.062
[32] Datta, S., Das, S. (2018). Multiobjective support vector machines: Handling class imbalance with pareto optimality. IEEE Transactions on Neural Networks and Learning Systems, 30(5): 1602-1608. https://doi.org/10.1109/TNNLS.2018.2869298
[33] Sleeman IV, W.C., Krawczyk, B. (2021). Multi-class imbalanced big data classification on spark. Knowledge-Based Systems, 212: 106598. https://doi.org/10.1016/j.knosys.2020.106598

[34] Abdel-Hamid, N.B., ElGhamrawy, S., Desouky, A.E., Arafat, H. (2018). A dynamic spark-based classification framework for imbalanced big data. Journal of Grid Computing, 16: 607-626. https://doi.org/10.1007/s10723-018-9465-z

[35] Bangare, S.L. (2022). Classification of optimal brain tissue using dynamic region growing and fuzzy min-max neural network in brain magnetic resonance images. Neuroscience Informatics, 2(3): 100019. https://doi.org/10.1016/j.neuri.2021.100019

[36] Bangare, S.L., Virmani, D., Karetla, G.R., Chaudhary, P., Kaur, H., Bukhari, S.N.H., Miah, S. (2022). Forecasting the applied deep learning tools in enhancing food quality for heart related diseases effectively: A study using structural equation model analysis. Journal of Food Quality, 2022: 6987569. https://doi.org/10.1155/2022/6987569

[37] Shelke, N., Chaudhury, S., Chakrabarti, S., Bangare, S.L., Yogapriya, G., Pandey, P. (2022). An efficient way of text-based emotion analysis from social media using LRA-DNN. Neuroscience Informatics, 2(3): 100048. https://doi.org/10.1016/j.neuri.2022.100048

[38] Gupta, S., Kumar, S., Bangare, S.L., Nuhmani, S., Alguno, A.C., Samori, I.A. (2022). Homogeneous decision community extraction based on end-user mental behavior on social media. Computational Intelligence and Neuroscience, 2022: 3490860. https://doi.org/10.1155/2022/3490860

[39] Awate, G., Bangare, S., Pradeepini, G., Patil, S. (2018). Detection of Alzheimers disease from MRI using convolutional neural network with tensorflow. arXiv preprint arXiv:1806.10170. https://doi.org/10.48550/arXiv.1806.10170

[40] Wu, X., Wei, D., Vasgi, B.P., Oleiwi, A.K., Bangare, S.L., Asenso, E. (2022). Research on network security situational awareness based on crawler algorithm. Security and Communication Networks, 2022: 3639174. https://doi.org/10.1155/2022/3639174

[41] Ladkat, A.S., Bangare, S.L., Jagota, V., Sanober, S., Beram, S.M., Rane, K., Singh, B.K. (2022). Deep neural network-based novel mathematical model for 3D brain tumor segmentation. Computational Intelligence and Neuroscience, 2022: 4271711. https://doi.org/10.1155/2022/4271711

[42] Pande, S., Chetty, M.S.R. (2019). Bezier curve based medicinal leaf classification using capsule network. International Journal of Advanced Trends in Computer Science and Engineering, 8(6): 2735-42.

[43] Pande, S.D., Chetty, M.S.R. (2021). Fast medicinal leaf retrieval using CapsNet. In: Bhattacharyya, S., Nayak, J., Prakash, K.B., Naik, B., Abraham, A. (eds) International Conference on Intelligent and Smart Computing in Data Analytics. Advances in Intelligent Systems and Computing, vol. 1312. Springer, Singapore. https://doi.org/10.1007/978-981-33-6176-8_16

[44] Pande, S., Chetty, M.S.R. (2018). Analysis of capsule network (Capsnet) architectures and applications. Journal of Adv Research in Dynamical & Control Systems, 10(10): 2765-2771.