Iran Journal of Computer Science (2020) 3:239–260
https://doi.org/10.1007/s42044-020-00061-3

REVIEW ARTICLE

Data stream classification: a review

Kapil K. Wankhade[1] · Snehlata S. Dongre[2] · Kalpana C. Jondhale[3]

Received: 9 January 2019 / Accepted: 7 May 2020 / Published online: 27 May 2020

Abstract
The tremendous amount of data is generated regularly through areas like networking, telecommunication, stock market, satellite, weather forecasting, etc. So, the classification process becomes important to extract knowledge from such a huge amount of data. The handling of concept drifting data stream and skewed data classification are the major issues and challenges in the data streams mining field. In the presence of concept drift, the performance of the learning algorithm always degrades. On the other side in skewed data problems, majority class accuracy always dominates minority class accuracy which generates the wrong result. This paper discusses the implemented methods which worked for skewed data and concept drifting data as well as their merits and demerits. This paper focused on metrics used in the classification process and issues that arise while learning with skewed and concept drifting data.

Keywords Concept drift · Skewed data · Classification · Clustering · Sampling · Semi-supervised learning

1 Introduction

Data stream classification is a huge growing field and has been gaining great attention from the research community from the last decade. Handling of concept drift and rare class data is the major issue in the data stream classification.

1.1 Background and motivation

The huge amount of data has been generated rapidly and very fast in today's growing world. Such a huge amount of data handling as well as analyzing is the most challenging task. In the research field, data stream mining has great attention because of the tremendous amount of data that have been generated from applications or industries such as networking, finance, the stock market, education, telecommunication, healthcare, weather forecasting and many more. The research industry has given great attention in solving data streams mining issues like scanning of data in a single pass, classification of such huge amount of data, suitable algorithm selection, concept drift, skewed data, performance in terms of time, accuracy and memory and learning approach. In the field of concept drift, it is very difficult for the classifier to detect time-changing classes or to classify time-changing data. Because, the occurrence of the concept drift is unpredictable or not fixed which leads to the inability to train the classifier with such data and results in performance degradation. On the other hand, in the field of skew data or rare class or imbalanced class distribution, it is very difficult for the classifier to learn small class events because of class overlapping which leads to misclassification of minority class samples. Hence, it significantly affects the classification accuracy of the classifier. For example, if there are two classes Class A and Class B in which Class A is the majority class and Class B is a minority class, a learning algorithm generates accuracy based on Class A samples without considering Class B samples and always tries to improve classification accuracy only considering Class A samples. Because of such a condition, it becomes a challenging task for the learning algorithm to optimize classification

✉ Kapil K. Wankhade
kaps.wankhade@gmail.com

Snehlata S. Dongre
dongre.sneha@gmail.com

Kalpana C. Jondhale
jondhale_kc@mgmcen.ac.in

1 Department of Computer Science and Engineering, Jhulelal Institute of Technology, Nagpur, MS, India

2 Department of Computer Science and Engineering, G H Raisoni College of Engineering, Nagpur, MS, India

3 Department of Electronics and Telecommunication Engineering, M. G. M.'s College of Engineering, Nanded, MS, India

accuracy without considering the relative distribution of every class. The condition becomes very critical when the dataset has multiple classes. In that case, the researcher has to give more attention to the minority class in the case of binary classification and multiple classes. To solve these issues, a great number of authors are attracted towards these fields and proposed the number of methods to solve these issues [1–19, 76–83].

1.2 Importance and significance

In the field of data stream mining, we cannot apply standard approaches for solving difficulties of data stream mining because day by day, new difficulties occur during the evaluation of data streams. While learning with data streams, single-scan learning has great importance. Data streams are always generated continuously, in huge amounts and very fast. During the classification of data, it must be scanned only once because of a tremendous amount, limited time and memory. If we are unable to process data samples in a single pass, then its complexity increases over time and memory which results in degradation of accuracy [20].

Data streams classification is done through two methods, namely single classification and ensemble classification. The single classification method is fast; it also takes less memory for computation, but as unknown patterns or unknown sample increase, its performance decreases. In vice versa, an ensemble method requires more time and memory but it performs well in the presence of unknown patterns. In the ensemble method, the generated output is the prediction of the different classifiers. For data stream handling, most of the researchers used an ensemble method because of easy implementation, ability to handle any type of data, and, most importantly, its high performance [27, 32, 42]. Bagging and boosting are types of ensemble methods. Most of the researches have tried ensemble method for data stream classification in which they used chunk-based [27, 28, 49] and windowing-based approaches for learning [21, 22]. These approaches play an important role during the evaluation process. Concept drift detection and handling are the major issues in the data stream mining field. The concept drift is the concept of data changes over time. Most of the researchers tried to solve concept drift problems with the help of different methods [23–26, 32].

1.3 Overall problems with the existing works

It is very difficult for the learning algorithm to identify rare class samples. Usually, traditional learning algorithms target a better accuracy rate, which creates an inherent result according to the majority class because the rare class has less impact on accuracy. As well as noisy data behave like rare class; hence, they are difficult to differentiate. Due to these issues, the rare class mining problem has attracted more and more attention from the research community. The examples of imbalanced data sets are fault diagnosis [84, 85], anomaly detection [86, 87], medical diagnosis [88–93], fraud detection in automobile insurance [94], text classification [95], direct marketing [96], e-mail foldering [97], face recognition [98], disaster information system [99], financial application [100], and so on. Usually, the misclassification cost of minority class samples is much higher than the misclassification of other class samples. An imbalanced class learning problem has been classified into the following categories,

a. Over-sampling: In the over-sampling method [101–105], the number of minority class samples increases to equalize class distribution which is used to solve the imbalanced class problem. A famous approach SMOTE [83] is used for over-sampling. This approach produces synthetic minority class samples by selecting the nearest minority class samples neighbors. Then, the generated new samples lie in between the minority samples and nearest neighbors samples. SMOTE shows a good result for solving the imbalanced class problem.

b. Under-sampling: Under-sampling method reduces the majority class samples and balances data distribution and tries to solve the imbalanced class problem. In this method, the majority class samples are selected randomly and combined with minority class samples as a training set. Many researchers used this method to overcome the class imbalance problem [94]. The performance of the over-sampling method is lower than that of the under-sampling method.

c. Ensemble: Ensemble has the rare ability to improve the classification accuracy of weak classifier [106, 107]. The ensemble is a set of individual classifiers whose predictions are combined to classify unknown samples. There are two types of ensemble method, namely bagging and boosting. Bagging is a bootstrap ensemble method in which training of each classifier is done by a training set of random redistribution of samples [108]. Every individual classifier in the ensemble is generated with a different random sampling of the training set. Bagging is effective on unstable learning algorithms where small changes in the training set result in a large change in final prediction. Boosting is an iterative method in which it improves the performance of weak classifier by focusing on hard samples which are difficult to classify [42]. Boosting uses voting strategies such as weighted voting, multiple voting, weighted majority, etc. which produces a final prediction.

d. Semi-supervised learning: Semi-supervised learning [109–112] is the approach that works on both labeled and unlabeled data. A semi-supervised learning method overcomes the disadvantage of supervised and unsupervised learning in the sense that supervised learning cannot work in the presence of unlabeled data and unsupervised cannot work in the presence of labeled data. In the semi-supervised learning method, the classifier is trained with labeled data and tested with unlabeled data to find out the class labels.

In this paper, we have discussed classification and clustering techniques used for data stream classification and rare class problem. Because, today, the tremendous amount of data is generated from real-life applications and it is important to extract knowledge from the generated data. But generated data are not fixed always; it changes over time, i.e., concept drift occurs in the data stream. In the presence of concept drift, the performance of the learning algorithm always decreases which results in wrong knowledge representation. So, handling of the concept drift is important in the field of data stream classification. Another issue is the rare class or imbalanced class or skewed class that has great attention from the research community. In the rear class problem, majority class accuracy always overwhelmed minority class accuracy which draws the wrong result. This issue affects the data stream classification process.

1.4 Paper outline

The rest of paper is organized as follows: a literature survey of the data stream classification is described in Sect. 2 with the evaluation metrics covered in Sect. 3; Sect. 4 includes a discussion on issues of concept drifting and skewed data classification; and Sect. 5 gives the conclusion.

2 Literature survey

Lots of work have been done by the research community to solve issues of data stream classification. This section mainly focused on the concept drift and skewed data. For solving both the issues, we have focused on supervised learning method, i.e., classification; unsupervised learning method, i.e., clustering; and hybrid methods, i.e., combination of both classification and clustering methods. Supervised learning models work in the presence of labeled data only. In the classification method, single classifier and ensemble classifiers work well in the presence of labeled data. Decision tree, SVM, neural networks, naive Bayes, random forest are some single classifiers. They work well in terms of time and memory but their performance decreases as new sample arrives. On the other hand, the ensemble is the combination of multiple classifiers and has good performance in terms of accuracy and requires more time and memory. Another approach for handling concept drift is unsupervised learning, i.e., clustering. This method works well in the presence of unlabeled data. In this method, strongly similar characteristic data points are grouped into one cluster and different characteristic data points in another cluster. In this way, it classifies data samples. *k*-means, DBSCAN, density-based clustering, grid and density-based clustering, convex hull are clustering methods that are used for data stream classification. The drawback of the clustering method is that they are unable to handle concept drift. The clustering method can handle both labeled and unlabeled data. For the skewed data classification, we have used another approach for improving performance, i.e., hybrid approaches. In the hybrid approach, we have surveyed techniques that use both supervised and unsupervised learning. It is the combination of classifier and clustering methods that uses the advantages of both methods to overcome the problem of skewed data. They are discussed in the following sub-sections as follows.

2.1 Concept drift

This sub-section covers classification and clustering methods used for data stream classification. Under the classification method, we have referred algorithms like dynamic weighted majority, online bagging, online boosting, OzaBagASHT, OzaBagADWIN, one-versus-all, AdaBoost, decision tree, UCFDT, ensemble tree, Hoeffding tree, recurring concept drift, entropy-based method, sliding window, particle swarm optimization, SVM, Bayesian networks, feature selection, etc. Under the clustering method, we have referred algorithms like CLUESTREAM, DBSCAN, DENSTREAM, rDENSTREAM, and convex hull.

2.1.1 Classification

A weighted classifier ensemble is proposed by Wang et al. [27] for mining concept drifting streaming data. In this proposed model, the ensemble classifier is trained with sequential data chunks of data stream instead of continuously updating a single model. In the ensemble classification method, maintaining the most up-to-date classifiers is not a good chance because discarding of earlier trained less accurate classifier may lead to loss of important information. This technique shows the importance of data distribution in the expiration of the old data which avoids overfitting and the concept conflicting problem. In the ensemble approach, weight is assigned to each classifier depending on the expected prediction accuracy of current test examples. This approach uses a single-partition and single-chunk approach for the classification of the data stream in which a single classifier is trained with a single data chunk to train the classifier. The disadvantages of the above-mentioned method have been removed by Masud et al. [28]. Authors have proposed a multi-partition and multi-chunk approach for classifying concept drifting data streams. In this approach, v-fold partitioning of the data is used in the r consecutive data chunks, and then these chunks are used to train a collection of v classifiers in the proposed method. This method reduces the expected classification error in comparison with the single-partition and single-chunk method. The dynamic weighted majority concept is used to maintain the size of an ensemble in dynamic weighted majority (DMW) [29] algorithm. This algorithm first creates a set of experts and assigns a weight to each of the experts. A newly arrived sample passed to the algorithm and gives prediction. The prediction depends on the weighted majority voting. The weight of an expert is decreased by a multiplicative constant when an expert wrongly classifies the samples. If the weight of an expert goes below the threshold, an ensemble removes that expert and is filled by a new one.

Online bagging and boosting methods are proposed by Oza et al. [30] which are used for handling the online huge amount of data. In the online bagging process, base model is trained with similar training examples with the same distribution which produces similar models that generate a good approximation to batch bagging. The standard batch bagging algorithm creates M base models with the help of given training dataset D having size N. The generated M models are trained with drawing random sampling with replacement policy from the original training data set D. The batch learning algorithms L_b trained each generated algorithm which uses bootstrap samples of size N. The online boosting algorithm [30] is designed in such a way that it works like AdaBoost. The weighted training sets are used to generate a sequence of base models by AdaBoost algorithm h_1, h_2, …., h_M. The model h_{m-1} misclassified training samples; half of the total weight is assigned to it when it generates the model h_m and the remaining half weight is assigned to correctly classified samples. In AdaBoost, based on the performance of the base model using the whole training set, the sample's weight is adjusted accordingly; while in online boosting, the sample's weight is adjusted according to the base model performance, depending on the samples which are processed earlier. The advanced version of the AdaBoost was proposed by Pelossof et al. [31], which updates the weights of a boosted classifier. While training the AdaBoost algorithm with x training samples, this online coordinate boosting algorithm minimizes the exponential loss function of AdaBoost.

The new bagging algorithms, OzaBagASHT and OzaBagADWIN, have been proposed by A. Bieft et al. [32] for evolving data streams. Adaptive Size Hoeffding Tree (ASHT) is used to improve the performance of the bagging method with increasing tree diversity. For the nth ASHT, the maximum allowed size of the tree is equal to double the maximum allowed size for the $(n - 1)$th tree. Adaptive Windowing (ADWIN) is working as a change detector and change estimator. When the change is detected, the ensemble dropped the worst-performing classifier and added a new classifier in the ensemble to maintain size. Law et al. [33] proposed a clustering-based approach using adaptive nearest neighbor for the classification of the data stream. It is a single classification-based approach. But it is unable to detect sudden concept drift. Aggrawal et al. [34] proposed the method, based on demand classification strategy. This method uses geometric time window and micro-clustering methods to classify a huge amount of data.

FACIL, an incremental rule learner, has proposed by Troyano et al. [35]. This proposed method uses partial instance memory which is based on the parameterized generalization and the samples which are residing on the border. This method can build and refine rules inconsistently. While refining new inconsistent rules, FACIL does not affect learning efficiency but it is unable to drop irrelevant features and recover the dropped features which are turned up as relevant later on. Hashemi et al. [36] has proposed One-Versus-All (OVA) decision tree classification method used for data stream classification. For solving the problem of k-class classification, One-Versus-All classifier uses binary classification where it distinguishes instances of one class from the instances of the remaining $(k - 1)$ classes. While classification, k binary classifiers are run and select who gives the highest confidence. Uncertainty handling and Concept-adapting Very Fast Decision Tree (UCVFDT) is proposed by Liang et al. [37] which is based on CVFDT and DTU concept to handle uncertain data streams. This proposed method especially focuses on the uncertain data found in the real-life applications.

Recurring of the concept drift is one of the issue in the data stream classification. To resolve this issue, the REcurring concept Drifts and Limited data method [38] is introduced where it uses k-means as a learning algorithm. In this method, concept clusters are generated in the tandem as leaves of incrementally building Hoeffding tree by the k-means algorithm. But the drawback of this method is that it consumes more space, fails to predict accurate recurring concept drift periods and fails to predict unknown concepts in advance. Li et al. [39] have proposed Mining Frequent Itemsets within a Transaction-sensitive Sliding Window method. This algorithm works efficiently for frequent itemsets mining in the field of data streams. This algorithm uses transaction sensitive sliding window for handling huge amount of data and data in a single scan. The performance of MSI-TransSW method is enhanced with the help of bit-sequence representation. Zliobaite et al. [40] have designed a new ensemble learning algorithm for handling concept drift. This method mainly focuses on variable new expert weight in three modifications to overcome the problem of relative weight reduction of the "right" experts.

Abdulsalam et al. [41] proposed streaming random forest for the classification of data streams. In this method, an entropy-based concept drift detection technique is used to handle the changing concept occurring in the data streams. Attar et al. [42] have proposed an ensemble method which is based on boosting and adaptive size windowing method.

This method works well in the concept changing environment. Sun et al. [47] proposed an incremental learning method. This method uses a modified Support Vector Machine (SVM) for incremental learning by selecting a kernel function to optimize the parameters. This algorithm improves the performance in terms of accuracy. This algorithm has tested for network data. Hosseini et al. [48] have exploited the occurrence of recurring concepts in the learning process and improved data streams classification process. This method uses a set of concepts which is used to detect re-occurrence of a concept using Bayesian and heuristic methods. Both these methods are used as an active classifier and weighted classifier in the classification process.

Feature selection algorithm plays a vital role to lighten the data mining process. In this paper, authors have proposed a feature selection algorithm, accelerated particle swarm optimization (APSO) designed especially for data stream mining [54]. This algorithm requires reasonable time for processing of data. This algorithm uses a wrapper-based feature selection model which retains the accuracy of each trial classifier built from a candidate feature subset and picks the highest possible fitness and deems the candidate feature subset as the choice output. In this paper [55], a construction of decision tree has been proposed for data stream which uses misclassification error for the derivation of new splitting criteria. An attribute is derived from the whole data streams with the attribute having the highest probability. This best-selected attribute is treated as a node. Using the Gini index, the splitting criteria are combined with this derived result. So, authors have proposed hybrid algorithms mDT, gDT and hDT and all are one-pass classifiers. The drawback of the mDT algorithm is that it cannot make a split and further grow of the tree. So, the gDT algorithm has better accuracy than the mDT in the presence of large data sets. mDT gives better accuracy when the tree is simple, but gDT performs better when the tree became more complex. hDT is defined as a disjunction of mDT and gDT and it produces an online tree with the splitting criteria.

The proposed class-based ensemble methods overcome the drawbacks of chunk-based ensembles. In this method, the occurred no of C classes in the stream, ensemble size keep as L micro-classifiers [64]. Therefore, C is the total number of ensembles in these proposed methods, CLAM and SCANR. In this proposed methods, r micro-classifiers are trained with data chunks using positive instances only. A decision boundary is formed around the training data during the training phase. The existing ensemble size is updated by the newly trained micro-classifier. *k*-means is used to form the decision boundary. The modified Hoeffding tree algorithm, Gaussian Decision Tree (GDT), has been proposed in [56]. This algorithm works better as compared to McDiarmid's bound. This proposed statistical method determines the best attribute in a node and selects the higher value of the split measure function; and with the help of Taylor's theorem and normal distribution properties, it selects high probability. The drawback of this method is that it cannot handle the concept drifting problem. This paper focused on class emerging or disappearing gradually [57]. For this purpose, the authors have proposed a class-based ensemble approach for class evolution (CBCE). In this approach, the base learner maintains for each class and when new data arrive, it updates base learner dynamically; then, class evolution can rapidly be adjusted by CBCE. With this issue, authors have proposed a novel under-sampling method for the base learner to handle the dynamic class-imbalance problem which is caused by the gradual evolution of classes.

Reacting to a concept drift is one of the most challenging tasks in the field of data stream learning. To solve this issue, Accuracy Updated Ensemble (AUE2), a new data stream classifier has been implemented by the authors. This AUE2 algorithm has the ability to react equally to different types of drifts. In the accuracy-based weighting mechanism, a block base ensembles with incremental Hoeffding Trees are used [58]. AUE2 is an extended version of AUE1. In this algorithm component classifiers has been updated incrementally which improves ensembles performance in the different types of concept drifting environment as well as it reduces chunk size impact. AUE2 requires less memory with average classification accuracy. This algorithm works well in the presence of concept drifts in a static environment. To solve the problems of recurring concepts learning in a dynamic feature space, Gomes et al. [59] have proposed a data stream classification system. They have proposed mining recurring concepts in a dynamic feature space (MReC-DFS) algorithm to solve this problem. With the help of the performance of the learning process and contextual information; this method has the ability to detect and adapt the concept changing the environment. In the proposed method, the stored models are combined in a dynamically weighted ensemble. The weight of the ensembles has been calculated on the basis of performance of the model and associated text information. This method reduces memory cost incurred on the associated past models as well as used incremental feature selection to reduce the combined feature space where, in turn, MReC-DFS stores only the most relevant features. This is an incrementally learning feature selection method where it dynamically determines the threshold between relevant and irrelevant features.

High-dimensional data stream mining has got great attention in the industrial system for efficient detection of system faults. To solve this issue, authors have proposed a method for fault detection in the high-dimensional time-changing data streams environment [60]. In the proposed method, an angle-based subspace anomaly detection (ABSAD) detects faults from low-dimensional subspace to high-dimensional datasets. ABSAD method discovers and selects subspace with low-dimensional and axis parallel which keeps a big portion of local outliers points. For each sample, the degree of variation from its neighbor sample is evaluated on the obtained subspace. The fault-relevant subspaces are selected with the help of evaluation of vectorial angles and the computation of the subspace projection of the local outlierness of an object and indicates whether it is abnormal or not. For online mode, this proposed method has extended with sliding window strategy for continuous monitoring of system states. In comparison with the ABSAD algorithm, a sliding window-based ABSAD is adaptive in nature with dynamically changing environment and significantly reduces type I error. Outlier detection becomes a challenging task while analyzing high-speed data streams. An incremental learning algorithm Local Outlier Factor (LOF) assumes memory which stores all the previous data points. This algorithm requires more cost on the memory. So to reduce the memory cost, the author has proposed a memory-efficient incremental algorithm for local outlier detection (MiLOF) method in the data streams environment as well as a more flexible version of MiLOF is MiLOF_F. Both these proposed algorithms perform nearly the same as incremental LOF but require fixed memory [61]. The proposed methods are mainly focused on the computation of LOF values but with limited memory. The proposed method keeps summaries dynamically in memory. This method detects outlier more accurately and it is more stable in the available memory size. In comparison with the iLOF, the proposed methods work better with respect to time and memory and also comparable accuracy.

E-tree, an ensemble tree, has been proposed by the author to classify each data stream record in time [62]. For the organization of all base models in an ensemble to speed up the prediction process, the proposed method uses the indexing structure. E-tree employs R-tree which is a height-balanced structure and reduces expected prediction time. This proposed method treats the ensemble as a spatial database. With the help of sharing patterns among all base classifiers, E-tree achieves sub-linear time prediction. Then, ensembles are built using decision trees without losing granularity. Every base classifier contains a set of decision rules and these decision rules are going to cover a rectangular area that is the decision area by considering the special object in special space. With the help of this, every ensemble model is transferred to a special database. With the help of basic three operations, namely search, insertion and deletion, E-tree automatically updates its size by adding new classifier and removes discarded old classifiers as well as adapting new patterns and concepts occurring in the time-changing data streams. The authors have proposed a KDE-Track method which is mainly focused on the spatio-temporal data streams and density estimation [63]. This proposed method models the data distribution as a set of resampling points with their estimated probability density function (PDF). An adaptive resampling strategy is used to control the number of resampling points. With linear time complexity, KDE-Track efficiently estimates the density function using interpolation on the kernel model which updates incrementally on the arrival of the new samples. The new interpolation KDE computes the PDF value of a new arriving data sample using interpolation of selected resampling points. This method uses linear time and space complexity for evaluating PDF of new observations with respect to the number of resampling points. Also for tracking of evolving density, the proposed method uses a sliding window strategy for estimating the density using most recent data samples.

Structural health monitoring (SHM) concept has been implemented by authors to diagnose and evaluate possible structural damages. For this purpose, an online convex–concave hull calculation method is used [65]. Initially, the proposed method finds convex hulls of the data set and after that, using vertices of the convex hulls, the concave hulls are formed. In this way, the misleading points in the intersections become the vertices of the concave hulls. The online clustering method updates old-concave hull using arrived new points. An SVM method is trained using the last updated vertices of the convex hull. For this purpose, authors have used online SVM method. Chen et al. [66] have focused on students educational experiences, opinions, feelings, concern about the learning process in this competitive environment. So, they have developed a workflow for integrating qualitative analysis and large-scale data mining models. To classify tweets which reflect students problems, authors have used multi-label naive Bayes classifier. In today's environment, data volume and feature space increase over time; so, continuous learning from doubly streaming is important. To resolve this problem, authors have proposed a new Online Learning with Streaming Features (OLSF) with its two versions, OLSF-I and OLSF-II [67]. This algorithm handles trapezoidal data streams. For continuous learning from the trapezoidal data streams, online learning and feature selection are combined by the proposed algorithm. Whenever instances arrive with the new features, the proposed classifier is updated by the existing features with the help of passive-aggressive update rule as well as it updates new features with the help of structural risk minimization principle. Then, feature sparsity has been applied for feature selection.

The distributed data sharing issue is generally found in the mobile phone network. For solving this issue, authors have proposed a Shadow Coding schema [68]. This schema utilizes a well-designed matrix set to preserve data privacy during the data collection, as well as ensures the data recovery which erases the add-in matrix set. A Shadow Matrix Computation is designed in Shadow Coding. This method models data as a matrix structure in base stations. Then, a shadow matrix pair set is used to merge it with data which preserves the privacy of data with low cost. Shadow matrix preserves data privacy during transmission and automatically erases itself when the data demander performs the public function. In the data stream mining, hashing-based and incremental algorithms are used to improve efficiency. In this paper, authors have discussed a stream mining application which is used for surveillance application [69]. In this application, multiple aerial and ground reconnaissance videos are collected by different cameras and processed in real time by a network of classifiers which are trained with this data to detect different high-level semantic features. As well as authors have also discussed the optimization problem of classifiers. In image stream mining method, when images arrive with the metadata, with the help of image mining system (IMS), these images can be processed in the real-time [70]. IMS extracts features of images and determines online classifier to make a prediction on extracted features using metadata of the image. The developed active learning algorithm achieves regret sub-linear in the number of images that have been already observed. Then, the proposed algorithm Uniformly Partitioned Contextual Experts (UPCE) creates a uniform partition of the context space and expert learns every set in the partition. In each time slot, UPCE follows the prediction of the expert with the help of the highest prediction accuracy.

In this sub-section, we have surveyed single classifier and ensemble classifier. In the single classification method, decision tree, UCVFDT, Hoeffding tree, SVM, Bayesian network, particle swarm optimization, and the sliding window are discussed. Every method has its advantages and disadvantages such as they require less time and memory but their performance decreases on the arrival of new data or in the presence of concept drift. Decision tree method is unable to handle concept drifting data streams as well as multi-class data. Hoeffding tree takes less time as compared with the decision tree but again is unable to handle concept drifting data. UCVFDT is a single classification method that handles concept drift but unable to classify multi-class data. SVM and Bayesian network methods are unable to classify concept drifting data as well as multi-class data and also they have low performance. Particle Swarm Optimization method is an optimization method which is unable to detect concept drift and to classify multi-class data. The sliding window is the data handling technique through which we can handle data very efficiently and reduces overall memory usage. Overall classification method cannot work on the unlabeled data and in the data stream, we cannot assume that generated data having labels. The advantages and disadvantages of the surveyed classification-based methods are tabulated in Table 1.

2.1.2 Clustering

Lughofer et al. [43], proposed a method based on divide and conquer method which is used to design dynamic split and merge method. This dynamic split and merge method is designed in such a way that it would handle time-changing data. During merging operation, the weighted average of the cluster centers has been calculated which is based on the homogeneity between two clusters and then merging operation is performed. Based on Bayesian information criteria, splitting is done by shrinking of a large cluster from the inside which separate clusters. This method has the ability to handle time-changing data and to provide a highly pure and clean cluster. The authors have proposed a density-based technique for clustering namely DENSTREAM [44], extension of DBSCAN and similar to CLUSTREAM. This proposed method determines both the structures of micro-cluster, i.e., potential and outlier. Whenever new data points arrive, the proposed method checks and try to combine it with both the structures either potential microcluster or outlier microcluster. If it is not combined with either of the structure of the microcluster, then it creates a new outlier microcluster. When clustering of the data point comes to arrival, then with the help of DBSCAN it generates output, i.e., final cluster. Based on some predefined criteria, some outlier micro-clusters are reshaped into potential micro clusters and the rest of the points are discarded. Discarding of the points leads to a loss of knowledge points.

Liu et al. have proposed rDENSTREAM algorithm [45]; it removes the drawback DENSTREAM algorithm. The proposed algorithm adds the review phase in the above steps on DENSTREAM. The generated result from macro-clustering of the potential microcluster is considered as classifier but the outlier micro-clusters are not discarded and they are stored in the buffer. Because the stored outlier micro-clusters are used in the relearning of the classifier which helps to improve the performance of the clustering in terms of accuracy as well as reduces the loss of knowledge points. Qian et al. [46] have used combined approaches of VFDT and clustering method. In this method, clustered data points cannot be temporarily classified into n class and are based on clustering output; the proposed method generates new branches of VFDT or replaces original ones. A novel class detection and ensemble classification technique has been implemented by Masud et al. [49]. This method specially focuses on time-constraint which requires classifying testing examples. This method uses the ensemble hybrid approach of classification with clustering.

Adapted *k*-means method is clustering-based technique proposed for data stream mining [50]. This method is simple, generated results are easy to interpret but it requires more space and is sensitive to the noise. In this method, data points are divided into chunks; then each of the chunks is

Table 1 Advantages and disadvantages of classification-based methods

S. no	Methods/algorithms/applications	Advantages and disadvantages
1	Weighted classifier ensemble [27]	Uses single-partition and single-chunk method Works on stationary data Requires less time and memory But unable to handle concept drift and data stream
2	Multi-partition and multi-chunk [28]	Reduces classification error in comparison with single-partition and single-chunk method Unable to handle a huge amount of data and abrupt concept drift
3	Dynamic weighted majority [29]	Assigns a weight to each classifier and after modification of weight if it goes below the threshold, removes that classifier Unable to classify a huge amount of data as well as to handle abrupt concept drift
4	OzaBag and OzaBoost [30]	Handles online data but in the presence of concept drift performance degrades
5	OzaBagASHT and OzaBagADWIN [32]	Requires less time and memory but unable to handle abrupt concept drift
6	FACIL [35]	Unable to drop irrelevant features and recover the dropped features which are turned up as relevant later Unable to handle a huge amount of data as well as concept drifting data
7	OVA [36]	Requires more time and memory Performance degrades in the presence of concept drifting data stream
8	UCVFDT [37]	Requires less time and memory but unable to handle abrupt concept drift
9	REcurring concept drift and limited data method [38]	Consumes more space Fails to predict accurate recurring concept drift periods Fails to predict unknown concepts in advance
10	Mining Frequent Itemset [39]	Uses frequent itemset mining for data stream Unable to handle concept drift
11	Streaming Random Forest [41]	Handles data stream But unable to handle abrupt concept drift
12	Fast and light classifier [42]	Requires less time and memory with moderate accuracy Unable to handle abrupt concept drift
13	Modified SVM [47]	Designed for network data Unable to handle concept drifting data
14	Recurring concept method [48]	Based on a single classification method Requires less time and memory Unable to detect concept drift
15	APSO [54]	Requires reasonable time for processing of data Performance degrades in the presence of concept drifting data
16	Construction of decision tree model [55]	Requires less time for the selection of splitting attribute Unable to handle abrupt concept drift
17	CLAM and SCANR [64]	It overcomes the drawback of chunk-based method Unable to handle abrupt concept drift
18	Gaussian decision tree (GDT) [56]	Requires less time and memory Unable to handle concept drifting data stream
19	CBCE [57]	Handles dynamic class-imbalance problem Performance degrades in the presence of abrupt concept drift
20	Accuracy updated ensemble (AUE-2) [58]	Requires less time Handles types of concept drift Average classification accuracy
21	MReC-DFS [59]	Requires less space Handles concept drifting data
22	ABSAD [60]	Reduces error rate Detects faults from low-dimensional sub-space to high-dimensional datasets
23	MiLOF [61]	Works better concerning with time and memory and comparative accuracy Designed for outlier detection
24	E-tree [62]	Automatically updates size Adapt concept changing data

Table 1 (continued)

S. no	Methods/algorithms/applications	Advantages and disadvantages
25	KDE-Track [63]	Mainly focused on the Spatio-temporal data streams and density estimation Uses linear time and memory
26	Convex–concave hull calculation method [65]	Reduces error rate Unable to handle concept changing data
27	Multi-label Naive Bayes classifier [66]	Designed for solving students problem
28	Online learning with streaming feature (OL_{SF}—I and II) [67]	Handles trapezoidal data streams Updates with the arrival of new features
29	Shadow Coding Schema [68]	Designed to solve data sharing issue in the mobile phone network
30	UPCE [70]	Creates uniform partition of the context space Performance degrades in the presence of concept drifting data stream

separately clustered in weighted centers. In the final stage, the centers are again clustered and they generate the final clusters. STREAM is an improved version of this adapted *k*-means algorithm [51]. This STREAM algorithm uses a sliding window and produces the center at each particular stage. This proposed method does not require the number of clusters in advance to specify which are expecting at the time of output. Instead, it evaluates the performance of the clustering algorithm with the help of combination of the sum of squared distances and the number of the clusters used in the processing of the data points. The proposed CLUSTREAM [52] algorithm has two phases during clustering of the data points, namely online and offline. Micro-clusters are generated during the online phase as output. Incoming data are absorbed by the micro-clusters and based on the time window, it clusters data points. Macro-clusters are generated during the offline phase which is based on cluster number, duration, etc. using the *k*-means method.

A novel DD stream method [53] is represented by the combination of grid and density-based methods. The motivation behind this is that incoming infinite data points are mapped into the finite number of cells forms grid-like structure which is based on the grid density clustering that is performed. The issue of loss of boundary points is overcome by the DD stream [53] using DCQ means algorithm. DCQ algorithm uses two phases which aims to no loose boundary points and provides good-quality cluster. The incoming data points are mapped into grid cells in the first phase. With density decay factor and eigenvector, it calculates the distance between data points and neighboring grid cell. In the second phase, DCQ means is used to determine the distance between boundary data points and grid cluster centers. After that, it adds that point to the minimum distance grid.

For the evaluation of the multiclass novel class detection in the data stream mining [71], the authors have proposed a new method which can deal with unsupervised learning. The proposed method can generate new patterns apart from true classes. The authors have proposed a micro-cluster-based approach, shared-density-based re-clustering approach for data stream clustering [72]. Generally, micro-cluster aggregates the information of many data points in a defined area and represents local density. During clustering, a shared density graph concept has been used which explicitly captures the density of the original data between micro-clusters and then re-clusters micro-cluster using graph. In this method, it directly determines the density in the shared region between micro-clusters.

This paper addresses issues like one-class learning from time-changing data streams and concept summarization learning [73]. Authors have proposed UOLCS—uncertain one-class learning and concept summarization framework, for uncertain data streams. This proposed method deals with uncertain data and summarizes concept in one-class data streams. This proposed algorithm works in two phases; in the first phase, the uncertain one-class classifier is constructed with the help of uncertain information into one-class SVM learning phase which is used to construct the more accurate classifier. Then in the second phase, user's concept drift is summarized from data streams by developing support vector-based clustering method which is based on history chunks, i.e., extended *k*-means clustering method applied on support vectors from every uncertain one-class classifier which is derived from history chunks for summarization of concept. In this paper, a proposed method compacted object sample extraction (COMPOSE) framework is used for focusing on the extreme verification latency in an initially labeled streaming environment, i.e., learning of concept drifting data stream in a non-stationary environment which provides only unlabeled data after initialization [74]. This proposed framework follows three steps: (1) first for the training of the semi-supervised learning, it combines initial labeled data with unlabeled data, (2) in the second step, for each class, it constructs a generalized convex hull α shapes to provide tight envelope around those data which are representing current class conditional distribution, and (3) in the last step, it shrinks α shapes and extracts core support samples which are representing a geometric center of each class distribution. This process iteratively works for new arriving unlabeled data. This algorithm is applicable only for gradual concept drifting data streams.

In this sub-section, clustering-based, i.e., unsupervised learning method has been surveyed like CLUESTREAM, DBSCAN, DENSTREAM, rDENSTREAM, and convex hull. The clustering method works on unlabeled data. The above-mentioned methods are unable to handle concept drifting data stream as well as a huge amount of data. Grid and density-based clustering method can handle the data stream. But on the arrival of the new samples, performance degrades. The advantages and disadvantages of the surveyed clustering-based methods are tabulated in Table 2.

2.2 Skewed data

This sub-section includes imbalanced learning algorithms like ensemble methods, hybrid methods and some state-of-the-art algorithms like ensemble methods—bagging, boosting, SMOTEBoost, RUSBoost, oversampling, undersampling, neural network, fuzzy systems, multi-layer perceptron, optimization, one-versus-all data balancing, Mahalanobis–Taguchi System, incremental clustering, *k*-means.

2.2.1 Ensemble method

An ensemble-based approach like Meta Imbalanced Classification Ensemble (MICE) is improved by choosing the optimal threshold based on the posterior probabilities [113]. It partitions the majority group firstly and then integrates the meta-information from the meta-learner. It improves the performance of the class imbalance problem based on two important steps. The first step is the partition with transformed features and second is ensemble learning with logistic regression. Then, RUSBoost, a new sampling boosting algorithm, has been proposed by Seiffert et al. [114]. Random Under Sampling Boosting method is faster and simpler than SMOTEBoost and uses a random under-sampling method. This algorithm has faster training time and favorable performance. For imbalanced biomedical data classification, Oh et al. [115] have proposed an active example selection (AES) algorithm with the ensemble method. This method builds a model by starting with a small balanced subset of training data and then it trains classifier iteratively through adding useful examples into the training set. Because of several iteration it requires for model training and example

Table 2 Advantages and disadvantages of clustering-based methods

S. no	Methods/algorithms/applications	Advantages and disadvantages
1	Split and merge method [43]	Provides highly clean and pure clusters Good accuracy Unable to handle a huge amount of data as well as abrupt concept drift
2	DENSTREAM [44]	Handles data stream Discard the points that leads to loss of knowledge points Unable to handle concept drifting data
3	rDENSTREAM [45]	Overcome drawback of DENSTREAM method Improves accuracy and reduces the loss of knowledge points Unable to handle concept drifting data stream
4	VFDT and Clustering method [46]	Requires less time and memory Moderate accuracy Unable to handle concept drifting data streams
5	Adapted *k*-means [50]	Ease of result interpretation Requires more space and sensitive to the noise Unable to handle a huge amount of data as well as concept drift
6	STREAM [51]	Removes drawback of the adapted *k*-means algorithm Unable to handle a huge amount of data as well as concept drift
7	CLUSTREAM [52]	Sensitive to the noise Unable to handle a huge amount of data as well as concept drift
8	DD stream method [53]	Handles a huge amount of data But performance degrades in the presence of concept drifting data
9	UOLCS [73]	Handles uncertain one-class data stream Performance degrades in the presence of a multi-class problem
10	COMPOSE [74]	Designed for handling gradual concept drifting data stream Takes more time for iteratively working for new arriving unlabeled data Unable to handle the abrupt concept drifting data stream

selection steps, AES requires more computational time. Despite this, AES performs well in imbalance environment. The drawback of this method is that it requires high cost and output of the classifier depends on initial training examples.

Yang et al. [116] have proposed a data sampling technique, i.e., sample subset optimization (SSO). For optimization purpose, it uses a genetic algorithm as well as the ensemble learning algorithm. SSO is employed as an under-sampling method for identifying a subset of highly discriminative samples in the majority class. In ensemble learning, SSO is utilized as a genetic ensemble technique where multiple optimization subsets of samples from each class are selected for building an ensemble classifier. So for this purpose, it requires more base classifiers and huge cost in terms of time and memory, as well as accuracy, is not up to the mark. Ditzler et al. [117] have proposed two ensemble methods, first combining SMOTE with Learn++.NSE algorithm for concept drift for learning imbalanced data and second is by replacing SMOTE from Learn++.NSE-SMOTE with a sub-ensemble method which makes strategic use of minority class data and then replacing Learn++.NSE. It's class independent error weighting mechanism with a penalty constraint enforces the algorithm to balance accuracy of all classes. The primary novelty of this approach is determining the voting weights for combining ensemble members based on each classifiers time and imbalance adjusted accuracy on current and past environments.

A cost-sensitive boosting algorithm has been proposed by Sun et al. [118], which improves the classification performance of imbalanced data, possesses multiple classes. The cost matrix is often unavailable for a problem domain, so because of this, genetic algorithm is applied to search the optimum cost setup of each class. AdaC2.M1 algorithm reduces its weight and updates parameter to minimize the overall training error of combined classifier which also considers misclassification cost. By setting up different cost values, AdaC2.M1 can adjust the data distribution and bias the learning focus among classes. One class classifier ensemble uses samples from a single distribution to derive a decision boundary [119]. After deriving decision boundary, this method employed on the minority class to boost up its recognition value. The proposed approach combines several one-class classifiers using random subspace approach and a diversity method is used to select members of the committee. Wang et al. [120] proposed OOB (over-sampling-based online bagging) and UOB (under-sampling-based online bagging) methods. They have used improved resampling strategy inside OOB and UOB. UOB is better at recognizing minority class examples in static data streams. OOB is more robust against dynamic changes in class imbalance datasets. Again, they have proposed two ensemble methods that maintain OOB and UOB with adaptive weights for final prediction, i.e., WEOB1 and WEOB2.

2.2.2 Hybrid approach

Apart from the above-mentioned approaches, hybrid methods are also used for solving the rare class problem. Ahumada et al. [121] have proposed a supervised and unsupervised-based hybrid approach. With the help of clustering method, recursive partitioning of the majority class (REPMAC) approach recursively splits the majority class into the several subsets and then creates decision tree unless and until the resulting sub-problem is balanced or easy to solve. SVM is also used to build a balanced classifier and decision tree. The disadvantage of this method is that it deals with the only binary class problem. Jeatrakul et al. [122] have proposed One Against All with Data Balancing (OAA-DB) algorithm which combines the multi-binary classification technique used to enhanced the classification performance in the case of multi-class imbalanced data without reducing the overall classification accuracy. OAA-DB is a combination of SMOTE and CMTNN (Complementary Neural Network) where CMTNN is applied as an under-sampling technique; while SMOTE is used as an over-sampling technique.

A variant of the supervised learning neural network from the Adaptive Resonance Theory (ART) family, i.e., Fuzzy ARTMAP (FAM) [123] which is equipped with a conflict-resolving facility is proposed to classify an imbalanced dataset that represents a real problem in the semiconductor industry. The FAM model is combined with Dynamic Decay Adjustment (DDA) algorithm to form hybrid FAM-DDA network. Cao et al. [124] proposed two ideas, first is diverse random subspace ensemble learning with evolutionary search and second is to improve the performance of the neural network on multiclass imbalanced data. An evolutionary search method is focused on optimization of multi-classification cost under the guidance of imbalanced data measured. The diverse random subspace ensemble uses a minimum overlapping mechanism to provide diversity to improve the performance of the learning algorithm and optimization of neural network.

2.2.3 Other methods

Most of the researchers have also tried different approaches to solve an imbalanced class problem like Fu et al. [125] have proposed certainty-based active learning method in which for every unlabeled sample, the proposed method utilizes only local behavior rather than the entire data set and generates the error minimization hypothesis. The proposed method tries to enhance the prediction of the hypothesis and it can determine the query probabilities properly. Antwi et al. [126] have proposed a method which interplays in between concept drift detection and imbalanced data sets to ensure reliable results. The proposed method considers all performance measures of confusion matrix but it only concentrates on labeled data. Emerging patterns and decision tree has been proposed by Alhammady et al. [127] to solve the rare class problem. Emerging patterns are those itemsets whose support in one class is higher than that of support in other classes. Emerging pattern and decision tree employ the power of emerging patterns to improve the quality of rare-class classification.

Wang et al. [128] have proposed low granularity classifier which handles easily rare class events on data streams. RBC-1 does not generate enough rules for the rare class, because it is likely that there are too few or even no records belonging to the rare class in the sliding window. In RBC-2, instead of using the same sliding window for all classes, the proposed method creates a fixed-size window for each class. Then, it merges all windows of different classes to form a uniform window and compute support and confidence for rules over the uniform window. Orriols-Puig et al. [129] have proposed the Michigan style Learning Classifier System (LCS). This is an online machine learning technique that incrementally evolves distributed sub-solution and individually solves a portion of the problem space. This method mainly focused to examine the effect of class imbalance on different LCS components. The analysis also focuses on XCS which is the most relevant Michigan style LCS. Due to complex LCS architecture, it decomposes the problem of learning from unbalanced domains in several elements and derives facet wise models for it. He et al. [130] proposed a rare category characterization (RACH) algorithm by exploring the compactness property of the rare categories. This method is based on an optimization framework which encloses rare class samples by a minimum-radius hyperball. RACH uses a filtering method for pre-processing of data and then high-dimensional feature space. Because of more complex shape, RACH becomes more flexible and may produce a more accurate result.

Hospedals et al. [131] have proposed a generative and discriminative model for the imbalanced class problem. In this approach, a unified active learning model jointly discovers new categories and learns to adapt query criteria online for classification of samples. Biased mini-max probability machine [132] does not remove or duplicate data. It forms an explicit connection between the classification accuracy and the bias, i.e., it provides an elegant way to incorporate a certain bias into the classification by directly controlling the classification accuracy. Su et al. [133] proposed Mahalanobis–Taguchi Systems (MTS) for the imbalanced data problem. Mahalanobis–Taguchi Systems is a diagnostic and forecasting technique for multivariate data. It establishes a classifier by constructing a continuous measurement scale rather than directly learning from the training set. MTS employed to analyze the radio frequency inspection process which is used to judge whether or not a classifier has the balanced ability to predict the positive and negative samples. Diamantini et al. [134] have proposed statistical decision theory-based model in which labeled vector quantizer model, where the gradient of the average risk can be calculated. In the Bayes vector quantizer, a stochastic gradient algorithm can be derived from it to find a local optimal approximation of the Bayes decision border when class probabilities are unknown. This method mainly focuses on the two-class problem and not a multi-class problem.

Castro et al. [135] have proposed a new cost-sensitive CSMLP algorithm which improves the discrimination ability of two classes MLP. The CSMLP formulation is based on a joint objective function that uses a single cost parameter to distinguish the importance of class errors. The learning rule extends the Levenberg–Marquardt's rule which ensures the computational efficiency of the algorithm. The advantage of this algorithm is that it handles noisy data. Kwak et al. [136] have proposed an online fault detection algorithm which is based on incremental clustering. The proposed method accurately finds wafer faults over in many class distribution that skews and efficiently processes massive sensor data in terms of reduction in the required storage.

Zhang et al. [137] have proposed cost-free learning (CFL) approach which seeks optimal classification result without requiring any cost information in class imbalance problem. Based on the information theory, CFL maximizes normalized mutual information of the target and the decision output of the classifiers. The proposed strategy can balance the error rates and rejects accordingly and automatically. Das et al. [138] have proposed probabilistic oversampling methods RACOG and wRACOG to synthetically generating and strategically selecting new minority class samples. The proposed algorithm is used for the joint probability distribution of data attributes and Gibbs sampling to generate new minority class samples. wRACOG selects those samples that have the highest probability of being classified by the existing learning model.

In the above-mentioned sub-section, we have surveyed algorithms for handling skewed data. The methods undersampling, oversampling, and SMOTEBoost are basic methods but performance degrades in the presence of multi-class data as well as unavailability of sufficient data for learning. Again, ensemble method improves the overall performance but detection rate of minority class is low. A neural network, fuzzy system, multi-layer perceptron are supervised learning methods, but their performance decreases in the presence of unlabeled data and multi-class data. Incremental clustering and *k*-means are sensitive to the noise and their performance decreases in the presence of noise. Mahalanobis–Taguchi System works on the small data. In the presence of a huge amount of data, its performance degrades. Other researcher uses hybrid approaches but still, they are unable to handle a huge amount of data. Rare class problem is the application of data stream classification. It contains both labeled and unlabeled data. The advantages and disadvantages of the surveyed method of the rare class problem are tabulated in Table 3.

3 Evaluation metrics

Day by day, research community tries to develop new algorithms to handle imbalanced learning problem. But the evaluation metric plays an important role in the assessment of calculating the performance of the algorithm. This section has focused on some standard evaluation metrics.

3.1 Confusion matrix

In the field of machine learning or statistical classification of data, confusion matrix plays a vital role. A confusion matrix is a tabular representation of the performance of an algorithm. The classified data are distributed or categorized into two classes as predicted class and actual class. The column in the confusion matrix is represented as predicted class and the row represented as an actual class and vice versa. A confusion matrix is shown in Table 4.

As per the tradition, in the field of machine learning, the performance of any algorithm is measured with the help of accuracy and error rate metrics. Let us assume, in the two-class classification problem, X_c (N, P) be the instances which belong to an actual class and Y_c (N, P) be the instances which belong to a predicted class, respectively. With the help of this, accuracy and error rate of any algorithm are calculated as

$$\text{Accuracy} = (\text{TP} + \text{TN})/(\text{total no. of } P \text{ samples} + \text{total no. of } N \text{ samples}) \tag{1}$$

$$\text{Error rate} = 1 - \text{Accuracy}. \tag{2}$$

For example, consider an intrusion detection dataset having 200 samples, out of 200, 175 samples are predicted as normal and 13 samples are predicted as intrusion as shown in Table 5. Normal is a majority class having 90% of data and intrusion class is a minority class having 10% of data. If any classifier applies to this dataset, it would give near about more than 90% accuracy. This means that the majority class accuracy suppresses minority class accuracy, i.e., minority class having 0% accuracy.

If some noisy data are added in this dataset, then the accuracy of the applied algorithm would be different. So, accuracy metrics are highly sensitive to the changes in the dataset. Finally, this is to say that the accuracy metric does not provide satisfactory results as per the classifier's behavior with respect to the type of classification required [42, 139–142].

From Eq. (1), the accuracy metric uses both the columns data of confusion matrix. Therefore, as distribution of samples over the class of dataset varies, the performance of classifier also changes with respect to accuracy. Again it is very difficult when comparing the performance of the classifier with respect to accuracy with other classifier using different datasets. In the imbalanced class learning scenario, the accuracy metric fails because it is sensitive over data distribution as well as relative analysis of classifier is also not possible. Therefore many researchers have used different metrics other than accuracy in the research field of imbalanced data learning. These metrics are as follows,

$$\text{Precision} = \text{TP}/(\text{TP} + \text{FP}) \tag{3}$$

$$\text{Sensitivity or Recall or True Positive Rate} = \text{TP}/(\text{TP} + \text{FN}) = \text{TP}/P \tag{4}$$

$$\text{Specificity or True Negative Rate} = \text{TN}/(\text{TN} + \text{FP}) = \text{TN}/N \tag{5}$$

$$F - \text{measure} = \frac{(1 + \beta)^2 \cdot \text{Precision} \cdot \text{Recall}}{\beta^2 \cdot \text{Recall} + \text{Precision}} \tag{6}$$

where β is a coefficient to adjust the relative importance of precision versus recall ($\beta = 1$ usually)

$$G - \text{mean} = \sqrt{\left(\frac{\text{TP}}{\text{TP} + \text{FN}}\right) \times \left(\frac{\text{TN}}{\text{TN} + \text{FP}}\right)}. \tag{7}$$

Precision is referred to as positive predictive value, i.e., it calculates correctly classified negative examples out of a total number of negative examples. Recall it is also called as sensitivity or true positive rate, i.e., it calculates a total number of positive examples of positive class classified correctly. Precision and Recall possess an inverse relationship with each other. Both metrics are not sensitive to data distribution as well as both works better in imbalanced learning field and evaluates the performance of classifier effectively. Precision does not have confidence in how many positive examples are classified incorrectly. Also, recall does not give any information about how many examples were incorrectly classified as positive.

F-measure also gives more importance to the functionality of the classifier than the accuracy measure. *F*-measure is defined in Eq. (6), and it uses both precision and recall metrics. β coefficient is used as a weighted parameter to bias solution. It is a weighted average of recall and precision. It is used as an effective measure to calculate classifiers performance.

G-mean is another metric used to find classifiers performance. It uses both positive accuracy and negative accuracy.

Table 3 Advantages and disadvantages of surveyed methods of rare class problem

S. no	Methods/algorithms/applications	Advantages and disadvantages
1	MICE [113]	Improves performance of class imbalance Takes more time
2	RUSBoost [114]	Faster and simpler than SMOTEBoost Having faster training time and favorable performance Reduces loss of knowledge points in the under-sampling method
3	AES [115]	Requires more computational time Performs well in imbalance environment Requires high cost and output of the classifier depends on initial training examples
4	SSO [116]	Uses under-sampling method and genetic algorithm Requires more base classifiers and huge cost in terms of time and memory, as well as accuracy is not up to the mark
5	SMOTE with Learn++.NSE and Learn++.NSE-SMOTE [117]	Balances accuracy of all the classes Requires more space
6	AdaC2.M1 [118]	Handles multiple classes Minimizes training error
7	OOB and UOB [120]	UOB works better at recognizing minority class samples in the static data stream OOB works better in dynamic changes in class imbalance datasets
8	REPMAC [121]	Recursively splits the majority class into the several subsets and then creates a decision tree unless and until the resulting sub-problem are balanced or easy to solve Requires more time and memory Deals with only binary class problem
9	OAA-DB [122]	Uses both undersampling and oversampling methods Requires more time and memory Unable to handle huge amount of data
10	FAM [123]	Used in the semiconductor industry Handles imbalanced data in the stationary data stream
11	Ensemble learning with evolutionary search [124]	Optimize multi-classification cost Requires more time and memory Unable to handle time-changing data
12	Certainty-based active learning method [125]	Minimizes error Works on the static data set
13	Emerging patterns and decision tree-based method [127]	Emerging patterns improves quality of rare-class classification Works only on the stationary data stream
14	Low granularity classifier [128]	Handles rare class events easily in the data streams Uses fixed-size sliding window Unable to handle time-changing data streams
15	LCS [129]	Works well on the rare class data Because of a complex structure, it takes more time and more for the computation
16	RACH [130]	More flexible and may produce a more accurate result Having a more complex shape Unable to handle time-changing data
17	Generative and discriminative model [131]	Works for imbalanced class data Only works on stationary data Unable to work on huge time-changing data
18	Biased minimax probability machine [132]	Works only on fixed-size data
19	MTS [133]	A diagnostic and forecasting technique for multivariate data Works only on stationary data
20	Statistical decision theory-based model [134]	Focuses only on two-class problem Unable to solve the multi-class problem
21	CSMLP [135]	Handles noisy data Solves the only two-class problem

Table 3 (continued)

S. no	Methods/algorithms/applications	Advantages and disadvantages
22	Online fault detection algorithm [136]	Designed for sensor data classification Uses an incremental clustering algorithm Reduces memory usage Unable to handle a huge amount of time-changing data
23	CFL [137]	Balances error rate Normalized mutual information Works on stationary data only
24	RACOG and wRACOG [138]	Generate new minority class samples Balances accuracy Unable to handle a huge amount of time-changing data

Table 4 Confusion matrix

Actual class	Predicted class	
	N	*P*
N	TN	FP
P	FN	TP

TN true negative, *TP* true positive, *FN* false negative, *FP* false positive, *P* positive (intrusion), *N* negative (normal)

Table 5 Confusion matrix example

Actual class	Predicted class	
	N	*P*
N	175	07
P	05	13

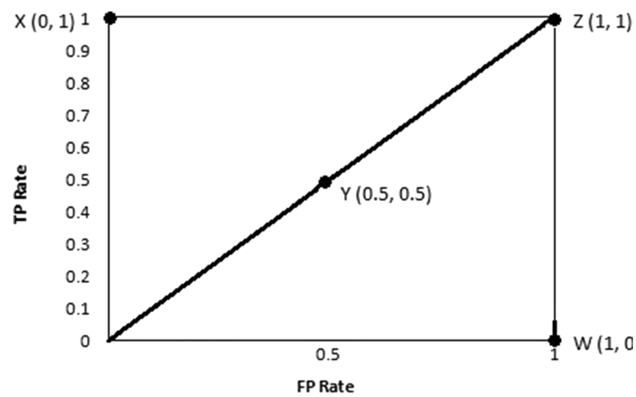

Fig. 1 ROC curve

F-measure and *G*-mean are both stable metrics used to handle imbalanced data.

3.2 Receiver operating characteristics (ROC) curve

ROC curve is created by plotting true-positive rate and false-positive rate.

$$\text{True Positive Rate(TPR)} = \text{TP}/\text{No. of } P \text{ samples} \tag{8}$$

$$\text{False Positive Rate(FPR)} = \text{FP}/\text{No. of } N \text{ samples.} \tag{9}$$

The true-positive rate is also called as sensitivity or recall and the false-positive rate is also called as fall-out and it is also calculated as 1 − Specificity. Any single point in the region of the ROC curve indicates the performance of the single classifier in terms of binary classification as well as multi-class. It provides useful information concerning data distribution such that it is a visual representation of truly classified positive examples versus falsely classified positive examples, i.e., benefit versus cost.

According to the graph, as shown in Fig. 1, point *X* (0, 1) corresponds to the perfect classification of samples. If any point lies towards *X* (0, 1), i.e., nearer to point *X*, it represents that the applied classifier is performing better; and if any point lies away from *X* (0, 1), it represents that one classifier performs badly. The points which lie on diagonal represents that the classifier prediction is 50%, i.e., the given point may predict correctly or may not. If any classifier's performance belongs to the lower right portion of curve i.e. below diagonal, represents that one who classifier predicts wrongly. The one that classifier belongs to the upper left portion of curve, i.e., above diagonal represents a better prediction of the classifier. Limitation of the ROC curve is that in the presence of highly skewed data, the representation of ROC curve becomes the unrealistic performance of the classifier.

3.3 Precision–recall curve

A precision–recall curve is represented by plotting precision rate versus recall rate. A curve dominates in ROC space if it dominates in precision–recall space and an algorithm that optimizes the AUC in the ROC space is not guaranteed to optimize the AUC in the precision–recall space [143].

However, the main objective of the ROC curve is to reside in the upper left corner of the ROC space; while, the precision–recall curve resides in the upper right hand of precision–recall space. The precision–recall curve is an efficient evaluation or visualization technique. Precision–recall curve works more efficiently than ROC curve in the presence of highly skewed data. For example, in a dataset, the negative examples are highly exceeded than positive examples. So in this scenario, there is a chance of large change in the number of false-positive examples or classifier classifies a large number of false-positive examples. It will not significantly change the false-positive rate because many negative examples are large in number significantly. It fails the ROC curve in this condition. On the other hand, consider the precision definition; this is the ratio of TP with TP + FP. Hence, it can correctly represent the given data [143] as well as classifiers performance when there is a large change in false-positive rate [144].

4 Discussion

Concept drifting data classification and imbalanced learning problem have a great attraction in the research field. Many of the researchers tried to handle data stream with concept drift and imbalanced data, but they have found problems like some fundamental issues, unavailability of benchmark datasets, issues about standard evaluation metrics, imbalanced data streams learning, semi-supervised imbalanced learning and deep learning. The above-mentioned issues are discussed as follows,

4.1 Fundamental issues

From the literature review, it has been observed that most of the researchers used a specific algorithm for solving concept drifting and imbalanced learning problem. Most of the algorithm tried to improve accuracy using some metrics. Suppose there are some specific techniques or methodologies that outperform other implemented algorithms and it will be applicable to every domain. But on the basis of deep study, the following issues have been found out:

1. Whether the generated solution will be helpful or can be able to handle various types of data?
2. None of the researchers have a strong decision about whether detection and handling of the concept drifting data work better or not?
3. None of the researchers have a strong opinion about whether the balanced distribution of data in the imbalanced learning field works better or not?
4. Unable to find out the effects of the concept drifting and imbalanced data distribution over computational complexity of learning algorithm.
5. Unable to give a strong opinion on working of the single scan of the huge amount of data.
6. Unable to find out the error rate of imbalanced data distribution.
7. The general learning methodology is unavailable that reduces limitations of learning from concept drifting and imbalanced dataset for specific algorithm and application domains.
8. Unavailability of standard metrics that balance the imbalanced dataset in a general way.

From the above observations, no evaluation metric is fixed for the classification of the concept drifting data stream. Evaluation metrics like cross-validation, hold-out test, interleaved-test-then-train methods are having their pros and cons. They are biased towards specific datasets, and thus are not suitable for all kinds of data. As well as the percentage of training data also varies which is used for learning of the algorithms.

The balancing of imbalanced data has been discussed in [145, 146] where undersampling and the oversampling rate have been covered very well to solve the balancing problem [146] and addressed the resampling method with a combination of different expression. While [145] has advised the fixed size of the training set, the relationship between the class distribution of training data and classifiers performance in terms of accuracy and AUC. Based on observation, if accuracy is selected as performance metric then naturally occurring class distribution must be best class distribution; and if AUC is selected as performance metric, then near to balanced class distribution would be best class distribution.

4.2 Unavailability of benchmark dataset

Availability of data resource is a serious problem in the research field of data engineering and knowledge processing. Currently, UCI machine learning datasets [75] and NIST Scientific and Technical databases [147] are publically available and which apply to a certain type of algorithm or methods. But the issue is that this repository database contains a very limited amount of benchmarking imbalanced datasets and concept drifting datasets. Most of the imbalanced datasets and concept drifting datasets do not have a benchmark to identify and suggest evaluation metrics as well as for suitable algorithm. Because of this, most of the dataset requires manipulation before being applied for the learning algorithm. Because of lack of uniform standard evaluation metrics to measure performance, lack of data sharing and interoperability between different domains requires additional cost for preparation of own dataset to research group.

4.3 Standard evaluation metrics

According to the literature review from Sect. 2 and evaluation metrics from Sect. 3, it has been observed that singular evaluation metric is not sufficient for concept drifting and imbalanced learning field. From the literature survey, most researchers tried for a single evaluation metric which leads to unequal evaluation of algorithm as well as it is difficult to compare the performance of the different algorithm. Also as data change, it becomes very difficult to evaluate the performance of the classification algorithm.

As mentioned in Sects. 2 and 3, every evaluation metrics work independently; so, it becomes very difficult for them to correlate with each other. Because of these issues, a standard set of evaluation metrics is required to evaluate performance as well as comparison.

4.4 Imbalanced learning using huge amount of data

A huge amount of data has been generated over time from applications like networking, weather forecasting, telecommunication, stock market, etc. Such type of huge amount of data is called data streams or streaming data. It is very difficult to scan data streams in a single scan as well as it requires more time and memory to process such data streams. Single classifier fails to handle a huge amount of data. Because of these issues, the ensemble classification method comes forward to solve the classification of data streams. But ensemble learning algorithm requires more time and memory as compared with single classifier with improving performance in terms of accuracy [32, 42, 152]. The data has nature to change over time called as concept drift. Concept drift always occurs in data streams. In data streams, the same type of data never occurs; so data streams is the best example of imbalanced learning.

Imbalanced learning from data streams requires new incremental learning algorithm which will work well, update itself, understand or learn new concepts and tools which can handle data streams and represents useful information. Data stream mining has more requirement in knowledge representation and decision-making process. The following features are required in imbalanced data streams for learning the field.

1. The algorithm must incrementally update its learning strategy as imbalanced data comes.
2. The algorithm must be able to classify new data using its history or past experience.
3. The algorithm must be able to handle concept drift and try to be stable in concept drifting environment.
4. The algorithm also tries to improve learning which will reflect in performance improvement.

Development in these areas enhances learning in real-world applications.

4.5 Semi-supervised learning for concept drifting and imbalanced class data

Supervised learning performs well with labeled data and unsupervised learning performs well with unlabeled data. Semi-supervised learning means dataset contains both labeled and unlabeled data. In the field of data mining, no one expects that generated would be data either labeled or unlabeled always. If the dataset contains both labeled and unlabeled data then the performance of supervised and unsupervised algorithm degrades. Because of this, semi-supervised learning is having a great impact on the data mining field. Many researchers tried semi-supervised learning algorithms [109–112] with different concepts; classifiers are trained with sufficient amount of labeled samples and tested with unlabeled data samples and then identified according to their labels. For this purpose, self-learning methods, semi-supervised SVM, graph-based models, hybrid generative model, a mixture of experts are used in semi-supervised learning. Following are some issues that occur with semi-supervised learning from concept drifting and imbalanced data,

1. It is very difficult to find out or identify a balanced or imbalanced distribution of unlabeled data.
2. It is also difficult to classify concept drifting data of unlabeled data.
3. It has the difficulty to apply the effective and efficient algorithm to discover labels of unlabeled data in imbalanced and concept drifting learning.
4. It is difficult to balance the learning process.

4.6 Deep learning

Deep learning is an important architecture which acts as a set of machine learning models that performs supervised or unsupervised feature learning to automatically form hierarchical representations in deep architectures in big data [148–151]. Because, big data consists of high volume, high variety and high veracity data sets with high-velocity processing requirement. In such a huge volume of data, concept drifting data or imbalanced data occurs naturally because data always change over time. In big data or such a huge volume of data, handling is a big issue. Following are some issues that occur in deep learning from imbalanced data,

1. It is very difficult to find out concept drifting or imbalanced data in big data.

2. It is very difficult to apply heterogeneous architectures to solve the imbalanced problem as well as concept drift problem.
3. Having difficulties with learning of big data.

5 Conclusion

This paper has covered a detailed review of data stream classification in concept drifting and imbalanced class environments. In this paper, we have discussed issues that occur during knowledge discovery in the presence of concept drift or imbalanced learning. This paper has covered classification, clustering, semi-supervised and hybrid methods for classification of concept drifting data and imbalanced class learning with their advantages and disadvantages. Evaluation metrics are also discussed which are used to evaluate the performance of the algorithm. In the end, issues like fundamental concepts, benchmark datasets, standard evaluation metrics, learning with a huge amount of data, deep learning, semi-supervised learning are discussed which will be useful for the researchers to handle data streams and its issues. Also, it focuses on opportunities and research area that are available in the field of the data stream and knowledge discovery.

References

1. Aggarwal, C.: Data Streams: Models and Algorithms. Springer, New York (2007)
2. Bifet, A.: Adaptive Stream Mining: Pattern Learning and Mining from Evolving Data Streams. IOS Press, Amsterdam (2010)
3. Bifet, A., Kirkby, R.: Data Stream Mining a Practical Approach. University of Waikato, Hamilton (2009)
4. Fan, W., Huang, Y., Wang, H., and Yu, P.S.: Active mining of data streams. In: Proceedings of the SIAM International Conference on Data Mining (SDM '04), (2004)
5. Gaber, M.M., Zaslavsky, A., Krishnaswamy, S.: Mining data streams: a review. ACM SIGMOD Rec. **34**(2), 1826 (2005)
6. Gama, J., Fernandes, R., Rocha, R.: Decision trees for mining data streams. Intell. Data Anal. **10**(1), 23–45 (2006)
7. Gao, J., Fan, W., and Hang, J.: On appropriate assumptions to mine data streams: analysis and practice. In: Proceedings of the IEEE International Conference on Data Mining (ICDM '07), Oct. (2007)
8. Han, J., Kamber, M.: Data Mining: Concepts and Techniques, 2nd edn. Elsevier, Amsterdam (2006)
9. Pfahringer, B., Holmes, G., and Kirkby, R.: New options for Hoeffding trees. In: Proceedings of the 20th Australasian Joint Conference on Artificial Intelligence (AI '07), pp. 90–99. (2007)
10. Gama, J.: A survey on learning from data streams: current and future trends. Prog. Artif. Intell. **1**(1), 45–55 (2012)
11. Grossi, V., Turini, F.: Stream mining: a novel architecture for ensemble-based classification. Knowl. Inf. Syst. **30**(2), 247–281 (2012)
12. Wankhade, K., Dongre, S., Thool, R.: New evolving ensemble classifier for handling concept drifting data streams. In: The Proceedings of 2012 2nd IEEE International Conference on Parallel, Distributed and Grid Computing (PDGC-12), INDIA, pp. 657–662. (2012)
13. Bose, R., van der Aalst, W., Zliobaite, I., Pechenizkiy, M.: Dealing with concept drifts in process mining. IEEE Trans. Neural Netw. Learn. Syst. **25**(1), 154–171 (2014)
14. Kuncheva, L., Faithfull, W.: PCA feature extraction for change detection in multidimensional unlabeled data. IEEE Trans. Neural Netw. Learn. Syst. **25**(1), 69–80 (2014)
15. Pratama, M., Anavatti, S., Angelov, P., Lughofer, E.: PANFIS: a novel incremental learning machine. IEEE Trans. Neural Netw. Learn. Syst. **25**(1), 55–68 (2014)
16. Lughofer, E., Angelov, P.: Handling drifts and shifts in on-line data streams with evolving fuzzy systems. Appl. Soft Comput. **11**(2), 2057–2068 (2011)
17. Kasabov, N.: Evolving fuzzy neural networks for supervised/unsupervised online knowledge-based learning. IEEE Trans. Syst. Man Cybern. B Cybern. **31**(6), 902–918 (2001)
18. Angelov, P., Lughofer, E., Zhou, X.: Evolving fuzzy classifiers using different model architectures. Fuzzy Sets Syst. **159**(23), 3160–3182 (2008)
19. Faisal, M.A., Aung, Z., Williams, J.R., Sanchez, A.: Data-stream-based intrusion detection system for advanced metering infrastructure in smart grid: a feasibility study. IEEE Syst. J. **9**(1), 31–44 (2015)
20. Domingos P., Hulten G.: Mining high-speed data streams. In: International Conference on Knowledge Discovery and Data Mining, 2000, pp. 71–80. (2000)
21. Widmer, G., Kubat, M.: Learning in the presence of concept drift and hidden contexts. Mach. Learn. **23**(1), 69–101 (1996)
22. Bifet, A., and Gavalda, R.: Learning from time-changing data with adaptive windowing. In: Proceedings of the 2007 SIAM International Conference on Data Mining, 2007, pp. 443–448. (2007)
23. Hulten, G., Spencer, L., and Domingos, P.: Mining time-changing data streams. In: Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data, 2001, pp. 97–106. (2001)
24. Fan, W., Huang, Y., and Yu, P.: Decision tree evolution using limited number of labeled data items from drifting data streams. In: Proceedings of the Fourth IEEE International Conference on Data Mining, 2004, pp. 379–382. (2004)
25. Liu, J., Li, X., Hong, W.: Ambiguous decision trees for mining concept-drifting data streams. Pattern Recogn. Lett. **30**(15), 1347–1355 (2009)
26. Vivekanandan, P., Nedunchezhian, R.: Mining rules of concept drift using genetic algorithm. J. Artif. Intell. Soft Comput. Res. **1**(2), 135–145 (2011)
27. Wang H., Fan W., Yu V., and Han J.: Mining concept-drifting data streams using ensemble classifiers. In: ACM SIGKDD, Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 226–235. (2003)
28. Masud M., Gao J., Khan L., Han J., and Thuraisingham, B.: A multi-partition multi-chunk ensemble technique to classify concept drifting data streams. In: Proceedings of the 13th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD'09), pp. 363–375. Springer, Berlin (2009)
29. Kolter, J., Maloof, M.: Dynamic weighted majority: a new ensemble method for tracking concept drift. J. Mach. Learn. Res. **8**, 2755–2790 (2007)
30. Oza N., Russell S.: Experimental comparisons of online and batch versions of bagging and boosting. In: Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'01), pp. 359–364 (2001)

31. Pelossof R., Jones M., Vovsha I., Rudin C.: Online coordinate boosting, pp. 1–9. https://arxiv.org/abs/0810.4553 (2008)
32. Bieft A., Holmes G., Pfahringr B., Kirkby R., Gavalda R.: New ensemble methods for evolving data streams. In: Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'09), Paris, France, pp. 139–148 (2009)
33. Y. Law, C. Zaniolo, "An Adaptive Nearest Neighbor Classification Algorithm for Data Streams", In Proceedings of 9th European Conference on Principals and Practice of Knowledge Discovery in Databases (PKDD-2005), Porto, Portugal, Springer-Verlag LNAI 3721, pp 108–120, 2005.
34. Agrawal, C., Han, J., Wang, J., Yu, P.: A framework for on-demand classification of evolving data streams. IEEE Trans. Knowl. Data Eng. **18**(5), 577–589 (2006)
35. Troyano, F., Ruiz, J., Riquelme, J.: Data streams classification by incremental rule learning with parameterized generalization. In: Proceedings of the 2006 ACM Symposium on Applied computing (SAC'06), France, ACM, pp 657–661 (2006)
36. Hashemi, S., Yang, Y., Mirzamomen, Z., Kangavari, M.: Adapted one-versus-all decision trees for data stream classification. IEEE Trans. Knowl. Data Eng. **21**(5), 624–637 (2009)
37. Liang, C., Zhang, Y., Song, Q.: Decision tree for dynamic and uncertain data streams. In: Proceedings of 2nd Asian Conference on Machine Learning (ACML2010), Tokyo, Japan, pp. 209–224 (2010)
38. Li, P., Wu, X., Hu, X.: Mining recurring concept drifts with limited labeled streaming data. In: Proceedings of 2nd Asian Conference on Machine Learning (ACML 2010), Tokyo, Japan, pp. 241–252 (2010)
39. Li, H., Lee, S.: Mining frequent itemsets over data streams using efficient window sliding techniques. J. Expert Syst. Appl. **36**, 1466–1477 (2009)
40. Zliobaite, I.: Ensemble learning for concept drift handling—the role of new expert. In: Proceedings of the 5th International Conference on Machine Learning and Data Mining in Pattern Recognition (MLDM 2007), Leipzig, Germany, pp. 251–260 (2007)
41. Abdulsalam, H., Skillicorn, D., Martin, P.: Classification using streaming random forests. IEEE Trans. Knowl. Data Eng. **23**(1), 22–36 (2011)
42. Attar, V., Sinha, P., Wankhade, K.: A fast and light classifier for data streams. Evol. Syst. **1**(3), 199–207 (2010)
43. Lughofer, E.: Dynamic evolving cluster models using on-line split-and-merge operations. In: Proceedings of 10th International Conference on Machine Learning and Applications, IEEE, 2011, pp. 20–26 (2011)
44. Cao, F., Ester, M., Qian, W., and Zhou, A.: Density-based clustering over an evolving data stream with noise. In: Proceedings of the SIAM Conference on Data Mining, 2006, pp. 326–337 (2006)
45. Liu, L., Huang, H., Guo, Y., Chen, F.: rDenStream, a clustering algorithm over an evolving data stream. In: 2009 International Conference on Information Engineering and Computer Science, IEEE, 2009. (2009)
46. Qian, L., and Qin, L.: A framework of cluster decision tree in data stream classification. In: 2012 4th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC), IEEE, vol. 1, 2012, pp. 38–41 (2012)
47. Sun N., and Guo, Y.: A modified incremental learning approach for data stream classification. In: Sixth International Conference on Internet Computing for Science and Engineering (ICICSE), 2012, pp. 122–125 (2012)
48. Hosseini, M.J., Ahmadi, Z., and Beigy, H.: Pool and accuracy based stream classification: a new ensemble algorithm on data stream classification using recurring concepts detection. In: IEEE 11th International Conference on Data Mining Workshops (ICDMW), 2011, pp. 588–595 (2011)
49. Masud, M.M., Gao, J., Khan, L., Bhavani, B.: Classification and novel class detection in concept-drifting data streams under time constraints. IEEE Trans. Knowl. Data Eng. **23**(6), 859–874 (2011)
50. Guha, S., Mishra, N., Motwani, R., O'Callaghan, L.: Clustering data streams, pp. 359–366. Annual IEEE Symposium, Foundations of Computer Science (2000)
51. O'callaghan, L., Mishra, N., Meyerson, A., Guha, S., and Motwani, R.: Streaming-data algorithms for high-quality clustering. In: Proceedings of the 18th International Conference on Data Engineering (ICDE.02), IEEE, 2002. pp. 685–694 (2002)
52. Aggarwal, C., Han, J., Wang, J., and Yu, P.: A framework for clustering evolving data streams. In: Proceedings of the 29th International Conference on Very Large Data Bases, 2003, Vol. 29, pp. 81–92 (2003)
53. Jia, C., Tan, C. Y., Yong, A.: A grid and density-based clustering algorithm for processing data stream. In: Proceedings of Second International Conference on Genetic and Evolutionary Computing, IEEE, 2008, pp. 517–521 (2008)
54. Fong, S., Wong, R., Vasilakos, A.V.: Accelerated PSO swarm search feature selection for data stream mining big data. IEEE Trans. Serv. Comput. **9**(1), 33–45 (2016)
55. Rutkowski, L., Jaworski, M., Pietruczuk, L., Duda, P.: A new method for data stream mining based on the misclassification error. IEEE Trans. Neural Netw. Learn. Syst. **26**(5), 1048–1059 (2015)
56. Rutkowski, L., Jaworski, M., Pietruczuk, L., Duda, P.: Decision trees for mining data streams based on the Gaussian approximation. IEEE Trans. Knowl. Data Eng. **26**(1), 108–119 (2014)
57. Sun, Y., Tang, K., Minku, L.L., Wang, S., Yao, X.: Online ensemble learning of data streams with gradually evolved classes. IEEE Trans. Knowl. Data Eng. **28**(6), 1532–1545 (2016)
58. Brzezinski, D., Stefanowski, J.: Reacting to different types of concept drift: the accuracy updated ensemble algorithm. IEEE Trans. Neural Netw. Learn. Syst. **25**(1), 81–94 (2014)
59. Gomes, J.B., Gaber, M.M., Sousa, P.A.C., Menasalvas, E.: Mining recurring concepts in a dynamic feature space. IEEE Trans. Neural Netw. Learn. Syst. **25**(1), 95–110 (2014)
60. Zhang, L., Lin, J., Karim, R.: Sliding window-based fault detection from high-dimensional data streams. IEEE Trans. Syst. Man Cybern. Syst. **47**(2), 289–303 (2017)
61. Salehi, M., Leckie, C., Bezdek, J.C., Vaithianathan, T., Zhang, X.: Fast memory efficient local outlier detection in data streams. IEEE Trans. Knowl. Data Eng. **28**(12), 3246–3260 (2016)
62. Zhang, P., Zhou, C., Wang, P., Gao, B.J., Zhu, X., Guo, L.: E-Tree: an efficient indexing structure for ensemble models on data streams. IEEE Trans. Knowl. Data Eng. **27**(2), 461–474 (2015)
63. Qahtan, A., Wang, S., Zhang, X.: KDE-track: an efficient dynamic density estimator for data streams. IEEE Trans. Knowl. Data Eng. **29**(3), 642–655 (2017)
64. Al-Khateeb, T., Masud, M.M., Al-Naami, K.M., Seker, S.E., Mustafa, A.M., Khan, L., Trabelsi, Z., Aggarwal, C., Han, J.: Recurring and novel class detection using class-based ensemble for evolving data stream. IEEE Trans. Knowl. Data Eng. **28**(10), 2752–2764 (2016)
65. Li, X., Yu, W., Villegas, S.: Structural health monitoring of building structures with online data mining methods. IEEE Syst. J. **10**(3), 1291–1300 (2016)
66. Chen, X., Vorvoreanu, M., Madhavan, K.: Mining social media data for understanding student's learning experiences. IEEE Trans. Learn. Technol. **7**(3), 246–259 (2014)

67. Zhang, Q., Zhang, P., Long, G., Ding, W., Zhang, C., Wu, X.: Online learning from trapezoidal data streams. IEEE Trans. Knowl. Data Eng. **28**(10), 2709–2723 (2016)
68. Liu, S., Qu, Q., Chen, L., Ni, L.M.: SMC: a practical schema for privacy-preserved data sharing over distributed data streams. IEEE Trans. Big Data **1**(2), 68–81 (2015)
69. Canzian, L., Van Der Schaar, M.: Real-time stream mining: online knowledge extraction using classifier networks. IEEE Netw. **29**(5), 10–16 (2015)
70. Tekin, C., van der Schaar, M.: Active learning in context-driven stream mining with an application to image mining. IEEE Trans. Image Process. **24**(11), 3666–3679 (2015)
71. de Faria, E.R., Gonçalves, I.R., Gama, J., de Leon Ferreira, A.C.P.C.: Evaluation of multiclass novelty detection algorithms for data streams. IEEE Trans. Knowl. Data Eng. **27**(11), 2961–2973 (2015)
72. Hahsler, M., Bolaños, M.: Clustering data streams based on shared density between micro-clusters. IEEE Trans. Knowl. Data Eng. **28**(6), 1449–1461 (2016)
73. Liu, B., Xiao, Y., Yu, P.S., Cao, L., Zhang, Y., Hao, Z.: Uncertain one-class learning and concept summarization learning on uncertain data streams. IEEE Trans. Knowl. Data Eng. **26**(2), 468–484 (2014)
74. Dyer, K.B., Capo, R., Polikar, R.: COMPOSE: a semi-supervised learning framework for initially labeled nonstationary streaming data. IEEE Trans. Neural Netw. Learn. Syst. **25**(1), 12–26 (2014)
75. UCI machine learning repository. https://archive.ics.uci.edu/ml/datasets.html
76. He, H., Garcia, E.A.: Learning from imbalanced data. IEEE Trans. Knowl. Data Eng. **21**(9), 1263–1284 (2009)
77. Galar, M., Fernandez, A., Barrenechea, E., Bustince, H., Herrera, F.: A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches. IEEE Trans. Syst. Man Cybern. Part C Appl. Rev. **42**(4), 463–484 (2011)
78. Wang, S., Yao, X.: Multiclass imbalance problems: analysis and potential solutions. IEEE Trans. Syst. Man Cybern. Part B Cybern. **42**(4), 1119–1130 (2012)
79. Azaria, A., Richardson, A., Kraus, S., Subrahmanian, V.S.: Behavioral analysis of insider threat: a survey and bootstrapped prediction in imbalanced data. IEEE Trans. Comput. Soc. Syst. **1**(2), 135–155 (2014)
80. Cano, A., Zafra, A., Ventura, S.: Weighted data gravitation classification for standard and imbalanced data. IEEE Trans. Cybern. **43**(6), 1672–1687 (2013)
81. Aksoylar, C., Qian, J., Saligrama, V.: Clustering and community detection with imbalanced clusters. IEEE Trans. Signal Inf. Process. Over Netw. **3**(1), 61–76 (2017)
82. Bae, S., Yoon, K.: Polyp detection via imbalanced learning and discriminative feature learning. IEEE Trans. Med. Imaging **34**(11), 2379–2393 (2015)
83. Pérez-Ortiz, M., Gutiérrez, P.A., Hervás-Martínez, C., Yao, X.: Graph-based approaches for over-sampling in the context of ordinal regression. IEEE Trans. Knowl. Data Eng. **27**(5), 1233–1245 (2015)
84. Wang, F., Xu, T., Tang, T., Zhou, M., Wang, H.: Bilevel feature extraction-based text mining for fault diagnosis of railway systems. IEEE Trans. Intell. Transp. Syst. **18**(1), 49–58 (2017)
85. Lee, T., Lee, K.B., Kim, C.O.: Performance of machine learning algorithms for class-imbalanced process fault detection problems. IEEE Trans. Semicond. Manuf. **29**(4), 436–445 (2016)
86. Khreich, W., Granger, E., Miri, A., Sabourin, R.: Iterative Boolean combination of classifiers in the roc space: an application to anomaly detection with HMMs. J Pattern Recogn. **43**(8), 2732–2752 (2010)
87. Tavallaee, M., Stakhanova, N., Ghorbani, A.: Toward credible evaluation of anomaly-based intrusion–detection methods. IEEE Trans. Syst. Man Cybern. Part C Appl. Rev. **40**(5), 516–524 (2010)
88. Huda, S., Yearwood, J., Jelinek, H.F., Hassan, M.M., Fortino, G., Buckland, M.: A hybrid feature selection with ensemble classification for imbalanced healthcare data: a case study for brain tumor diagnosis. IEEE Access **4**, 9145–9154 (2016)
89. Dai, H.: Imbalanced protein data classification using ensemble FTM-SVM. IEEE Trans. NanoBiosci **14**(4), 350–359 (2015)
90. Liu, N., Koh, Z.X., Chua, E.C., Tan, L.M., Lin, Z., Mirza, B., Ong, M.E.H.: Risk scoring for prediction of acute cardiac complications from imbalanced clinical data. IEEE J. Biomed. Health Inf. **18**(6), 1894–1902 (2014)
91. Yu, H., Ni, J.: An improved ensemble learning method for classifying high-dimensional and imbalanced biomedicine data. IEEE/ACM Trans. Comput. Biol. Bioinform. **11**(4), 657–666 (2014)
92. Chen, P., Hu, S., Zhang, J., Gao, X., Li, J., Xia, J., Wang, B.: A sequence-based dynamic ensemble learning system for protein ligand-binding site prediction. IEEE/ACM Trans. Comput. Biol. Bioinform. **13**(5), 901–912 (2016)
93. Cuendet, G.L., Schoettker, P., Yüce, A., Sorci, M., Gao, H., Perruchoud, C., Thiran, J.: Facial image analysis for fully automatic prediction of difficult endotracheal intubation. IEEE Trans. Biomed. Eng. **63**(2), 328–339 (2016)
94. Phua, C., Alahakoon, D., Lee, V.: Minority report in fraud detection: classification of skewed data. ACM SIGKDD Explor. Newsl. **6**(1), 50–59 (2004)
95. del Castillo, M.D., Serrano, J.I.: A multi strategy approach for digital text categorization from imbalanced documents. ACM SIGKDD Explor. Newsl. **6**(1), 70–79 (2004)
96. Ling, C. X., Li, C.: Data mining for direct marketing: problems and solutions. In: The Proceedings of 4th International Conference on Knowledge Discovery and Data Mining (KDD), 1998, pp. 73–79 (1998)
97. Bermejo, P., Gámez, J.A., Puerta, J.M.: Improving the performance of naive Bayes multinomial in e-mail foldering by introducing distribution based balance of datasets. J. Expert Syst. Appl. **38**(3), 2072–2080 (2011)
98. Liu, Y.H., Chen, Y.T.: Total margin-based adaptive fuzzy support vector machines for multiview face recognition. Proc. IEEE Int. Conf. Syst. Man Cybern. **2**, 1704–1711 (2005)
99. Kim, S., Kim, H., Namkoong, Y.: Ordinal classification of imbalanced data with application in emergency and disaster information services. IEEE Intell. Syst. **31**(5), 50–56 (2016)
100. Sanz, J.A., Bernardo, D., Herrera, F., Bustince, H., Hagras, H.: A compact evolutionary interval-valued fuzzy rule-based classification system for the modeling and prediction of real-world financial applications with imbalanced data. IEEE Trans. Fuzzy Syst. **23**(4), 973–990 (2015)
101. Cao, H., Tan, V.Y.F., Pang, J.Z.F.: A parsimonious mixture of Gaussian trees model for oversampling in imbalanced and multimodal time-series classification. IEEE Trans. Neural Netw. Learn. Syst. **25**(12), 2226–2239 (2014)
102. Pérez-Ortiz, M., Gutiérrez, P.A., Tino, P., Hervás-Martínez, C.: Oversampling the minority class in the feature space. IEEE Trans. Neural Netw. Learn. Syst. **27**(9), 1947–1961 (2016)
103. Amin, A., Anwar, S., Adnan, A., Nawaz, M., Howard, N., Qadir, J., Hawalah, A., Hussain, A.: Comparing oversampling techniques to handle the class imbalance problem: a customer churn prediction case study. IEEE Access **4**, 7940–7957 (2016)
104. Abdi, L., Hashemi, S.: To Combat multi-class imbalanced problems by means of over-sampling techniques. IEEE Trans. Knowl. Data Eng. **28**(1), 238–251 (2016)

105. Chawla, N.V., Hall, L.O., Bowyer, K.W.: SMOTE: synthetic minority oversampling technique. J. Artif. Intell. Res. **16**, 321–357 (2002)
106. Wankhade, K., Dongre, S.: A new adaptive ensemble boosting classifier for concept drifting stream data. Int. J. Model. Optim. (IJMO) **2**(4), 488–492 (August 2012)
107. Wang, B., Pineau, J.: Online bagging and boosting for imbalanced data streams. IEEE Trans. Knowl. Data Eng. **28**(12), 3353–3366 (2016)
108. Breiman, L.: Bagging predictors. J Mach. Learn. **24**(2), 123–140 (1996)
109. Zhu, X.: Semi-supervised learning literature survey. Technical Report TR-1530, University of Wisconsin-Madison, 2007
110. Rosenberg, C., Hebert, M., Schneiderman, H.: Semi-supervised self-training of object detection models. In: The Proceeding of IEEE Workshops Application of Computer Vision, 2005, pp. 29–36 (2005)
111. Sindhwani, V., Keerthi, S. S.: Large scale semi-supervised linear SVMs. In: The Proceeding of International SIGIR Conference on Research and Development in Information Retrieval, 2006, pp. 477–484 (2006)
112. Fujino, A., Ueda, N., Saito, K.: A hybrid generative/discriminative approach to semi-supervised classifier design. In: The Proceeding of National Conference on Artificial Intelligence, 2005, pp. 764–769 (2005)
113. Lin, S., Wang, C., Wu, Z., Chung, Y.: Detect rare events via MICE algorithm with optimal threshold. In: The Proceeding of Seventh International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing, IEEE, 2013, pp. 70–75 (2013)
114. Seiffert, C., Khoshgoftaar, T.M., Hulse, J.V., Napolitano, A.: RUSBoost: a hybrid approach to alleviating class imbalance. IEEE Trans. Syst. Man Cybern. Part A Syst. Hum. **40**(1), 185–197 (2010)
115. Oh, S., Lee, M.S., Zhang, B.: Ensemble learning with active example selection for imbalanced biomedical data classification. IEEE/ACM Trans. Comput. Biol. Bioinform. **8**(2), 316–325 (2011)
116. Yang, P., Yoo, P.D., Fernando, J., Zhou, B.B., Zhang, Z., Zomaya, A.Y.: Sample subset optimization techniques for imbalanced and ensemble learning problems in bioinformatics applications. IEEE Trans. Cybern. **44**(3), 445–455 (2014)
117. Ditzler, G., Polikar, R.: Incremental learning of concept drift from streaming imbalanced data. IEEE Trans. Knowl. Data Eng. **25**(10), 2283–2301 (2013)
118. Sun, Y., Kamel, M. S., Wang, Y.: Boosting for learning multiple classes with imbalanced class distribution. In: The Proceeding of Sixth International Conference on Data Mining (ICDM), 2006, pp. 592–602 (2006)
119. Huang, K., Kuo, Y., Yeh, I.: A novel fitness function in genetic algorithms to optimize neural networks for imbalanced data sets. In: The Proceeding of the Eighth International Conference on Intelligent Systems Design and Application, IEEE, 2008, pp. 647–650 (2008)
120. Wang, S., Minku, L.L., Yao, X.: Resampling-based ensemble methods for online class imbalance learning. IEEE Trans. Knowl. Data Eng. **27**(5), 1356–1368 (2015)
121. Ahumada, H., Grinblat, G. L., Uzal, L. C., Granitto, P. M., Ceccatto, A.: REPMAC: A new hybrid approach to highly imbalanced classification problems. In: The Proceeding of Eighth International Conference on Hybrid Intelligent Systems, IEEE, 2008, pp. 386–391 (2008)
122. Jeatrakul, P., Wong, K.W.: Enhancing Classification Performance of Multi-Class Imbalanced Data Using the OAA-DB Algorithm, In the proceeding of IEEE World Congress on Computational Intelligence (WCCI), pp. 1–8. Brisbane, IEEE (2012)
123. Tan, S. C., Watada, J., Ibrahim, Z., Khalid, M., Jau, L. W., Chew, L. C.: Learning with imbalanced datasets using fuzzy ARTMAP-based neural network models. In: The Proceeding of 2011 IEEE International Conference on Fuzzy Systems, 2011, Taiwan, pp. 1084–1089 (2011)
124. Cao, P., Li, B., Zhao, D., Zaiane, O.: A novel cost sensitive neural network ensemble for multiclass imbalance data learning. In: The Proceeding of International Joint Conference on Neural Networks (IJCNN), IEEE, 2013, pp. 1–8 (2013)
125. Fu, J., Lee, S.: Certainty-enhanced active learning for improving imbalanced data classification. In: The Proceeding of 11th IEEE International Conference on Data Mining Workshops, IEEE, 2011, pp. 405–412 (2011)
126. Antwi, D. K., Viktor, H. L., Japkowicz, N.: The PerfSim algorithm for concept drift detection in imbalanced data. In: The Proceeding of 12th IEEE International Conference on Data Mining Workshops, IEEE, 2012, pp. 619–628 (2012)
127. Alhammady, H., Ramamohanarao, K.: Using emerging patterns and decision trees in rare-class classification. In: Proceedings of the Fourth IEEE International Conference on Data Mining (ICDM'04), 2004, pp. 315–318 (2004)
128. Wang, P., Wang, H., Wu, X., Wang, W., Shi, B.: A low-granularity classifier for data streams with concept drifts and biased class distribution. IEEE Trans. Knowl. Data Eng. **19**(9), 1202–1213 (2007)
129. Orriols-Puig, A., Bernadó-Mansilla, E., Goldberg, D.E., Sastry, K., Lanzi, P.L.: Facetwise analysis of XCS for problems with class imbalances. IEEE Trans. Evol. Comput. **13**(5), 1093–1119 (2009)
130. He, J., Tong, H., Carbonell, J.: Rare category characterization. In: The Proceeding of IEEE International Conference on Data Mining, 2010, pp. 226–235 (2010)
131. Hospedales, T.M., Gong, S., Xiang, T.: Finding rare classes: active learning with generative and discriminative models. IEEE Trans. Knowl. Data Eng. **25**(2), 374–386 (2013)
132. Huang, K., Yang, H., King, I., Lyu, M.R.: Imbalanced learning with a biased minimax probability machine. IEEE Trans. Syst. Man Cybern. Cybern. **36**(4), 913–923 (2006)
133. Su, C., Hsiao, Y.: An evaluation of the robustness of MTS for imbalanced data. IEEE Trans. Knowl. Data Eng. **19**(10), 1321–1332 (2007)
134. Diamantini, C., Potena, D.: Bayes vector quantizer for class-imbalance problem. IEEE Trans. Knowl. Data Eng. **21**(5), 638–651 (2009)
135. Castro, C.L., Braga, A.P.: Novel cost-sensitive approach to improve the multilayer perceptron performance on imbalanced data. IEEE Trans. Neural Netw. Learn. Syst. **24**(6), 888–899 (2013)
136. Kwak, J., Lee, T., Kim, C.O.: An incremental clustering-based fault detection algorithm for class-imbalanced process data. IEEE Trans. Semicond. Manuf. **28**(3), 1–11 (2015)
137. Zhang, X., Hu, B.: A new strategy of cost-free learning in the class imbalance problem. IEEE Trans. Knowl. Data Eng. **26**(12), 2872–2885 (2014)
138. Das, B., Krishnan, N.C., Cook, D.J.: RACOG and wRACOG: two probabilistic oversampling techniques. IEEE Trans. Knowl. Data Eng. **27**(1), 222–234 (2015)
139. Weiss, G.M.: Mining with rarity: a unifying framework. ACM SIGKDD Explor. Newsl. **6**(1), 7–19 (2004)
140. Sun, Y., Kamel, M.S., Wong, A.K.C., Wang, Y.: Cost-sensitive boosting for classification of imbalanced data. Pattern Recogn. **40**(12), 3358–3378 (2007)

141. Provost, F. J., Fawcett, T.: Analysis and visualization of classifier performance: comparison under imprecise class and cost distributions. In: The Proceeding of International Conference on Knowledge Discovery and Data Mining, 1997, pp. 43–48 (1997)
142. Provost, F. J., Fawcett, T., Kohavi, R.: The case against accuracy estimation for comparing induction algorithms. In: The Proceeding of International Conference on Machine Learning, 1998, pp. 445–453 (1998)
143. Davis, J., Goadrich, M.: The relationship between precision-recall and ROC curves. In: The Proceeding of International Conference on Machine Learning, 2006, pp. 233–240 (2006)
144. Bunescu, R., Ge, R., Kate, R., Marcotte, E., Mooney, R., Ramani, A., Wong, Y.: Comparative experiments on learning information extractors for proteins and their interactions. Artif. Intell. Med. **33**, 139–155 (2005)
145. Weiss, G.M., Provost, F.: Learning when training data are costly: the effect of class distribution on tree induction. J. Artif. Intell. Res. **19**, 315–354 (2003)
146. Estabrooks, A., Jo, T., Japkowicz, N.: A multiple resampling method for learning from imbalanced data sets. Int. J. Comput. Intell. **20**(1), 18–36 (2004)
147. NIST Scientific and Technical Databases. https://nist.gov/srd/online.htm (2009)
148. Park, S., Ha, Y.: Large imbalance data classification based on MapReduce for traffic accident prediction. In: The Proceeding of Eighth International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing, 2014, pp. 45–49 (2014)
149. Ngiam, J., Khosla, A., Kim, M., Nam, J., Lee, H., and Ng, A. Y.: Multimodal deep learning. In: The Proceeding of 28th International Conference Machine Learning, 2011, pp. 689–696 (2011)
150. Srivastava, N., and Salakhutdinov, R.: Multimodal learning with deep Boltzmann machines. In: Proceeding of the Advance Neural Information Processing Systems, pp. 2222–2230 (2012)
151. Zhang, Q., Yang, L.T., Chen, Z.: Deep computation model for unsupervised feature learning on big data. IEEE Trans. Serv. Comput. **9**(1), 161 (2016)
152. Wankhade, K., Jondhale, K., Thool, V.: A hybrid approach for classification of rare class data. Knowl. Inf. Syst. **56**(1), 197–221 (2018)