JOURNAL OF INFORMATION AND COMMUNICATION TECHNOLOGY
http://e-journal.uum.edu.my/index.php/jict

How to cite this article:
Taha, A. Y., Tiun, S., Abd Rahman, A. H., & Sabah, A. (2021). Multilabel over-sampling and under-sampling with class alignment for imbalanced multilabel text classification. *Journal of Information and Communication Technology, 20*(3), 423-456. https://doi.org/10.32890/jict2021.20.3.6

Multilabel Over-sampling and Under-sampling with Class Alignment for Imbalanced Multilabel Text Classification

[1]Adil Yaseen Taha, [2]Sabrina Tiun, [3]Abdul Hadi Abd Rahman & [4]Ali Sabah
[1,2,3&4] Faculty of Information Science and Technology
Universiti Kebangsaan Malaysia, Malaysia

adil.yaseen89, alisabahphd @gmail.com
sabrinatiun, abdulhadi @ukm.edu.my

Received: 9/11/2019 Revised: 11/1/2021 Accepted: 24/1/2021 Published: 11/6/2021

ABSTRACT

Simultaneous multiple labeling of documents, also known as multilabel text classification, will not perform optimally if the class is highly imbalanced. Class imbalance entails skewness in the fundamental data for distribution that leads to more difficulty in classification. Random over-sampling and under-sampling are common approaches to solve the class imbalance problem. However, these approaches have several drawbacks; under-sampling is likely to dispose of useful data, whereas over-sampling can heighten the probability of overfitting. Therefore, a new method that can avoid discarding useful data and overfitting problems is needed. This study proposed a method to tackle the class imbalance problem by combining multilabel over-sampling and

under-sampling with class alignment (ML-OUSCA). In the proposed ML-OUSCA, instead of using all the training instances, it drew a new training set by over-sampling small size classes and under-sampling big size classes. To evaluate the proposed ML-OUSCA, evaluation metrics of average precision, average recall, and average F-measure on three benchmark datasets, namely Reuters-21578, Bibtex, and Enron datasets, were performed. Experimental results showed that the proposed ML-OUSCA outperformed the chosen baseline random resampling approaches: K-means SMOTE and KNN-US. Therefore, based on the results, it can be concluded that designing a resampling method based on class imbalance together with class alignment will improve multilabel classification even better than just the random resampling method.

Keywords: Data mining, multilabel text classification, class imbalance problem, resampling method, class alignment.

INTRODUCTION

Multilabel classification is a task applied in various data mining applications, such labeling video, images, music, and texts. Multilabel classification assigns documents to several classes at the same time based on their belongings. This task differs from the traditional single label, which associates each document to one class. The classification task of the single label can also be considered as multiclass or binary classification. In multiclass classification, each document can belong to more than one label category, but only one label category is assigned. Whereas, in the multilabel classification, it is a generalization of the multiclass and binary classification, as it does not enforce any limits to the number of components that are held at the outputs (Charte et al., 2013; Siblini et al., 2019). Methods of multilabel text classification suffer from a high level of class imbalance, and because of that, they will not work efficiently (Ali et al., 2019; Glazkova, 2020; Japkowicz & Stephen, 2002; Koziarski et al., 2020). The main issue in class imbalance occurs when a certain class has an extremely higher number of instances than other classes (Tanha et al., 2020; Weng et al., 2018; Zhang et al., 2020). In actual conditions, skewness often occurs in the distribution of examples of certain classes that rarely appear. Such an issue affects learning

algorithms leaning toward the majority classes. Numerous solutions for imbalanced classification have been proposed by García et al. (2018), Pereira et al. (2020), Patel et al. (2020), Qiao et al. (2017), and Song et al. (2016). However, previous works principally focused on binary classifications that were less complex than multilabel imbalanced classification (Cascar et al., 2019; Sáez et al., 2016). Results of earlier studies have shown that random over-sampling and under-sampling are the most efficient approaches to solve imbalanced classifications (García et al., 2018; Sáez et al., 2016; Zhang et al., 2020). However, the approaches have several drawbacks; under-sampling is likely to dispose of useful data, whereas over-sampling can heighten the probability of overfitting (Charte et al., 2015; Qiao et al., 2017; Sáez et al., 2016). Therefore, a new method that can avoid discarding useful data and overfitting problems is needed.

This study presents a proposed method on handling the class imbalance problem for a multilabel learning model on text classification based on a new sampling and class alignment. The proposed method combines multilabel over-sampling and under-sampling, and class alignment, which is called the ML-OUSCA algorithm. This combination aims to deal with the limitations of previous approaches in tackling the class imbalance problem. Likewise, this study aims to balance the classes in the training set examples by joining and exploiting the power of over-sampling, under-sampling, and non-sampling methods. However, the proposed method draws a new training set using under-sampling by only discarding a few non-useful majority class samples from the set. The discarding strategy in the under-sampling method is based on the interdependency between the training set samples. By contrast, over-sampling is performed by duplicating few randomly selected minority class samples from the dataset. In addition, the samples in the classes that are not too high or low in their numbers are identified as class alignment (balanced classes) and kept without resorting to over-sampling and under-sampling.

This work is henceforth structured into five sections. The next section presents a short review of previous related studies. The proposed method and its assessment are presented in the following section. The fourth section describes the carried-out experiments. Finally, the proposed method is concluded in the last section.

RELATED WORKS

Multilabel class imbalance is a highlighted topic among the artificial intelligence (AI) community in recent years (Daniels & Metaxas, 2017). Class imbalance also affects multilabel learning, whereby the high and low instance distributions of each label are largely imbalanced and broadly varied. The situation is exacerbated in the presence of numerous labels and low densities (Maheshwari et al., 2017; Zhang et al., 2020). Besides, the level of imbalance among multilabel datasets is greater than the binary or multiclass datasets (Charte et al., 2013).

Class imbalance is generally resolved using the under-sampling method. For instance, Rao and Reddy (2020) presented the under-sampling strategy (i.e., KNN-US) to reduce the less prominent instances from majority subsets to solve imbalanced dataset. The proposed method in Rao and Reddy (2020) identified the mostly misclassified instances based on the k-nearest neighbor (KNN) technique. Onan (2019) presented consensus clustering-based under-sampling method to lessen the number of instances of the majority class. Lin et al. (2017) presented two under-sampling strategies that also utilize the clustering technique. Zhang et al. (2018) proposed an approach based on the stacking and inverse random under-sampling methods. Zhang et al. (2018) used the inverse random under-sampling method to under-sample the majority class samples and applied the stacking approach to separate and classify the minority from the majority class. The inverse random under-sampling method was also employed by Tahir et al. (2012) before applying an ensemble classifier. A bidirectional resampling method (i.e., multilabel decoupling bidirectional resampling (ML-DBR)) at the data level was proposed by Zhou et al. (2020). The disparity of the labels was minimized by decoupling the extremely concurrent data of the majority and minority labels and by calculating the effect of the labels during resampling. The independence of the instances was then guaranteed. However, the ML-DBR approach was tested using seven benchmark multilabel datasets, including the Enron text dataset. The results showed that the proposed method was able to outperform several methods, namely REMEDIAL, REMEDIAL-HwR-ROS, and REMEDIAL-HwR-HUS (Charte et al., 2019). Three classifiers were used to classify the dataset, which were label powerset (LP), binary relevance (BR), and multilabel k-nearest neighbor (ML-kNN) (Zhang & Zhou, 2007). In terms of Micro-F values, the usage of ML-DBR

helped the employed classifiers to achieve higher results than the compared methods on five out of seven datasets. Kim et al. (2019) presented the principles of under-sampling technique to solve a class imbalance problem. Pereira et al. (2020) presented a Multilabel Tomek Link (MLTL) based on the Tomek Link resampling method. This under-sampling algorithm detected and eliminated the so-called Tomek links from the multilabel dataset. If they were the nearest neighbors but belonged to different groups, a pair of instances was considered a Tomek link. In addition to being a subsampling method, MLTL could be implemented in a post-process cleaning stage for the ML-SMOTE method. According to Pereira et al. (2020), the justification for using it as a post-process cleaning stage relied on the fact that the class groups were typically not well specified after applying ML-SMOTE, i.e., some instances from the majority class might invade the space of the minority class or vice versa. Consequently, the feature space could be cleaned and the edges between classes smoothed by the MLTL method.

Over-sampling is the second widely used method to resolve class imbalance. Sáez et al. (2016) applied this method in analyzing class characteristics, whereby the subsets of certain instances were identified in each class and increased individually. A novel reverse-nearest neighborhood-based over-sampling method for the class imbalance of a multilabel dataset was introduced by Sadhukhan and Palit (2019). All those points that included the query point as one of their neighbors had the reverse nearest neighborhood of a query point. However, the proposed method was tested using ten mutilabel datasets including the Enron text dataset. The results showed that the proposed method was able to outperform label-specific features (LIFT) (Zhang & Wu, 2014), random k-labelset (RAKEL) (Tsoumakas et al., 2010), improved baseline restoration (IBLR) (Cheng & Hüllermeier, 2009), cross-coupling aggregation (COCOA) (Zhang et al., 2020), calibrated label ranking (CLR) (Fürnkranz et al., 2008), synthetic minority over-sampling technique (SMOTE) (Chawla et al., 2002), adaptive synthetic sampling (ADASYN) (He et al., 2008), and USAM (Fernández et al., 2017). In terms of F-measure, the proposed method achieved higher results than the compared methods on nine out of ten datasets. Last et al. (2017) presented a combination of k-means clustering and SMOTE over-sampling, which was called K-means SMOTE. The proposed method avoided noise generation and effectively overcame the imbalance problem between and within

classes. Another over-sampling technique was introduced by Abdi and Hashemi (2016) based on the Mahalanobis distance. Moreo et al. (2016) presented a new over-sampling method, i.e., distributional random over-sampling (DRO), explicitly designed to identify the imbalanced text dataset for which the distributional hypothesis held, according to which the importance of a feature was somehow determined by its distribution in large data corporations. The proposed method generated new random minority class synthetic documents by exploiting the distributional properties of the terms in the collection. However, the proposed method was evaluated on three mutilabel datasets, including Reuters-21578 text dataset. The proposed method was compared against three methods, namely random over-sampling (RO) and SMOTE (Chawla et al., 2002), BSMOTE (Han et al., 2005) and DECOM (Chen et al., 2011). The proposed method obtained higher results than the comparative methods on all datasets in terms of F-measure. Li et al. (2014) presented an over-sampling approach that used the clustering technique and the Euclidean distance. Meanwhile, Rivera (2017) introduced an over-sampling approach based on noise reduction and selective sampling of the minority class to achieve good predictive abilities concerning its membership. Another widely used modification of over-sampling was the synthetic minority over-sampling technique (SMOTE) (Charte et al., 2015; Díez-Pastor et al., 2015; Jian et al., 2016). Koziarski et al. (2019) presented a radial-based over-sampling (RBO) method, which could find areas where artificial organisms of the minority class must be created based on estimating the imbalanced distribution of defects with radial basis functions. Two over-sampling methods, namely borderline-SMOTE1 and borderline-SMOTE2, were presented by Han et al. (2005) to over-sample the minority examples around the borderline.

The hybrid sampling method proposed by several studies (Dubey et al., 2014; Shi et al., 2018; Song et al., 2016; Wang, 2014) is a combination of the under-sampling and over-sampling techniques. This method showed promising results in comparison with the stand-alone methods. Dubey et al. (2014) carried out a systematical analysis of various sampling techniques by studying the effectiveness of different rates and types of under-sampling and over-sampling and a combination of both methods. Shi et al. (2018) proposed an under-sampling that selected the informative instances and features from the original dataset, whereas over-sampling balanced the majority class

instances. Song et al. (2016) proposed a hybrid of SMOTE and under-sampling technique by applying k-means. Wang (2014) proposed a simple integration between under-sampling and over-sampling to improve the classification result of support vector machine (SVM). All the results reported in the studies above showed that the hybrid sampling method is better than the stand-alone methods in terms of classification performance. For instance, in Song et al. (2016), the proposed hybrid sampling method of under-sampling and over-sampling achieved 6.4 percent higher than the under-sampling method in terms of F-measure across four datasets. Whereas the over-sampling in isolation achieved 2 percent lower than the hybrid sampling method.

Other types of hybrids entail the combination of one of the sampling methods and other methods, such as the combination of SMOTE and artificial immune recognition system (AIRS) (Wang & Adrian, 2013). Fang et al. (2017) presented a new method dealing with imbalance problem for multilabel classification called DEML. DEML transformed the whole label set of the multilabel dataset into some subsets and each subset was treated as a multilabel dataset with balanced class distribution to solve the class imbalance problem. DEML was tested using ten multilabel datasets including Bibtex and Enron datasets. The results showed that the proposed method was able to outperform CLR (Fürnkranz et al., 2008), RAkEL (Tsoumakas et al., 2010), ensemble of classifier chains (ECC) (Read et al., 2011), ML-kNN (Zhang et al., 2007), and BR (Tsoumakas et al., 2007). DEML achieved a higher average rating in terms of the micro-F1 and macro-F1 values. Xu et al. (2020) presented a hybrid of SMOTE and under-sampling with nearest neighbor based on random forest to solve the class imbalance problem. Galar et al. (2013) presented a novel approach to improve the ensembles of classifiers via a combination of under-sampling and boosting techniques known as EUSBoost. Feng et al. (2020) presented a hybrid method cluster-based under-sampling and SMOTE (CUSS) to handle class imbalance classification. Sun and Lee (2017) presented a two-stage multilabel hypernetwork (TSMLHN) method to deal with the class imbalance problem in multilabel learning. In TSMLHN, class labels were divided into two groups, i.e., common labels and imbalanced labels based on their imbalance ratios. The correlations between common labels and imbalanced labels were used to improve the learning performance of imbalanced labels. TSMLHN was tested

using 15 multilabel datasets including Bibtex and Enron datasets. The results showed that the proposed method was able to outperform BR-SVM (Boutell et al., 2004), ML-kNN (Zhang et al., 2007), CLR (Fürnkranz et al., 2008), RAkEL (Tsoumakas et al., 2010), ECC (Read et al., 2011), IBLR (Cheng & Hüllermeier, 2009), COCOA (Zhang et al., 2020), ML-ROS, ML-RUS, and MLSMOTE (Charte et al., 2015), and MLHN (Sun et al., 2016). In terms of macro-F, TSMLHN achieved higher results than the compared methods on 9 out of 12 datasets.

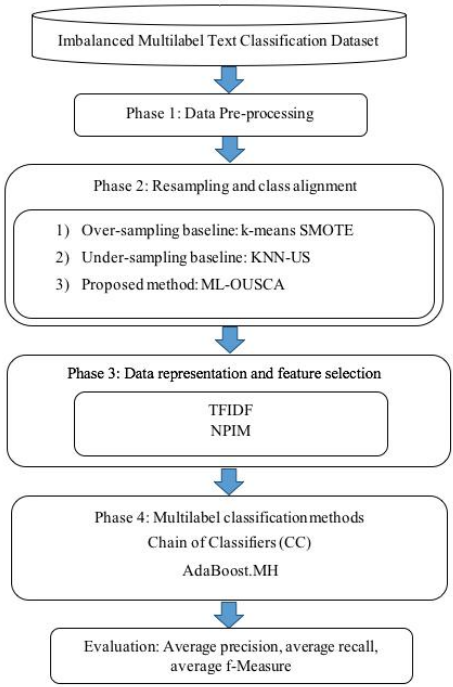
Class imbalance is yet an issue that is highly investigated in recent years. When the instances of a specific class outnumber other classes, this usually causes a poor result (Feng et al., 2020; García et al., 2018; Maurya et al., 2017; Sáez et al., 2016; Zhou et al., 2020). In machine learning, presenting an imbalanced dataset usually result in low classification accuracy. The reason is because the machine learning method can learn very little about the minority class. A true good classifier is when the classifier is able to classify a balanced amount of classes with high accuracy (Haixiang et al., 2017; Qiao et al., 2017; Xu et al., 2020). Generally, the most commonly used approaches to handle imbalanced classes are under-sampling and over-sampling and they provide competitive results when compared with more complex methods found in the literature (Charte et al., 2015; Moreo et al., 2016; Sun et al., 2017; Sáez et al., 2016). Over-sampling aims to balance classes' training examples through reproducing the minority class examples (last et al., 2017; Tahir et al., 2012; Tanha et al., 2020). On the other hand, under-sampling targets to balance the classes' training examples through the elimination of majority class examples (Charte et al., 2015; Pereira et al., 2020; Rao & Reddy, 2020; Sáez et al., 2016). Both of these approaches have limitations. For instance, under-sampling can discard potentially useful data, while over-sampling can increase the likelihood of overfitting. In order to overcome random over-sampling and under-sampling limitations for balancing the classes, this study proposes a new method by combining Multilabel Over-Sampling and Under-Sampling with Class Alignment (ML-OUSCA). The aim behind the combination is to deal with both the limitations of previous approaches in addressing the class imbalance problem.

METHODS

A complete framework of multilabel text classification was set up to evaluate the ML-OUSCA method (Figure 1). The framework consisted of four phases, namely (i) data pre-processing; (ii) resampling and class alignment; (iii) data representation and feature selection; and (iv) multilabel classification models. In the framework configuration, two baseline resampling algorithms, namely k-nearest neighbor under-sampling strategy (KNN-US) and K-means SMOTE, were used against the proposed ML-OUSCA algorithm. The learning algorithms and ensemble methods were constructed to determine the performance of ML-OUSCA against the two baselines of resampling model. Figure 1 shows the different combinations of the multilabel text classification architecture. The details of each phase are described in subsequent subsections.

Figure 1

Methodology for ML-OUSCA in Multilabel Text Classification



Data Pre-processing

Pre-processing is an essential stage before the application of machine learning approaches. It includes four steps: (i) tokenization, (ii) normalization, (iii) stop-word removal, and (iv) stemming. First, tokenization aims to turn the text of a certain document into an appropriate format for machine learning. The tokenization process involves a text to discrete fragmentation in between the separated units distinguished by a space or a certain indicator so that every unit matches a single word. Second, the normalization step focuses on cleaning the data by eliminating noise or unwanted data, such as special characters. Third, the stop-word task is applied to discard unnecessary words, such as conjunctions, pronouns, and prepositions. Finally, stemming refers to figuring out the root or stem of words. Stemming extracts the word's root form from its inflectional or derivational form, which is a necessary step for addressing high dimensional and sparse data, especially with multilabel text data classification.

Resampling and Class Alignment

This subsection describes two baseline methods, namely under-sampling: KNN-US method and over-sampling: K-SMOTE method. It also describes the new resampling proposed method (ML-OUSCA) for handling the class imbalance problem in multilabel text classification.

Baseline 1: Under-sampling: KNN-under Sampling Strategy (KNN-US)

KNN-US that was proposed by Rao and Reddy (2020) is the first baseline resampling method used in this work. KNN-US is one of the latest developments in under-sampling methods and is considered one of the state-of-the-art methods of resampling. The main idea of KNN-US is to recognize the mostly misclassified instances by taking into account the k-nearest neighbor technique. If all the nearest neighboring instances of a particular instance are of other classes, it means that the specific record is listed as a noisy or outlier instance and may therefore be excluded. The dataset is split into minority subsets and majority subsets , respectively, at the first stage of KNN-US. P is the minority subset of instances that are much lower when

compared to the other classes in the dataset. N is the majority subset of instances, which are more than the other classes in percentage. Through analyzing the intrinsic properties of the instances, the noisy and outlier records can be easily detected. Two main steps are taken into account. First (step 1), minority set data cleaning: pi = m′; where m′ ($0 \leq m' \leq m$), if $m/2 \leq m' < m$, then pi is an often-misclassified instance. Then, delete the m› instances from the minority set. Second (step 2), majority set data cleaning: ni = m′; where m' ($0 \leq m' \leq m$), if $m/2 \leq m' < m$, then pi is a mostly misclassified instance. Then, delete the m› instances from the m′ from the majority set (see Algorithm 1).

Algorithm 1: KNN-under sampling (KNN-US)

Input: Minority class dataset , Majority class dataset
P= set of the minority instances
N= set of the majority instances
m'= the number of majority nearest neighbors
T= the whole training set
m= the number of nearest neighbors

Step 1. Find mostly misclassified instances pi
1. Let us consider
2. m' = the number of majority nearest neighbors
3. pi = m'; where m' ($0 \leq m' \leq m$)
4. if $\leq m' < m$ then pi is a mostly misclassified instance. Then remove the instances m' from the minority set.

Step 2. Find noisy instances pi'
5. pi' = m'; where m' ($0 \leq m' \leq m$)
6. If m^'= m, i.e. all the m nearest neighbors of pi are majority examples,
pi' is considered to be noise or outliers or missing values and are to be n removed.
7. ni' = m'; where m' ($0 \leq m' \leq m$)
8. If m^'= m, i.e.all the m nearest neighbors of pi are minority examples,
ni' is considered to be noise or outliers or missing values and are to be removed.

Output: A new minority class dataset Sm

Baseline 2: Over-sampling: K-SMOTE (K-means Synthetic Minority Over-sampling Technique)

Over-sampling aims to increase the number in the training set of minority class members. The over-sampling method generates new minority class instances to eliminate the harms of skewed distribution. To evaluate over-sampling techniques, this work adopted K-SMOTE (Last et al., 2017), as shown in Algorithm 2. K-SMOTE is one of the recent advances of SMOTE and is considered to be one of the state-of-the-art over-sampling methods (Last et al., 2017).

K-SMOTE consists of three steps: clustering, filtering, and over-sampling. In the clustering step, the input space is clustered into k groups using k-means clustering. The filtering step selects those groups for over-sampling in order to maintain the minority class instances with a high percentage.

Algorithm 2: K-SMOTE - Synthetic Minority Over-sampling

Input: X (matrix of observations)
y (target vector)
n (number of samples to be generated)
k (number of clusters to be found by k-means)
irt (imbalance ratio threshold)
knn (number of nearest neighbors considered by SMOTE)
de (exponent used for computation of density; defaults to the number of features in X)

begin
Step 1: Cluster the input space and filter clusters with more minority instances than majority instances.
1. clusters ← k-means(X)
2. filtered clusters ← ∅
3. for c ∈ clusters do
4. if imbalance ratio < irt then
5. filtered clusters ← filtered clusters ∪ {c}
6. end
7. end

Step 2: For each filtered cluster, compute the sampling weight based on its minority density.
8. for f ∈ filtered clusters do
9. average minority distance (f) ← mean (Euclidean distances(f))

(continued)

10. density Factor(f) ← $\frac{\text{minorityCount(f)}}{\text{averageM inorityDistance(f) de}}$
11. sparsity Factor(f) ← $\frac{1}{\text{densityFactor(f)}}$
12. end

Step 3: Oversample each filtered cluster using SMOTE. The number of samples to be generated is computed using the sampling weight.

13. Generated samples ← ∅
14. for f ∈ filtered clusters do
15. number of samples ←
16. Generated samples ← generated samples ∪ {SMOTE (f, number of samples, knn)}
17. end
18. return generated samples
19. end

Proposed Method: ML-OUSCA

An imbalanced dataset is caused by unbalanced data distribution, leading to the poor performance of multilabel text classification algorithms because the classifiers are more inclined toward the majority than the minority data. This study proposes a new method based on both under-sampling and over-sampling of imbalanced classes. In the method, class labels are grouped into three major groups, namely major classes, minor classes, and class alignment (balanced classes). Over-sampling entails the random elimination of the majority classes to attain balanced distribution. In contrast, under-sampling involves the replication of the minority classes in achieving a balanced distribution with the majority classes.

For minority classes, new documents will be added based on the size of a minority class, average class size, and standard deviation. The aim is to increase their sizes to be nearest to the balanced class sizes. For majority classes, documents will be deleted.

In multilabel text classification, let $X \in R^d$ be the domain of documents and $Y = \{l_1, l_2, ..., l_q\}$ denote the finite set of labels. $D = \{(x_i, y_i) | 1 \leq i \leq N, x_i \in X, y_i \subseteq Y\}$ denotes the training data that consists of N documents and its related labels. y_i is a vector consisting of 1 and 0. Documents linked to a certain label are treated as positive or negative instances.

To overcome random over-sampling and under-sampling limitations, the proposed work balances the classes of training examples by combining and exploiting the power of over-sampling, under-sampling, and non-sampling methods. Under-sampling can only discard a few non-useful majority class examples, whereas over-sampling prevents overfitting by duplicating few randomly selected minority class examples. Furthermore, class alignment (balanced classes) that have suitable training examples (number of training examples that are not too high or too low) are identified, and their training examples are kept without being over-sampling and under-sampling.

The main idea of the proposed ML-OUSCA algorithm is derived based on median outlier detection and Chebyshev's Theorem (Amidan et al., 2005). Chebyshev's Theorem is applied to solve the class imbalanced data in multiple works (Amidan et al., 2005; Su & Hsiao, 2007) by estimating the likelihood of arriving at a value that differs from the mean by less than some degree of standard deviation. It then shows a percentage of how far the data is outside the standard deviation from the mean. The theory is described in Equation 1:

$$(1 - (\frac{1}{r^2})) \quad (1)$$

Chebyshev's Theorem states that at least $(1 - (\frac{1}{r^2}))$ of the items in any dataset will be within r standard deviations of the mean, where r is any value greater than 1. Based on Chebyshev's Theorem, at least 75 percent of the items must be within r = 2 standard deviations of the mean. At least 89 percent of the items must be within r = 3 standard deviations of the mean. At least 94 percent of the items must be within r = 4 standard deviations of the mean. For data that have a normal distribution, approximately 68 percent of the data values will be within r = 1 standard deviation of the mean and 95 percent of the data values will be within r = 2 standard deviations of the mean. Almost all of the items (99%) will be within r = 3 standard deviations of the mean.

The proposed ML-OUSCA algorithm (Algorithm 3) consists of the following main steps:

Step 1: Group samples according to their classes.

In this step, the samples in dataset *D* are rearranged, where

each sample is distributed into sample *S*, in which the total number of samples are equal to *Q* (number of labels), $D = \{S_1, S_2, S_3, ..., Q\}$. . They are distributed based on their belonging to each label.

Step 2: Obtain majority classes, minority classes, and class alignment (balanced classes) based on class sizes' median and quartiles.

This step starts by ranking the groups in *D* in ascending order. Then, the median of the samples is computed using Equation 2:

$$\text{median} = (Q + 1)/(2 \quad (2)$$

In order to identify the extreme values at the tails of the distribution, the samples are divided into quartiles. The following quantities (called fences) are calculated using Equations 3 and 4:

$$\text{lower inner fence} = \text{Quar}_1 - 1.5\text{IQ} \quad (3)$$
$$\text{upper outer fence} = \text{Quar}_3 + 3.0\text{IQ} \quad (4)$$

where *lower inner fence* is the median of the values from the high values quartile (Quar_1). *upper inner fence* represents the median of the values from the low values quartile (Quar_3).

The major (called $\text{Major}_{\text{classes}}$) and minor labels (called $\text{Minor}_{\text{classes}}$) are identified based on the median of (Quar_1) and (Quar_3). The class alignment (called $\text{Balanced}_{\text{classes}}$) that do not belong to (Quar_1) and (Quar_3) are identified and their training examples are kept without over-sampling and under-sampling. In other words, class alignment (balanced classes) are classes whose size is not more than or less than one standard deviation away from the mean.

Step 3: The mean and standard deviation of the class alignment (balanced classes) are calculated to determine the reduction size of majority classes and increment size of minority classes.

In order to recognize the amounts of examples to be added to the minority classes and removed from the majority classes,

the means and standard deviation of the class alignment (balanced classes) are calculated based on Class Mean Size (CMS) and cross-sectional standard deviation (CSSD) using Equations 5 and 6:

$$\text{Class Mean Size (CMS)} = \frac{\sum_{i=1}^{q} |S^j|}{q} \quad (5)$$

where CSSD is the cross-sectional standard deviation.

$$\text{CSSD} = \sqrt{\frac{\sum_{i=1}^{q} (|S^j| - \text{CMS})^2}{q-1}} \quad (6)$$

Step 4: Major classes are under-sampled.

In this step for majority classes, new documents will be deleted based on the size of a majority class, average class size, and standard deviation using Equation 7. The aim is to reduce their sizes to be nearest to the balanced class sizes.

$$\text{Reduct size} = |\text{MajorL}_i| - |\text{CMS} + 1 * \text{CSSD}| \quad (7)$$

Step 5: Minor classes are over-sampled.

For the minority classes, new documents will be added based on the size of a minority class using Equation 8. In addition, cosine similarity between x and other documents are used to increase their sizes of minority classes to avoid overfitting. It is added based on the size of a minority class, average class size, and standard deviation. The aim is to increase their sizes to be nearest to the balanced class sizes.

$$\text{Increment size} = |\text{CMS} - 1 * \text{CSSD}| - |\text{MinorL}_i| \quad (8)$$

Algorithm 3: ML-OUSCA

Input: Dataset: *D* with m features and q classes
Outputs: Pre-processed balanced data sample

Algorithm
Step 1 //Group samples according to their classes
1. For do
2. amples with Label(j)
3. End for

(continued)

Step 2 //Obtain majority classes, minority classes, and class alignment (balanced classes) based on class sizes' median and quartiles.

4. Rank *D* in ascending order
5. Calculate the median using Equation 2
6. Compute the lower inner fence Equation 3
7. Compute the upper inner fence Equation 4
8. For do
9. If ($|S_j|$ > upper inner fence)
10. $Major_{classes} \leftarrow Major_{classes} + l_j$
11. Else If ($|S_j|$< ower inner fence)
12. $Minor_{classes} \leftarrow Minor_{classes} + l_j$
13. Else
14. $Balanced_{classes} \leftarrow Balanced_{classes} + l_j$
15. End if
16. End for

Step 3 // Compute the mean and standard deviation of class alignment (balanced classes) for determining reduction size of majority classes and increment size of minority classes

17. Calculate Average Class Size using
18. Calculate class size standard deviation using cross-sectional standard deviation via 6

Step 4 // Under-sampling of major classes – Set reduction proportion (P)

19. For Each ($Major_L$ in $Major_{classes}$) do
20. Reduct size = $|MajorL_i| - |CMS + 1 * CSSD|$ Equation 7
21. for j = 1 to Reduct size do
22. x ← random (1;$|MajorL_i|$)
23. $S_j \leftarrow$ deletedocument (x,S_j)
24. End for

Step 5 // Over-sampling of minor classes

25. For Each ($Major_L$ in $Major_{classes}$) do
26. Increment size = $|CMS - 1 * CSSD| - |MinorL_i|$ Equation 8
27. for j = 1 to Increment size do
28. x ← random (1;$|MajorL_i|$) // Get randomly index of a document from
29. // using cosine similarity between x and S_j other documents
30. CX←GET_CLOSETdocuments(x,S^j)
31. End for

Data Representation and Feature Selection

This section describes the term frequency-inverse document frequency (TF-IDF) method, which is used as data representation in the experiment. It also demonstrates the used feature selection method called normalized pointwise mutual information (NPMI).

TF.IDF Model

In text classification, the feature values and a vector of features (terms) are used to describe a document (Adel et al., 2019; Johnson & Zhang, 2014; Mao et al., 2019; Taha & Tiun, 2016). TF.IDF is a well-known text representation method, which works by assigning a weight to each word (feature) (Chen et al., 2016; Mashaan Abed et al., 2013; Zubiaga, 2018). It finds the important phrases or words in a specific document and calculates the combination of the term frequency and inverse document frequency. This scenario entails the frequency of the word w in document *D*. The weight of a term is determined using two measures: (1) – the frequency of a term in a single document; and (2) – the number of documents in the corpus containing the specified term. is the total number of documents. From each document, only a few terms are selected (terms that have the highest). All other terms (terms that have the lowest*)* are removed from the document. Terms in a document are assigned their using Equation 9:

$$\text{TF.IDF}_{i} = \text{TF}_{i} \cdot \log\left(\frac{1}{i,2}\right) \tag{9}$$

Normalized Pointwise Mutual Information Features Selection

The mutual information feature selection measures the common information that is found between the terms and the labels (Kermani, et al., 2019; Lim et al., 2017). The common information MI (*t, c*) is found in between the class *c*, while the term *t* is distinct on the level of co-occurrence between a feature f_j and a class c_i (Li et al., 2017; Lim et al., 2017). In this work, the NPMI feature selection method was adopted to select features for each class according to co-occurrence measure between a feature f_j and c_i a class. NPMI between the feature and its classes (Lim et al., 2017) is calculated using Equations 10 and 11:

$$\text{NPMI (class} = c_i, f_j) = \frac{\text{PMI}(c_i, f_j)}{\sum_{f_k} \text{PMI}\,(c_i, f_k)} \tag{10}$$

$$\text{PMI (class} = c_i, f_j) = \text{lin}\,\frac{p(c_i, f_j)}{p(c_i)\,p(f_j)} \tag{11}$$

Multilabel Classification Models

For evaluation, two multilabel learning models, namely (i) chain of classifier (CC) based on a binary relevance method, and (ii) AdaBoost. MH, were adopted. These approaches were selected because they are considered as the state-of-the-art multilabel classification algorithms and often used in the works of imbalanced data (Al-Salemi, et al., 2018; Pant et al., 2018; Taha & Tiun, 2016).

Chain Classifiers Based on Binary Relevance Method

A combination of multiple classifiers to solve a single task is called chained classifiers (CC). The classifiers can be trained independently by different datasets (Taha & Tiun, 2016). This work utilized the proven binary classifiers, i.e., Naive Bayes (NB) classifier, k-nearest neighbor (KNN) classifier, and SVM (Mirończuk & Protasiewicz, 2018).

AdaBoost.MH.

AdaBoost.MH constructs several weak classifiers iteratively and subsequently groups them into a final classifier that can estimate the multiple labels for a particular instance. Through integration and training, a boosting algorithm transfers a weak classifier to a strong one, which is what the AdaBoost algorithm does as an adaptive booster. The AdaBoost algorithm is capable of adjusting the weight distribution of the training samples adaptively and selecting the best weak classifier out of the sample weight distribution consistently to integrate all the weak classifiers and vote by a given weight to build a robust classifier. AdaBoost.MH is a multilabel version of AdaBoost algorithm (Al-Salemi et al., 2018; Pant et al., 2018).

Evaluation Measurements

The performance of these classification methods is measured by classifying the experimental results into four groups using Equations 5, 6, and 7, respectively. The first group is true positive (TP), entailing correctly assigned documents. The second group is false positive (FP), consisting of falsely assigned documents. The third is false negative (FN), as the set of documents that

were not incorrectly assigned to the class. Finally, the fourth is true negative (TN), as the set of documents that were not correctly assigned to the class. Besides, this study adopted three multilabel evaluation measurements that are commonly used in multilabel classification (Sharef et al., 2014; Taha et al., 2020; Taha & Tiun, 2016), which can be referred to in Equations 12, 13, and 14:

i. Average precision metric, M_PRECISION, evaluates the proportion of the correctly predicted relevant, as shown in Equation 12:

$$\text{M\_PRECISION} = \sum_{i=1}^{d} \frac{TP_i}{PT_i + FP_i} \tag{12}$$

ii. Average recall metric, M-RECALL, calculates the proportion of the correctly predicted relevant (true) labels that were correctly identified, as shown in Equation 13:

$$\text{M\_RECALL} = \sum_{i=1}^{d} \frac{TP_i}{PT_i + FN_i} \tag{13}$$

iii. Average F-measure metric, is the balance mean of both M_PRECISION and M_RECALL, as shown in Equation 14:

iv.

$$M_{F\beta} = \sum_{i=1}^{d} \frac{(\beta^2 + 1)\text{M\_Precision} \times \text{M\_Recall}}{\beta^2 \text{M\_Precision} + \text{M\_Recall}} \tag{14}$$

RESULTS AND DISCUSSION

This study evaluated the strengths of the proposed ML-OUSCA algorithm in the multilabel text classification context, in which AdaBoost.MH and CC were used as the classifiers for multilabel text classification. Main experiments involving K-SMOTE, KNN-US, and ML-OUSCA had been carried out using the framework of Figure 1. In addition, a five-fold cross-validation was utilized to evaluate all the experiments.

Dataset

As described in Table 1, the Bibtex, Enron, and Reuters-21578 corpus datasets, which are publicly available multilabel text classification domains, were used. Table 1 shows the number of instances, number of attributes, number of labels, cardinality, density, diversity, and average imbalance ratio per label (avgIR). Cardinality measured the average number of classes for each instance, whereas density entailed cardinality divided by the number of labels. Diversity involved the percentage of class sets present in the dataset divided by the number of possible label sets. The avgIR measured the average degree of imbalance of all classes. Therefore, the greater the avgIR, the greater the imbalance of the dataset.

Table 1

Summary of the Multilabel Text Classification Standard Data

Dataset	In-stances	Attri-butes	Classes	Cardi-nality	Den-sity	Diver-sity	avgIR
Bibtex	7395	1836	159	2.402	0.015	0.386	12.498
Enron	1702	1001	53	3.378	0.064	0.442	73.953
Reuters-21578	6000	500	103	1.462	0.014	0.135	54.081

Results

This study conducted two kinds of experiments using AdaBoost and CC classifiers for evaluation. The first experiment was conducted with baseline models (K-SMOTE and KNN-US) and the proposed ML-OUSCA method using AdaBoost for evaluation.

The second experiment employed the same settings and datasets that were used in the first experiment, and CC was applied instead of AdaBoost. The experiments were categorized based on the usage of AdaBoost and CC. Each experiment had three resampling methods, which were K-SMOTE, KNN-US, and ML-OUSCA.
NPMI was used as a feature selection method with feature sizes ranging from 250 to 2250 and with a constant increase of 250 each time. Tables 2, 3, and 4 show the selected features (labeled feature selection set) for each dataset using both classification methods.

Table 2 describes the results of using K-means SMOTE as the over-sampling method, and Table 3 shows the results of using KNN-US as the under-sampling method. Table 4 presents the ML-OUSCA results.

Table 2

Performance (Average F-measure) of Over-sampling Algorithm: K-means SMOTE on CC and AdaBoost

	Bibtex dataset		Enron dataset		Reuters-21578 dataset	
Feature selection set	AdaBoost	CC	AdaBoost	CC	AdaBoost	CC
250	74.39	75.81	72.57	71.36	69.98	69.85
500	77.99	75.78	74.28	68.52	71.35	71.91
750	78.76	74.31	74.76	72.82	74.68	68.95
1000	80.9	74.2	74.79	72.51	73.11	69.42
1250	**81.46**	75.2	76.33	72.88	77.23	70.74
1500	80.08	77.61	75.15	**73.56**	76.34	71.19
1750	80.74	**79.35**	75.63	73.56	70.2	**72.74**
2000	78.56	76.15	**76.71**	72.65	76.9	71.04
2250	80.72	77.79	73.72	71.78	74.61	71.46

Table 3

Performance (Average F-measure) of Under-sampling Algorithm: KNN-US on CC and AdaBoost

	Bibtex dataset		Enron dataset		Reuters-21578 dataset	
Feature selection set	AdaBoost	CC	AdaBoost	CC	AdaBoost	CC
250	79.95	72.99	71.07	66.16	72.55	73.98
500	75.98	75.39	73.86	73.78	76.96	**80.74**
750	78.65	78.45	74.57	73.48	77.99	77.43
1000	80.67	**84.74**	75.9	72.91	81.95	79.06
1250	75.64	76.97	71.59	73.63	**83.65**	77.07
1500	82.36	82.81	72.35	73.61	79.76	79.35
1750	82.87	80.59	72.89	74.08	79.77	79.6
2000	**84.41**	77.83	74.8	76.32	77.46	79.19
2250	82	76.73	**77.68**	**80.96**	81.05	77.95

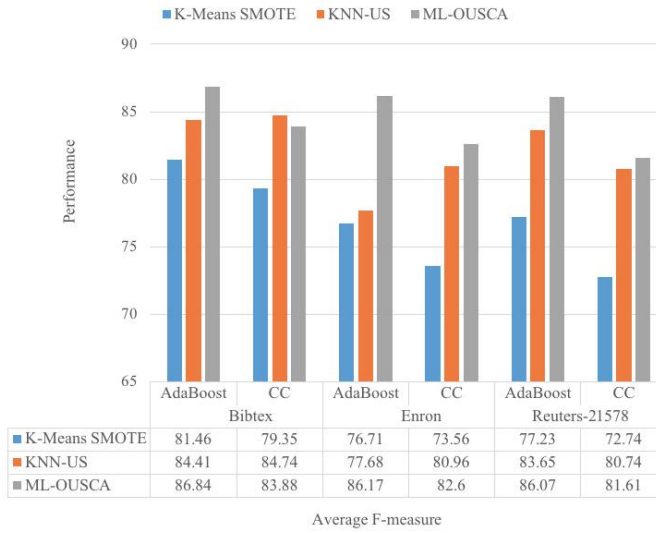
Table 4

Performance (Average F-measure) of the Proposed ML-OUSCA on CC and AdaBoost.MH

	Bibtex Dataset		Enron Dataset		Reuters-21578 Dataset	
Feature selection set	AdaBoost	CC	AdaBoost	CC	AdaBoost	CC
250	78.2	81.92	82.39	79	78.13	77.17
500	82.72	81.5	83.02	76.8	79.21	80.86
750	83.57	78.42	83.19	79.72	85.25	76.16
1000	**86.84**	77.6	83.86	79.77	82.95	78.97
1250	85.45	79.56	84.72	**82.6**	85.45	77.63
1500	84.21	82.95	82.78	81.6	84.71	79.18
1750	86.05	**83.88**	84.08	81.04	80.5	**81.61**
2000	84.51	81.32	**86.17**	81.13	**86.07**	78.68
2250	85.03	81.71	84.05	78.69	84.12	79.91

Figure 2

Performance of ML-OUSCA, KNN-US and K-means SMOTE on the Multilabel Text Classification Problem (Applied with AdaBoost and CC) using Full Size of Datasets



The results shown in Tables 2, 3, 4, and 5 are summaries of using the best sets of features for each classification method on all the datasets. Table 5 categorizes the experiment into evaluation models (AdaBoost and CC). Each experiment had three resampling methods (labeled K-means SMOTE, KNN-US, and ML-OUSCA) applied to each of the described datasets in Table 1.

Table 5

Summary of K- means SMOTE, KNN-US, and ML-OUSCA Best Results Given in Tables 2, 3, and 4.

Clas-sification method	Resampling method	FS sets Bibtex/ Enron/ Reuters	Bibtex Dataset	Enron Dataset	Reuters Dataset
AdaBoost	K- means SMOTE	1250/ 2000/ 1750	81.46	76.71	77.23
	KNN-US	2000/ 2550/ 1250	84.41	77.68	83.65
	ML-OUSCA	**1000/ 2000/ 1750**	**86.84**	**86.17**	**86.07**
CC	K- means SMOTE	1750/ 1500/ 1750	79.35	73.56	72.74
	KNN-US	1000/ 2250/ 500	84.74	80.96	80.74
	ML-OUSCA	**1750/ 1500/ 1750**	**83.88**	**82.6**	**81.61**

DISCUSSION

The obtained results are summarized in Figure 2, presenting the effect of the proposed ML-OUSCA method on the multilabel text classification models based on all the datasets. It compared the classification accuracy of ML-OUSCA and the baseline methods, namely KNN-US and K-means SMOTE. The results also demonstrated that the multilabel text classification models could be improved further if the inherited imbalance problem was solved.

The results obtained by ML-OUSCA with AdaBoost.MH was stable (consistently high) regardless of the imbalance problem. As seen in Figure 2, though the avgIR value of the Enron dataset was more than 70, ML-OUSCA obtained F-measure of 86.17 percent using AdaBoost and 82.6 percent using CC, respectively. Therefore, the proposed ML-OUSCA method was capable of handling imbalanced text problem, even with

high diversity in the size of imbalanced data (i.e., large value of avgIR). Figure 2 shows that ML-OUSCA significantly outperformed the other baseline sampling methods on all the multilabel text classification models (AdaBoost.MH and CC). Thus, to verify whether the above observations were statistically significant, a paired t-test was carried out involving the attained results of the proposed method and the two baseline methods on all datasets. First, the t-test result obtained between the proposed ML-OUSCA and baseline method KNN-US was $p = 0.000388$. Second, the t-test between results obtained by the proposed ML-OUSCA and baseline method K-means SMOTE was $p = 0.009999$. However, in order to conclude the t-test, a significance level of 0.05 was employed in this study. Based on the archived p values, it can be concluded that the results of the proposed method were significantly better than those of the baseline methods.

In analyzing the cause for these results, it is believed that KNN-US might increase the likelihood of overfitting, whereas K-means SMOTE method might lead to overgeneralization due to disregarding the majority class instances. Therefore, ML-OUSCA could provide an effective solution for the problem of classifying the imbalanced dataset to overcome several limitations of the baseline methods, such as losing important information and adding trivial information. The proposed method drew a new training set by over-sampling small size classes and under-sampling big size classes according to training examples by combining and exploiting the power of over-sampling, under-sampling, and non-sampling methods. The results showed that the classification performances of ML-OUSCA significantly outperformed the other baseline sampling methods in all datasets.
Besides, with the consistently higher results of AdaBoost.MH as compared to CC in all of the experiments (see Table 5) across all the datasets, AdaBoost.MH should be chosen as the ensemble classifier. This is because the AdaBoost.MH model aims to reduce the number of misclassified labels. It works by setting the weights to the training samples and classifiers in order to ensure the accuracy of the classification.

In other words, it can be concluded that for the best model for multilabel text classification, given the choice of baseline resampling and the proposed ML-OUSCA method to tackle imbalanced dataset

and ensemble classifiers of AdaBoost.MH and CC, one should choose the proposed ML-OUSCA with AdaBoost.MH as the classifier.

CONCLUSION

This study presented a new method, ML-OUSCA, to solve the class imbalance problem in multilabel classification. Instead of using all training instances, the proposed method constructed a new training set by using over-sampling on the minority classes, and under-sampling on the majority classes. Over-sampling and under-sampling were used to avoid the curse of class imbalance problem, a common problem in a majority of large-scale multilabel classification problems. The proposed ML-OUSCA was applied on well-known multilabel text classification datasets, namely Reuters-21578, Bibtex, and Enron. The results indicated the superiority of the proposed ML-OUSCA method as opposed to the baseline methods identified in the literature. Based on the results, the study concludes that combining multilabel over-sampling and under-sampling can help to achieve higher classification accuracy than using any of the above methods isolation.

ACKNOWLEDGMENT

This work was supported partially by Universiti Kebangsaan Malaysia under Research Grant FRGS/1/2020/ICT/02/UKM/02/1.

REFERENCES

Abdi, L., & Hashemi, S. (2015). To combat multi-class imbalanced problems by means of over-sampling techniques. *IEEE transactions on Knowledge and Data Engineering*, *28*(1), 238–251. https://doi.org/10.1109/TKDE.2015.2458858

Adel, A., Omar, N., Albared, M., & Al-Shabi, A. (2019). Feature selection method based on statistics of compound words for Arabic text classification. *International Arab Journal of Information Technology, 16*(2), 178–185.

Kermani, F. Z., Eslami, E., & Sadeghi, F. (2019). Global Filter–Wrapper method based on class-dependent correlation for text

classification. *Engineering Applications of Artificial Intelligence*, *85*, 619-633. https://doi.org/10.1016/j.engappai.2019.07.003

Ali, H., Salleh, M. N. M., Saedudin, R., Hussain, K., & Mushtaq, M. F. (2019). Imbalance class problems in data mining: A review. *Indonesian Journal of Electrical Engineering and Computer Science*, *14*(10.11591). https://doi.org/10.11591/ijeecs.v14.i3.pp1560-1571

Al-Salemi, B., Ayob, M., & Noah, S. A. M. (2018). Feature ranking for enhancing boosting-based multi-label text categorization. *Expert Systems with Applications*, *113*, 531–543. https://doi.org/10.1016/j.eswa.2018.07.024

Amidan, B. G., Ferryman, T. A., & Cooley, S. K. (2005, March). Data outlier detection using the Chebyshev theorem. In *2005 IEEE Aerospace Conference* (pp. 3814–3819). IEEE. https://doi.org/10.1109/AERO. 2005.1559688

Boutell, M. R., Luo, J., Shen, X., & Brown, C. M. (2004). Learning multi-label scene classification. *Pattern Recognition*, *37*(9), 1757–1771. https://doi.org/10.1016/j.patcog.2004.03.009

Cascaro, R. J., Gerardo, B. D., & Medina, R. P. (2019, December). Aggregating filter feature selection methods to enhance multiclass text classification. In *Proceedings of the 2019 7th International Conference on Information Technology: IoT and Smart City* (pp. 80–84). https://doi.org/10.1145/33 77170.3377209

Charte, F., Rivera, A. J., del Jesus, M. J., & Herrera, F. (2015). MLSMOTE: Approaching imbalanced multilabel learning through synthetic instance generation. *Knowledge-Based Systems*, *89*, 385–397. https://doi.org/10.1016/j.knosys.2015.07.019

Charte, F., Rivera, A. J., del Jesus, M. J., & Herrera, F. (2015). Addressing imbalance in multilabel classification: Measures and random resampling algorithms. *Neurocomputing*, *163*, 3–16. https://doi.org/10.1016/j.neucom.2014.08.091

Charte, F., Rivera, A. J., del Jesus, M. J., & Herrera, F. (2015). MLSMOTE: Approaching imbalanced multilabel learning through synthetic instance generation. *Knowledge-Based Systems*, *89*, 385–397. https://doi.org/10.1016/j.knosys.2015.07.019

Charte, F., Rivera, A., del Jesus, M. J., & Herrera, F. (2013, September). A first approach to deal with imbalance in multi-label datasets. In *International Conference on Hybrid Artificial Intelligence Systems*, (Vol. 8073, pp. 150–160). Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-40846-5_16

Charte, F., Rivera, A. J., del Jesus, M. J., & Herrera, F. (2019). Dealing with difficult minority labels in imbalanced mutilabel data sets. *Neurocomputing*, *326*, 39–53. https://doi.org/10.1016/j.neucom.201 6.08.158

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, *16*, 321–357. https://doi.org/10.1613 /jair.953

Chen, E., Lin, Y., Xiong, H., Luo, Q., & Ma, H. (2011). Exploiting probabilistic topic models to improve text categorization under class imbalance. *Information Processing & Management*, *47*(2), 202–214. https://doi.org/10.1016/j.ipm.2010.07.003

Chen, K., Zhang, Z., Long, J., & Zhang, H. (2016). Turning from TF-IDF to TF-IGM for term weighting in text classification. *Expert Systems with Applications*, *66*, 245–260. https://doi.org/10.1016/j.eswa. 2016.09.009

Cheng, W., & Hüllermeier, E. (2009). Combining instance-based learning and logistic regression for multilabel classification. *Machine Learning*, *76*(2–3), 211–225. https://doi.org/10.1007/s10994-009-5127-5

Daniels, Z., & Metaxas, D. (2017, February). Addressing imbalance in multi-label classification using structured Hellinger forests. In *Proceedings of the AAAI Conference on Artificial Intelligence*, (Vol. 31, No. 1). https://ojs.aaai.org/index.php/AAAI/article/view/10908

Díez-Pastor, J. F., Rodríguez, J. J., García-Osorio, C., & Kuncheva, L. I. (2015). Random balance: ensembles of variable priors classifiers for imbalanced data. *Knowledge-Based Systems*, *85*, 96–111. https://doi.org/10.1016/j.knosys.2015.04.022

Dubey, R., Zhou, J., Wang, Y., Thompson, P. M., Ye, J., & Alzheimer's Disease Neuroimaging Initiative. (2014). Analysis of sampling techniques for imbalanced data: An n=648 ADNI study. *NeuroImage*, *87*, 220–241. https://doi.org/10.1016/j.neuroimage.2013.10.005

Fang, M., Xiao, Y., Wang, C., & Xie, J. (2014, November). Multi-label classification: Dealing with imbalance by combining labels. In *2014 IEEE 26th International Conference on Tools with Artificial Intelligence* (pp. 233-237). IEEE. https://doi.org/10.1109/ICTAI.2014.42

Feng, S., Zhao, C., & Fu, P. (2020). A cluster-based hybrid sampling approach for imbalanced data classification. *Review of Scientific Instruments*, *91*(5), 055101. https://doi.org/10.1063/5.0008935

Fernández, A., del Río, S., Chawla, N. V., & Herrera, F. (2017). An insight into imbalanced big data classification: Outcomes and challenges. *Complex & Intelligent Systems*, *3*(2), 105–120. https://doi.org/10.1007/s40747-017-0037-9

Fürnkranz, J., Hüllermeier, E., Mencía, E. L., & Brinker, K. (2008). Multilabel classification via calibrated label ranking. *Machine Learning*, *73*(2), 133–153. https://doi.org/10.1007/s10994-008-5064-8

Galar, M., Fernández, A., Barrenechea, E., & Herrera, F. (2013). EUSBoost: Enhancing ensembles for highly imbalanced datasets by evolutionary undersampling. *Pattern recognition*, *46*(12), 3460–3471. https://doi.org/10.1016/j.patcog.2013.05.006

García, S., Zhang, Z. L., Altalhi, A., Alshomrani, S., & Herrera, F. (2018). Dynamic ensemble selection for multi-class imbalanced datasets. *Information Sciences*, *445*, 22–37. https://doi.org/10.1016/j.ins. 2018. 03.002

Glazkova, A. (2020). A comparison of synthetic oversampling methods for multi-class text classification. *arXiv preprint arXiv:2008.04636*.

Haixiang, G., Yijing, L., Shang, J., Mingyun, G., Yuanyue, H., & Bing, G. (2017). Learning from class-imbalanced data: Review of methods and applications. *Expert Systems with Applications*, *73*, 220 –239. https://doi.org/10.1016/j.eswa.2016.12.035

Han, H., Wang, W. Y., & Mao, B. H. (2005, August). Borderline-SMOTE: A new over-sampling method in imbalanced data sets learning. In *International conference on intelligent computing 3644*, 878– 887. Springer, Berlin, Heidelberg. https://doi.org/10.1007/11538059_91

He, H., Bai, Y., Garcia, E. A., & Li, S. (2008, June). ADASYN: Adaptive synthetic sampling approach for imbalanced learning. In *2008 IEEE international joint conference on neural networks (IEEE world congress on computational intelligence)* (pp. 1322–1328). IEEE. https://doi.org/10.1109/IJCNN.2008.4633969

Japkowicz, N., & Stephen, S. (2002). The class imbalance problem: A systematic study. *Intelligent Data Analysis*, *6*(5), 429-449. https://doi.org/10.3233/IDA-2002-6504

Jian, C., Gao, J., & Ao, Y. (2016). A new sampling method for classifying imbalanced data based on support vector machine ensemble. *Neurocomputing*, *193*, 115–122. https://doi.org/10.1016/j.neucom.2016. 02.006

Johnson, R., & Zhang, T. (2014). Effective use of word order for text categorization with convolutional neural networks. *arXiv preprint arXiv:1412.1058*.

Kim, Y. G., Kwon, Y., & Paik, M. C. (2019). Valid oversampling schemes to handle imbalance. *Pattern Recognition Letters*, *125*, 661–667. https://doi.org/10.1016/j.patrec.2019.07.006

Koziarski, M., Krawczyk, B., & Woźniak, M. (2019). Radial-based oversampling for noisy imbalanced data classification. *Neurocomputing*, *343*, 19–33. https://doi.org/10.1016/j.neucom.2018.04.089

Koziarski, M., Woźniak, M., & Krawczyk, B. (2020). Combined Cleaning and Resampling Algorithm for Multi-Class Imbalanced Data with Label Noise. *Knowledge-Based Systems*, *204*, 106223. *arXiv preprint arXiv:2004.03406*.

Last, F., Douzas, G., & Bacao, F. (2017). Oversampling for imbalanced learning based on k-means and SMOTE. *arXiv preprint arXiv:1711.00837*. https://doi.org/10.1016/j.ins.2018.06.056

Li, F., Miao, D., & Pedrycz, W. (2017). Granular multi-label feature selection based on mutual information. *Pattern Recognition*, *67*, 410–423. https://doi.org/10.1016/j.patcog.2017.02.025

Li, H., Zou, P., Han, W. H., & Xia, R. Z. (2014). Imbalanced Data Classification Based on Clustering. In *Applied Mechanics and Materials* (Vol. 443, pp. 741–745). Trans Tech Publications Ltd. https://doi.org/10.4028/www.scientific.net/AMM.443.741

Lim, H., Lee, J., & Kim, D. W. (2017). Optimization approach for feature selection in multi-label classification. *Pattern Recognition Letters*, *89*, 25–23. https://doi.org/10.1016/j.patrec.2017.02.004

Lin, W. C., Tsai, C. F., Hu, Y. H., & Jhang, J. S. (2017). Clustering-based undersampling in class-imbalanced data. *Information Sciences*, *409*, 17–26. https://doi.org/10.1016/j.ins.2017.05.008

López, V., Fernández, A., García, S., Palade, V., & Herrera, F. (2013). An insight into classification with imbalanced data: Empirical results and current trends on using data intrinsic characteristics. *Information sciences*, *250*, 113–141. https://doi.org/10.1016/j.ins.2013.07.007

Maheshwari, S., Jain, R. C., & Jadon, R. S. (2017). A review on class imbalance problem: Analysis and potential solutions. *International Journal of Computer Science Issues (IJCSI)*, *14*(6), 43–51. https://doi.org/10.20943/01201706.4351

Mao, X., Chang, S., Shi, J., Li, F., & Shi, R. (2019). Sentiment-aware word embedding for emotion classification. *Applied Sciences*, *9*(7), 1334. https://doi.org/10.3390/app9071334

Mashaan Abed, A. L. I., Tiun, S., & Albared, M. (2013). Arabic term extraction using combined approach on Islamic document. *Journal of Theoretical & Applied Information Technology*, *58*(3).

Mirończuk, M. M., & Protasiewicz, J. (2018). A recent overview of the state-of-the-art elements of text classification. *Expert Systems with Applications*, *106*, 36–54. https://doi.org/10.1016/j.eswa.2018.03. 058

Moreo, A., Esuli, A., & Sebastiani, F. (2016, July). Distributional random oversampling for imbalanced text classification. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval* (pp. 805–808). https://doi.org/10.1145/2911451.2914722

Onan, A. (2019). Consensus clustering-based undersampling approach to imbalanced learning. *Scientific Programming*, *2019*. https://doi.org/10.1155/2019/5901087

Pant, P., Sabitha, A. S., Choudhury, T., & Dhingra, P. (2019). Multi-label classification trending challenges and approaches. In *Emerging Trends in Expert Applications and Security* (pp. 433–444). Springer, Singapore. https://doi.org/10.1007/978-981-13-2285-3_51

Patel, H., Singh Rajput, D., Thippa Reddy, G., Iwendi, C., Kashif Bashir, A., & Jo, O. (2020). A review on classification of imbalanced data for wireless sensor networks. *International Journal of Distributed Sensor Networks*, *16*(4), 1550147720916404. https://doi.org/10.1177/1550147720916404

Pereira, R. M., Costa, Y. M., & Silla Jr, C. N. (2020). MLTL: A multi-label approach for the Tomek Link undersampling algorithm. *Neurocomputing*, *383*, 95–105. https://doi.org/10.1016/j.neucom .2019. 11.076

Qiao, L., Zhang, L., Sun, Z., & Liu, X. (2017). Selecting label-dependent features for multi-label classification. *Neurocomputing*, *259*, 112–118. https://doi.org/10.1016/j.neucom.2016.08.122

Rao, K. N., & Reddy, C. S. (2020). A novel under sampling strategy for efficient software defect analysis of skewed distributed data. *Evolving Systems*, *11*(1), 119–131. https://doi.org/10.1007/s12530-018-9261-9

Read, J., Pfahringer, B., Holmes, G., & Frank, E. (2011). Classifier chains for multi-label classification. *Machine Learning*, *85*(3), 333. https://doi.org/10.1007/s10994-011-5256-5

Rivera, W. A. (2017). Noise reduction a priori synthetic over-sampling for class imbalanced data sets. *Information Sciences*, *408*, 146–161. https://doi.org/10.1016/j.ins.2017.04.046

Sadhukhan, P., & Palit, S. (2019). Reverse-nearest neighborhood based oversampling for imbalanced, multi-label datasets. *Pattern Recognition Letters*, *125*, 813–820. https://doi.org/10.1016/j.patrec.201 9.08.009

Sáez, J. A., Krawczyk, B., & Woźniak, M. (2016). Analyzing the oversampling of different classes and types of examples in multi-class imbalanced datasets. *Pattern Recognition*, *57*, 164–178. https://doi.org/10.1016/j.patcog.2016.03.012

Sharef, B. T., Omar, N., & Sharef, Z. T. (2014). An automated Arabic text categorization based on the frequency ratio accumulation. *International Arab Journal of Information Technology*, *11*(2), 213–221.

Shi, H., Gao, Q., Ji, S., & Liu, Y. (2018, July). A hybrid sampling method based on safe screening for imbalanced datasets with sparse structure. In *2018 International Joint Conference on Neural Networks* (*IJCNN*) (pp. 1–8). IEEE. https://doi.org/10.1109/IJCNN.2018.8489569

Song, J., Huang, X., Qin, S., & Song, Q. (2016, June). A bi-directional sampling based on K-means method for imbalance text classification. In *2016 IEEE/ACIS 15th International Conference on Computer and Information Science* (*ICIS*) (pp. 1–5). https://doi.org/IEEE. 10.1109/ICIS.2016.7550920

Su, C. T., & Hsiao, Y. H. (2007). An evaluation of the robustness of MTS for imbalanced data. *IEEE Transactions on Knowledge and Data Engineering*, *19*(10), 1321–1332. https://doi.org/10.1109/TKDE. 2007.190623

Sun, K. W., & Lee, C. H. (2017). Addressing class-imbalance in multi-label learning via two-stage multi-label hypernetwork. *Neurocomputing*, *266*, 375–389. https://doi.org/10.1016/j.neucom. 2017.05.04 9

Sun, K. W., Lee, C. H., & Wang, J. (2016). Multilabel classification via co-evolutionary multilabel hypernetwork. *IEEE Transactions on Knowledge and Data Engineering*, *28*(9), 2438–2451. https://doi.org/10.1109/TKDE.2016.2566621

Taha, A. Y., & Tiun, S. (2016). Binary relevance (BR) method classifier of multi-label classification for Arabic text. *Journal of Theoretical & Applied Information Technology*, *84*(3).

Taha, A. Y., Tiun, S., Abd Rahman, A. H., Ayob, M., & Sabah, A. (2020). A dynamic two-Layers MI and clustering-based ensemble feature selection for multi-labels text classification. *Journal of Advanced Computer Science and Applications, 11*(7).

Tahir, M. A., Kittler, J., & Yan, F. (2012). Inverse random under sampling for class imbalance problem and its application to multi-label classification. *Pattern Recognition*, *45*(10), 3738–3750. https://doi.org/ 10.1016/j.patcog.2012.03.014

Tanha, J., Abdi, Y., Samadi, N., Razzaghi, N., & Asadpour, M. (2020). Boosting methods for multi-class imbalanced data classification: An experimental review. *Journal of Big Data*, *7*(1), 1–47. https://doi. org/10.1186/s40537-020-00349-y

Toshniwal, D., & Venkoparao, G. (2017). Distributed sparse class-imbalance learning and its applications. *IEEE Transactions on Big Data*, *13*(9). https://doi.org/10.1109/TBDATA.2017.2688372

Tsoumakas, G., Katakis, I., & Vlahavas, I. (2006, September). A review of multi-label classification methods. In *Proceedings of the 2nd ADBIS workshop on data mining and knowledge discovery* (*ADMKD 2006*) (pp. 99–109).

Tsoumakas, G., Katakis, I., & Vlahavas, I. (2010). Random k-labelsets for multilabel classification. *IEEE Transactions on Knowledge and Data Engineering*, *23*(7), 1079–1089. https://doi.org/10.1109/TKDE.2010.164

Wang, K. J., & Adrian, A. M. (2013). Breast cancer classification using hybrid synthetic minority over-sampling technique and artificial immune recognition system algorithm. *International Journal Compute Science Electronics Engineering (IJCSEE), 1(*3), 408–412.

Weng, W., Lin, Y., Wu, S., Li, Y., & Kang, Y. (2018). Multi-label learning based on label-specific features and local pairwise label correlation. *Neurocomputing*, *273*, 385–394. https://doi.org/10.1016/j.neuc om.2017.07.044

Xu, Z., Shen, D., Nie, T., & Kou, Y. (2020). A hybrid sampling algorithm combining M-SMOTE and ENN based on random forest for medical imbalanced data. *Journal of Biomedical Informatics*, *107*, 103465. https://doi.org/10.1016/j.jbi.2020.103465

Zhang, L., Zhang, C., Quan, S., Xiao, H., Kuang, G., & Liu, L. (2020). A class imbalance loss for imbalanced object recognition. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *13*, 2778–2792. https://doi.org/10.1109/JSTARS.2020.2995703

Zhang, M. L., & Wu, L. (2014). Lift: Multi-label learning with label-specific features. *IEEE transactions on pattern analysis and machine intelligence*, *37*(1), 107–120. https://doi.org/10.1109/TPAMI.2014.2339815

Zhang, M. L., & Zhou, Z. H. (2007). ML-KNN: A lazy learning approach to multi-label learning. *Pattern recognition*, *40*(7), 2038–2048. https://doi.org/10.1016/j.patcog.2006.12.019

Zhang, M. L., Li, Y. K., Yang, H., & Liu, X. Y. (2020). Towards class-imbalance aware multi-label learning. *IEEE Transactions on Cybernetics*. https://doi.org/10.1109/TCYB.2020.3027509

Zhang, Y., Liu, G., Luan, W., Yan, C., & Jiang, C. (2018, March). An approach to class imbalance problem based on stacking and inverse random under sampling methods. In *2018 IEEE 15th International Conference on Networking, Sensing and Control* (*ICNSC*) (pp. 1–6). IEEE. https://doi.org/ 10.1109/ICNSC.2018.8361344

Zhou, S., Li, X., Dong, Y., & Xu, H. (2020). A decoupling and bidirectional resampling method for multilabel classification of imbalanced data with label concurrence. *Scientific Programming*, *2020*. https://doi.org/10.1155/2020/8829432

Zubiaga, A. (2020). Exploiting class labels to boost performance on embedding-based text classification. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management* (pp. 3357–3360) *arXiv preprint arXiv:2006.02104*.