Over-Under Sampling Approach with Adaptive Synthetic and Tomek Links Methods to Handle Data Imbalance in Sentence Classification on Halal Assurance Certificate Documents

Dadang Heksaputra[1, 2, 3,*], Rahmat Gernowo[1], R. Rizal Isnanto[1]

[1]Doctoral Program of Information System School of Postgraduate Studies, Diponegoro University, Semarang, Indonesia

[2]Faculty of Computer and Engineering, Department of Information System, Alma Ata University, Yogyakarta, Indonesia

[3]Alma Ata Center for Medical Informatics, Alma Ata University, Yogyakarta, Indonesia

Emails: dadang@almaata.ac.id; rahmatgernowo@lecturer.undip.ac.id; rizal_isnanto@yahoo.com

Abstract

Data imbalance is a common problem in machine learning, specifically in classification, in which examples in a dominant class outnumber examples in a minority class many times over. Besides, such a problem keeps a model unable to discover meaningful patterns for a minority class —hence, such a problem reduces model performance specifically in terms of Recall and F1-Score. In current work, activity is performed in overcoming data imbalance problem in sentence classification model of documents of assurance certificate for halal with a combination of over-sampling and under-sampling techniques, namely Adaptive Synthetic (ADASYN) and Tomek Links. Text Classification technique is adopted in classifying sentences regarding assurance of halal in documents of assurance certificate for halal Text Classification; since incorrect classification of such sentences is not preferable, therefore, it is important to make sure no information about halal product is missed out. Over-sampling techniques considered include the SMOTE, Borderline SMOTE, ADASYN, and SMOTENC, and under-sampling techniques include the Random Under-Sampler, Near Miss, and Tomek Links. As comparative result, best performance gain in terms of Accuracy (0.759), F1-Score (0.748), Recall (0.759), and Precision (0.768) is generated with ADASYN. In our use case, ADASYN + Tomek Links is effective; recall is important in case of classification of documents for assurance certificate for halal and therefore, we cannot miss any relevant sentences. The proposed approach remarkably enhances the accuracy level for halal-related sentence identification and can be adopted in the halal product checking systems in industries with a halal feature.

Keywords: Data Imbalance; Halal Assurance Documents; Adaptive Synthetic (ADASYN); Tomek Links; Text Classification; Halal Information Systems

1. Introduction

Data imbalance is one of the most complex problems in machine learning because one class in a dataset overshadows the rest, leading to a lopsided classification distribution [1]. [2] reveals that this phenomenon can restrain the efficacy of classification models. Most machine learning algorithms seem to struggle with the so-called dominant class being overrepresented and the minority-class being underrepresented. From a practical point of view, this is glaringly evident in cases such as reviewing halal assurance certificate documents where the data imbalance problem is arguably the most severe because it undermines the model's accuracy and yield unreliable predictions for the minority class that holds key consequences for the certification's authenticity.

In the case of halal products like food, supplements, cosmetics, etc, primary stepping-stone in their manufacturing and marketing is obtaining legality or halal certification. Halal certified products mean that the item meets all relevant requirements of the halal certification body. However, there is a slow motioned, albeit smart, method used to consider the diversity in raw materials and technical processes provided in the halal certification request, its reliability is

DOI: https://doi.org/10.54216/FPA.190215
Received: December 23, 2024 Revised: February 15, 2025 Accepted: March 05, 2025

established through manual assessment which is often slow and inaccurate and tends to overlook critical pieces of information when dealing with large amounts of intricate data. Devices of artificial intelligence such as machine learning models augment the efficiency in the examination of halal assurance documents; therefore, these systems hold great promise for the future.

Regardless, shortage of data in comparison to the problematic documents poses a great challenge in machine learning application and halal document assessment. As it stands, the model does not have the ability to identify which documents need more focus. Attempts to fix data imbalance issues are done via oversampling and under sampling approaches. Adaptive Synthetic Sampling (ADASYN) is an oversampling method where synthetic samples for the minority-class are crafted [3]. The method shifts from SMOTE by producing synthetic data that is more adaptable to complicated patterns by taking both the local distribution of data and its complexity into account [4].

In the meantime, under-sampling approaches such as Tomek Links are used for removing pairs of samples who are nearest neighbors within the majority and minority classes and are viewed to be outliers or noise. The simultaneous use of Adasyn for oversampling and Tomek Links for under sampling provides the best results as it optimally increases class diversity in the minority class and simultaneously decreases diversity in the majority classroom [4].

This strategy has worked well for other problems such as fraud detection and medical diagnosis. Still, its use for assessing halal assurance documents is yet to be fully researched. Given the rapid rising desire for expedited and precise halal certification, this method of approach makes a lot of sense. The intention of this research is to research the application of ADASYN and Tomek Links in the form of artificial intelligence on documents detailing the halal assurance to determine whether it can raise model accuracy and speed up the certification process.

What is clear from this research is that it offers a solution to the data imbalance in the evaluation of halal document processes, but more importantly, any other industry that is grappling with a similar problem. In so doing, the study lays a framework upon which more sophisticated approaches to solving data imbalance along with the use of artificial intelligence in improving the efficacy and productivity of business processes can be built.

2. Literature Review

2.1 Over-Sampling Approach

The problem of data imbalance is crucial when it comes to the building of machine learning models, especially for classification problems. This problem stems from the fact that a given dataset may have an unequal class distribution where one class, the majority class, is much larger than the other may, the minority class may. Data imbalance may cause a model to develop a bias towards the majority class, leading to a discrimination against the minority class. This is especially problematic in real-life scenarios where the inclusion of the minority-class is of utmost importance, like the analysis of halal assurance certificate documents.

In order to solve this problem, techniques to over-sample the dataset are applied to attempt to boost the presence of the minority class in the dataset. The best known over-sampling technique is the Synthetic Minority Oversampling Technique (SMOTE) which creates new synthetic examples by blending existing minority class examples [5]. SMOTE has been shown to be effective at classification because it reduces the bias towards the majority class and the model's sensitivity to the minority-class. The major disadvantage of SMOTE, however, is that it can produce irrelevant synthetic samples, which can change the prediction accuracy by being too different from the original data set's distribution.

To address this gap, different variations of SMOTE have been created. One of these is BorderlineSMOTE, which focuses on minority samples that are close to the decision line separating the two classes. This method aims to produce synthetic instances that are more appropriate in context to mitigate the chances of misclassification in extreme cases of imbalance. BorderlineSMOTE has been found to be useful where delicate sensitivity to the presences of the minority class is needed, such as in medical or financial data [6].

A more flexible approach is Adaptive Synthetic Sampling (ADASYN). ADASYN extends the concept of SMOTE by taking into consideration the distribution of the minority class and using it to determine how many synthetic samples will be created. ADASYN increases the sample allocation to regions which are more difficult to predict, optimizing the model's sensitivity to the minority class which makes it very useful for highly skewed data sets [7]. Still, ADASYN comes with the danger of generating some unrelated synthetic samples in some situations.

Furthermore, SMOTE-NC is an extension of SMOTE that deals with dataset that has both numerical and categorical features [8]. This technique uses numerical interpolation for the numeric features and probabilistic methods for the categorical features, thus giving it an edge when it comes to dealing with mixed features.

DOI: https://doi.org/10.54216/FPA.190215
Received: December 23, 2024 Revised: February 15, 2025 Accepted: March 05, 2025

2.2 Under-Sampling Approach

As with any aspect of machine learning, under-sampling plays a very dominant role in correcting data imbalance [9]. Data imbalance describes a case when one class overshadows the other, and this greatly alters how well the model performs. Models which are trained on imbalanced datasets are often overfitted as they are set to the most dominant class and lack the ability to identify patterns in the smaller, less sampled classes [9]. This is a major hurdle in day-to-day scenarios like fraud detection, where the chances of fraud happening are very low, but the impact is hugely detrimental.

In order to solve this problem, under sampling adjusts the number of samples taken from the majority class in order to achieve equilibrium. When compared to others, RandomUnderSampler is one of the easier and more straightforward techniques of under sampling. It takes a sample from the majority class and strips it until a desired ratio is reached between the classes. Although this method is simple and easy on the CPU, the most apparent problem it poses is the removal of valuable details from the majority class, which has a negative impact on the performance of the model, particularly in the case of complex datasets.

The Condensed Nearest Neighbour (CNN) uses under-sampling approaches which have k-NN algorithm methods, such as the sous ensemble of samples whereby irrelevant samples are neglected [4]. For removing imbalance within a dataset Eddited Neighbor Average (ENN), approach aims misclassified samples of the majority class as noise for efficient cleansing. A model can bingo masked irrelevant data, hence ethical dilemmas are avoided.

Under-sampling methods, however, has its drawbacks such as information loss from the majority class. As a solution, it is common to combine under-sampling with techniques such as over-sampling with SMOTE in which synthetic samples of minority class are created to achieve balanced samples [10].

2.3 Testing with Classification Method

Various over-sampling and under-sampling techniques need the appropriate classification models in order to testing them efficiency, which is what makes this process so intricate. At this stage of the research study, Logistic Regression, MultinomialNB, Random Forest, and XGBClassifier models were employed to test sampling techniques, which allowed for easy analysis and gave clear measures of effectiveness [11]. These models were chosen for their specific features and capabilities, particularly for their effectiveness toward data imbalance when used with the proper sampling modifiers.

Logistic regression is one of the first models that can be chosen because of how straight forward and easy to interpret it is. Along with the issue of meeting the other model’s complexity, logistic regression paired with appropriate sampling techniques provides a wide benchmark for the model's performance in the case of imbalanced datasets [12]. It makes it easy to understand how different features affects the final prediction value. However, without the proper sampling technique, this model does not foster sensitivity toward minority-class.

Random Forest is particularly efficient in working with datasets with intricate structure [13]. It improves its performance by encompassing several decision trees, which leads to more reliable outcomes [14]. Alongside sampling techniques of over-sampling or under-sampling, Random Forest works well with data that is disproportionally distributed.

XGBClassifier (Extreme Gradient Boosting) is an efficient boosting model that works well with tremendously inappropriate class distributions [15]. XGBClassifier updates the loss function recurringly, resulting in a more potent and precise model targeting the minorities. Its range of benefits includes addressing most challenges, including overfitting and multicollinearity, and providing great assistance in cases involving class imbalance.

3 The Materials and Methods

By integrating such methodologies, in this work, an attempt is made to develop a model for classification not only with high accuracy in terms of predicting positive, neutral, and negative classes but with balanced performance in terms of dealing with class imbalance as well. Performance evaluation will disclose to what extent accuracy and efficiency can be maximized in documents' evaluation for halal certification with the application of sampling techniques. Besides, through effective use of proper models and a systemic approach, in this work, an attempt is made to make the process of halal certification easier and allow SMEs to obtain legally approved and officially accredited certifications.

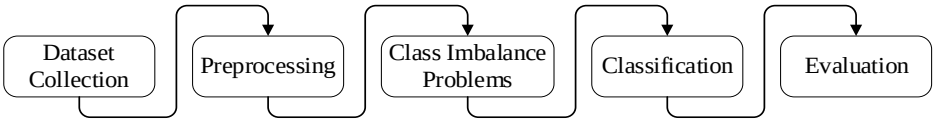

Figure 1. Research Stage Architecture

DOI: https://doi.org/10.54216/FPA.190215
Received: December 23, 2024 Revised: February 15, 2025 Accepted: March 05, 2025

3.1 Dataset Collection

The evaluation of halal certification documents for businesses, particularly Micro, Small, and Medium Enterprises (MSMEs), plays a crucial role in ensuring that market products comply with halal standards set by certification bodies. To support this certification process, a comprehensive and representative dataset is required for training classification models. The dataset used in this study comprises documents related to halal certificates obtained from certification agencies and data uploaded by MSMEs applying for halal certification. This dataset contains approximately 150 sentence-level data entries from a single company's documents, categorized into three main sentiment classes: negative, neutral, and positive. A major challenge with this dataset is the imbalance in the number of sentences among these classes, which can lead the classification model to favor the majority class while neglecting the minority ones. The sentiment categories relevant to this analysis are defined as follows:

$$Sentiment\ Categories = \{Positive, Negative, Neutral\} \quad (1)$$

For the initial training phase, manual labeling of a subset of the data was conducted to create a ground truth dataset. Specifically, given a document D_i, the labeling process is defined as:

$$Label(D_i) = Positive/Negative/Neutral \quad (2)$$

Here D_i refers to the document or sentence analyzed during the sentiment analysis process, and $Label(D_i)$ represents the sentiment label assigned based on the content of D_i, indicating whether the sentiment in the text is positive, negative, or neutral [16]. Labeling the sentences in halal certification documents may involve domain experts to ensure that the assigned labels are accurate. This process can be formalized as:

$$Label(D_i) = f(DomainExpert, D_i) \quad (3)$$

Where f is a function representing the contribution of domain experts in assigning appropriate labels to D_i [17]. This function ensures that sentiment labels are based on the deep knowledge and expertise of domain experts, enhancing the accuracy and relevance of the labels assigned to the sentences.

3.2 Preprocessing

To address class imbalance, preprocessing of data comes into play, such as dealing with missing values, feature normalization, and encoding categorical values [18]. Missing values are addressed through imputation, such as replacing them with mean or median values of numerical features [11]. In contrast, feature normalization aims at feature scales with value ranges that vary, and in the process, model performance is boosted [19]. Encoding comes in, in converting categorical values, such as types of products and certification, into numerical values that can then be processed via classification algorithms.

3.3 Class Imbalance Problems

The two sampling techniques used in this study are oversampling with ADASYN (Adaptive Synthetic Sampling) and under sampling with Tomek Links. ADASYN generates synthetic data for the minority-class by considering the distribution of data in that class. This technique focuses on the rarer or more difficult areas in the feature space to increase the diversity of the minority-class data [3]. As a result, the model can learn more complex patterns from the positive, neutral, and negative classes. On the other hand, Tomek Links identifies pairs of data that are close together but have different labels and removes data from the majority class that is near the minority class. This technique helps reduce noise and improves the balance between the three classes, allowing the model to learn from data that are more relevant.

ADASYN (Adaptive Synthetic Sampling) is a method for handling class imbalance by generating synthetic samples for the minority class, especially in areas that are difficult to separate from the majority class [20]. This process aims to improve the representation of the minority-class and enhance the performance of predictive models in classifying imbalanced data. The following are the steps and formulas used in ADASYN:

1. Identifying Nearest Neighbors (KNN): For each data point x_i from the minority class, its k-nearest neighbors (KNN) from the majority or other minority-class are identified. Euclidean distance is typically used to measure the proximity between points.
2. Determining the Number of Synthetic Samples: ADASYN calculates the number of synthetic samples to be generated for each point based on the difficulty level of separation between the minority and majority classes. If $N_{minority}$ is the number of data points in the minority-class and $N_{majority}$ is the number of data points in the majority class, ADASYN generates more synthetic points for the minority-class that is harder to separate.
3. Calculating the Difficulty Weight: The difficulty weight is calculated based on the number of minority-class points surrounding the data point x_i. The more minority-class points that are far from x_i, the higher the difficulty weight [21]. The difficulty weight for point x_i can be calculated as:

DOI: https://doi.org/10.54216/FPA.190215
Received: December 23, 2024 Revised: February 15, 2025 Accepted: March 05, 2025

$$w(x_i) = \frac{N_{minority}(x_i)}{k} \quad (4)$$

where $N_{minority}(x_i)$ is the number of nearest neighbors that belong to the minority class, and k is the number of nearest neighbors considered [20].

4. Generating Synthetic Points: Synthetic data points x_s are created by linearly interpolating between the data point x_i and one of its nearest neighbors x_j based on the difficulty weight. The formula for generating synthetic points is:

$$x_s = x_i + \lambda \cdot (x_j - x_i) \quad (5)$$

where λ is a random number drawn from the range [0, 1], and $(x_j - x_i)$ is the difference vector between points x_i and x_j.

5. Adding Synthetic Points to the Dataset: The generated synthetic points are then added to the dataset, improving the class balance and helping the model better predict the minority-class [22].

Tomek Links is a technique used to clean datasets by identifying pairs of points from different classes that are nearest neighbors and disrupt the decision boundary between the classes [23]. This method aims to reduce the overlap between the majority and minority classes, thereby creating a cleaner dataset and making it easier for the learning model to distinguish between the classes. Tomek Links is commonly used as a preprocessing step in class imbalance problems [24].

A pair of points x_i and x_j is called a Tomek Link if it satisfies two conditions. First, both points must come from different classes, i.e., "label"$(x_i) \neq$ "label"(x_j). Second, both points must be nearest neighbors [23]. This means that the distance between x_i and x_j must be smaller than the distance between x_i and any other point x_k, and the distance between x_j and any other point x_k. Mathematically, this can be formulated as:

$$d(x_i, x_j) = \min\{d(x_i, x_k) | \forall x_k \neq x_i\} \quad (6)$$

$$d(x_j, x_i) = \min\{d(x_j, x_k) | \forall x_k \neq x_j\} \quad (7)$$

where $d(x_i, x_j)$ is the distance between x_i and x_j, typically computed using Euclidean distance:

$$d(x_i, x_j) = \sqrt{\sum_{l=1}^{n} (x_{i,l} - x_{j,l})^2} \quad (8)$$

After identifying the Tomek Link pairs, the next step is to handle the points in these pairs. Typically, data from the majority class in the pair is removed because it is considered to disrupt the decision boundary. However, if the data is too noisy, both points may be deleted to further clean the dataset [24]. This step helps reduce the overlap between classes, ultimately improving the performance of the machine-learning model. Tomek Links is often used in conjunction with other resampling techniques, such as SMOTE (Synthetic Minority Over-sampling Technique). After balancing the dataset using SMOTE, Tomek Links can be applied to remove overlapping point pairs, resulting in a dataset that is not only balanced but also cleaned from noise. Therefore, Tomek Links plays a significant role in improving data quality in class imbalance problems.

3.4 Classification

After performing preprocessing and data balancing, the next step is to select the appropriate classification model. The choice of classification model is based on the characteristics of the dataset and the ability of each model to handle the challenges faced in evaluating halal certificate documents.

Logistic Regression is a technique in statistics and machine learning used for classification, especially when the dependent or target variable is categorical (binary or more). Below is an explanation of the basic formula in Logistic Regression:

In Logistic Regression calculate a linear function of the input features $x_1, x_2, \dots, x_n$, expressed as:

$$z = w_0 + w_1 x_1 + w_2 x_2 + \cdots + w_n x_n \quad (9)$$

where:

- $w_0, w_1, \dots, w_n$ are the coefficients or weights determined during the training process
- $x_1, x_2, \dots, x_n$ are the values of the input features

DOI: https://doi.org/10.54216/FPA.190215
Received: December 23, 2024 Revised: February 15, 2025 Accepted: March 05, 2025

The result of the linear function z is then processed by the sigmoid activation function to transform the output into a probability, which is a value between 0 dan 1 [25]:

$$p(y = 1 \mid x) = \frac{1}{1+e^{-z}} = \frac{1}{1+e^{-(w_0+w_1x_1+\cdots+w_nx_n)}} \quad (10)$$

- $p(y = 1 \mid x)$ is the probability that the target class y equals 1, given the features x.
- The sigmoid function $\frac{1}{1+e^{-z}}$ transforms the input value z into a probability.

After calculating the probability $p(y = 1 \mid x)$, the predicted class $\hat{y}$ is determined using a threshold (usually 0.5):

- If $p(y = 1 \mid x) \geq 0.5$, then the prediction $\hat{y} = 1$ (positive class).
- If $p(y = 1 \mid x) < 0.5$, then the prediction $\hat{y} = 0$ (negative class).

The loss function used in Logistic Regression is log loss or binary cross-entropy [26], which measures how well the model predicts the correct probabilities. Its formula is:

$$L = -\frac{1}{m}\sum_{i=1}^{m}\left[y_i\log\left(p(y = 1 \mid x_i)\right) + (1 - y_i)\log\left(1 - p(y = 1 \mid x_i)\right)\right] \quad (11)$$

where:

- m is the number of samples in the dataset.
- y_i is the actual (ground truth) value for sample i.
- $p(y = 1 \mid x_i)$ is the predicted probability for sample i.

Through the training process (usually using optimization methods like gradient descent), the coefficients $w_0, w_1, \dots, w_n$ are adjusted to minimize the value of the loss function and improve prediction accuracy.

Naive Bayes Classifier is a probabilistic classification method that uses Bayes' theorem to predict the class of a given data point. The fundamental principle of this method is the assumption of independence between features, referred to as "naive" because in reality, features are typically not independent of one another. Bayes' theorem is used to calculate the conditional probability of a class C given features $x = (x_1, x_2, \dots, x_n)$. Mathematically, Bayes' theorem can be written as follows:

$$P(C \mid x) = \frac{P(x \mid C) \cdot P(C)}{P(x)} \quad (12)$$

Where:

- $P(C \mid x)$ is the posterior probability of class C given the features x.
- $P(x \mid C)$ is the likelihood, or the probability of the features x given class C.
- $P(C)$ is the prior probability of class C before knowing the features x.
- $P(x)$ is the total probability of the features x (normalizing constant).

In Naive Bayes, it is assumed that every feature in the data is independent of one another, regardless of the class. This means that the probability $P(x \mid C)$ can be calculated as the product of the individual probabilities of each feature [27], expressed as:

$$P(x \mid C) = P(x_1, x_2, \dots, x_n \mid C) = \prod_{i=1}^{n} P\,(x_i \mid C) \quad (13)$$

By combining the independence assumption of features and Bayes' theorem, the prediction for class C_k given the features x can be written as:

$$P(C_k \mid x) = \frac{P(x_1 \mid C_k) \cdot P(x_2 \mid C_k) \cdot \cdots \cdot P(x_n \mid C_k) \cdot P(C_k)}{P(x)} \quad (14)$$

However, since $P(x)$ is constant for all classes, it often only consider the numerator, which determines the class with the highest probability [28]:

DOI: https://doi.org/10.54216/FPA.190215
Received: December 23, 2024 Revised: February 15, 2025 Accepted: March 05, 2025

$$\hat{C} = \underset{C_k}{\operatorname{argmax}} \left(P(C_k) \prod_{i=1}^{n} P\,(x_i \mid C_k) \right) \quad (15)$$

Random Forest is an ensemble of decision trees trained using the bagging (Bootstrap Aggregating) technique [14]. Each tree in the forest is trained on a different subset of data, which is randomly selected using the bootstrap technique (sampling with replacement). In addition, at each split in the decision tree, only a random subset of features is considered, which helps reduce the correlation between trees in the forest. Below are the equations and formulas associated with the Random Forest Classifier.

To create the subset dataset used to train the decision trees, bootstrap sampling is performed:

$$X_b = \{(x_1, y_1), (x_2, y_2), \dots, (x_m, y_m)\} \quad (16)$$

where:

- X_b is the subset of the dataset taken through random sampling (with replacement).
- (x_i, y_i) is the data and label for the i-th data in the original dataset X, and m is the number of samples in the dataset.

In each decision tree T_k that is trained, the data is split based on features that result in the best separation according to a specific criterion (e.g., Gini Impurity or Entropy). The splitting process for each node j in the decision tree is:

$$T_k(x) = \underset{c}{\operatorname{argmax}} P(c \mid x) = \frac{P(x \mid c) \cdot P(c)}{P(x)} \quad (17)$$

where:

- $T_k(x)$ is the prediction for data x using the k -th decision tree.
- $P(c \mid x)$ is the probability of class c given feature x.

After training, the prediction for class $\hat{y}$ is based on majority voting from all decision trees. If $n_{\text{estimators}}$ decision trees, the final prediction is the class selected by the majority of the trees:

$$\hat{y} = \underset{c}{\operatorname{argmax}} \sum_{k=1}^{n_{\text{estimators}}} \mathbb{I}\,(T_k(x) = c) \quad (18)$$

where:

- $\mathbb{I}(T_k(x) = c)$ is an indicator whether tree k predicts class c for data x.
- $\hat{y}$ is the final prediction, which is the class most frequently selected.

Random Forest reduces overfitting by selecting random subsets of data and features, resulting in a more robust model. This process reduces both bias and variance simultaneously:

$$\text{Varians}(F) = \frac{1}{n_{\text{estimators}}} \sum_{k=1}^{n_{\text{estimators}}} \text{Varians}\left(T_k(x)\right) \quad (19)$$

where:

- $\text{Varians}\left(T_k(x)\right)$ adis the variance of the predictions from the k -th decision tree.
- Ultimately, the total variance of Random Forest is lower compared to a single decision tree.

One of the measures used to determine the best split in a decision tree is Gini Impurity or Entropy:

- Gini Impurity for node j is calculated as:

$$Gini_j = 1 - \sum_{c=1}^{C} p\,(c \mid j)^2 \quad (20)$$

where $p(c \mid j)$ is the probability of class c at node j and C is the number of classes.

- Entropy for node j is calculated as:

DOI: https://doi.org/10.54216/FPA.190215
Received: December 23, 2024 Revised: February 15, 2025 Accepted: March 05, 2025

$$Entropy_j = -\sum_{c=1}^{C} p\,(c \mid j)\log_2 p(c \mid j) \quad (21)$$

where $p(c \mid j)$ is the probability of class c at node j.

Extreme Gradient Boosting (XGBoost) is a most common algorithm for both classification and regression in machine learning [29]. XGBoost is an optimized version of a Gradient Boosting Machine (GBM) and supports a range of techniques for model performance improvement, such as regularization, bias reduction, and overfitting controlling. In XGBoost, below mentioned are the mathematical equations, in which two important parts in objective function have been combined: Loss Function (difference between prediction and actual values) and a Regularization Term (penalty for complex model, for overfitting avoidance). In general, objective function $L(\theta)$ can be represented as:

$$L(\theta) = \sum_{i=1}^{n} \ell\,(y_i, \hat{y}_i) + \sum_{k=1}^{K} \Omega\,(f_k) \quad (22)$$

where:

- $\ell(y_i, \hat{y}_i)$ is the loss function that measures the error between the actual value y_i and the prediction $\hat{y}_i$.
- $\Omega(f_k)$ is the regularization term for the decision tree f_k, which is usually a penalty for the complexity of the tree (e.g., the number of leaves or tree depth).
- n is the number of samples, and K is the number of trees in the model.

For binary classification, the commonly used loss function is log loss:

$$\ell(y_i, \hat{y}_i) = -[y_i \log(\hat{y}_i) + (1 - y_i)\log(1 - \hat{y}_i)] \quad (23)$$

where:

- y_i is the actual label (0 or 1) for the i -th data.
- $\hat{y}_i$ is the predicted probability for the i -th data.

At each iteration step in boosting, XGBoost updates the model's prediction by adding the prediction from a new decision tree $f_k(x)$ that is built to reduce the residual error:

$$\hat{y}_i^{(t)} = \hat{y}_i^{(t-1)} + \alpha_t \cdot f_t(x_i) \quad (24)$$

where:

- $\hat{y}_i^{(t)}$ is the model's prediction at the t-th iteration for the i-th data.
- $\hat{y}_i^{(t-1)}$ is the model's prediction at the previous iteration.
- α_t is the weight for the t-th decision tree, optimized during training.
- $f_t(x_i)$ is the prediction from the t-th decision tree for the i-th data.

The final prediction for a data point x is a combination of the predictions from all decision trees in the model:

$$\hat{y} = \sum_{t=1}^{T} \alpha_t \cdot f_t(x) \quad (25)$$

where:

- T is the number of trees built in the model.
- α_t is the weight assigned to the t-th decision tree..
- $f_t(x)$ is the prediction from the t -th decision tree for data x.

XGBoost uses gradient descent to minimize the objective function. For each tree, parameter updates are performed by calculating the gradient and Hessian of the loss function with respect to the model's predictions:

$$g_i = \frac{\partial \ell(y_i, \hat{y}_i)}{\partial \hat{y}_i} \quad (26)$$

$$h_i = \frac{\partial^2 \ell(y_i, \hat{y}_i)}{\partial \hat{y}_i^2} \quad (27)$$

DOI: https://doi.org/10.54216/FPA.190215
Received: December 23, 2024 Revised: February 15, 2025 Accepted: March 05, 2025

where:

- g_i is the gradient for the i-th data, which measures the change in the loss function with respect to the prediction.
- h_i is the Hessian, which measures the change in the gradient with respect to the prediction.

3.5 Evaluation

Evaluating the performance of a classification model is crucial to measure how well the model can accurately predict sentence classes in halal certification documents. Since this dataset tends to be imbalanced, the evaluation metrics used must account for class imbalance. Some relevant metrics include Accuracy, F1-Score, Recall, and Precision[30]. Precision measures the accuracy of predictions for the positive class (halal), while Recall measures the model's ability to find all actual positive class examples. F1-Score, which is the harmonic mean of Precision and Recall, provides a comprehensive view of the balance between the three metrics. Accuracy measures the model's overall ability to classify all classes correctly across different thresholds. By using these metrics, a comparison can be made between models trained on original data, data after ADASYN, and data after the combination of ADASYN and Tomek Links, to observe the impact of sampling techniques on model performance.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (28)$$

$$\text{Precision} = \frac{TP}{TP+FP} \quad (30)$$

$$\text{Recall} = \frac{TP}{TP+FN} \quad (29)$$

$$\text{F1} - \text{Score} = \frac{2*(Recall*Precision)}{(Recall+Precission)} \quad (31)$$

Multiple aspects of performance need to be taken into account for the evaluation of a specific sentiment analysis model, which include measuring the model predictions accuracy. True Positive (TP) is defined as the instances in which the model predicted a positive sentiment and the actual label was positive. True Negative (TN) corresponds to the cases that the model predicted negative sentiment and the actual outcome was truly negative. On the other hand, FP (False Positive) occurs when the model predicts a positive sentiment when the text actually has neutral or negative sentiment and FN (False Negative) occurs when the model predicts the sentiment is negative while in fact, it is positive. These measurements help in understanding the level accuracy the model is able to achieve when determining the sentiment of the text and how many errors where made while attempting this classification.

4 Results and Discussions

4.1 Data Imbalance Analysis

The Over-Under Sampling with Adaptive Synthetic (ADASYN) and Tomek Links method is especially relevant in addressing data imbalance in the case of classification of documents for halal certification. Data imbalance, by leading to a dominance of the majority class in a classification model, can pose a serious issue, especially in high accuracy demands such as in the case of scanning for halal certification documents. In such an issue, even the less-prevalent minority class is often of high importance, denoting documents that need to be scanned for, and verified to meet, halal specifications.

The ADASYN mechanism works through adding samples to the minority class adaptively, in areas in which the model finds it challenging to make a prediction. It proves particularly useful in the case of documents that can vary a lot in terms of format and contents, and therefore, can make the minority class even more difficult to identify. With ADASYN, the model can manage such variation in data and learn complex patterns.

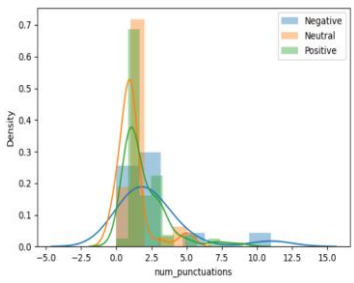

Figure 2. Numbers Punctuations

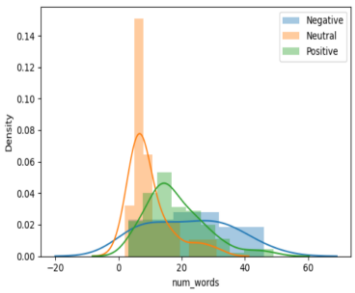

Figure 3. Numbers Words

DOI: https://doi.org/10.54216/FPA.190215
Received: December 23, 2024 Revised: February 15, 2025 Accepted: March 05, 2025

On the other hand, the Tomek Links algorithm, an under-sampling algorithm, actually discards pairs of samples with adjacent locations but in different classes. It helps in improving distribution in the data and reducing redundancy in the dominant class, allowing the model to devote more attention to significant features in the minority class. Tomek Links discards noise responsible for misclassification and helps in allowing the model to draw cleaner decision borders between classes. By integrating both of these methods, ADASYN for minority-class representation enhancement and Tomek Links for majority-class purification, it can improve performance in handling unbalanced data. By focusing on the purity of the data rather than its amount, such a method can provide a purer and more accurate model for assessing documents for halal certification, and in turn, improve effective halal certification management and adherence to applicable standards. To address data imbalance, over-sampling and under-sampling approaches are the most commonly adopted techniques. Over-sampling aims at growing samples taken from the minority class, utilizing approaches such as SMOTE (Synthetic Minority Over-sampling Technique) and ADASYN (Adaptive Synthetic Sampling). SMOTE generates artificial samples for the minority class through interpolation between samples in the dataset. However, in a few scenarios, SMOTE can produce untypical or excessive artificial samples, and overfitting can occur.

On the other hand, ADASYN is an improvement over SMOTE in that it synthetically generates additional samples in areas that are most difficult to classify, providing a truer reflection of complex minority classes. This can be beneficial in scenarios such as when processing documents for halal certification, where format and content variations in documents can make classification complex. By taking full use of ADASYN's ability to assign importance to challenging-to-predict areas, the model can handle complex cases in halal certification reports in a better manner, improving its performance and accuracy in predicting minority-class samples. It also reduces overfitting risks through generating more balanced data, in contrast to simply adding samples in a blind manner. In addition to over-sampling, under-sampling is performed to downsize samples of the dominant class in an effort to have a balanced distribution of samples. One of the most used under-sampling approaches is Tomek Links, whose function is to remove pairs of samples in close proximity but in a different class. It effectively reduces noise in the samples and clarifies the boundary between the decision and dominant/minority classes, allowing room for the model to work with a more representative group of samples. Using Tomek Links can reduce overfitting when over-sampling the dominant class and allow room for the model to work with the harder-to-classify samples of the minority class.

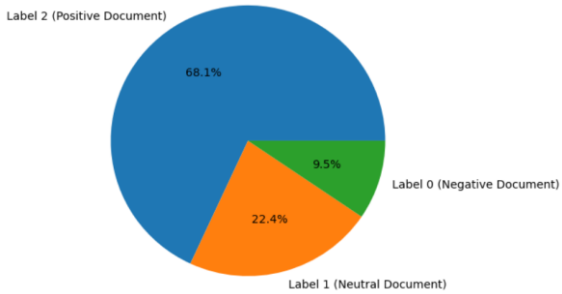

Figure 4. Imbalance Dataset before prosessed method

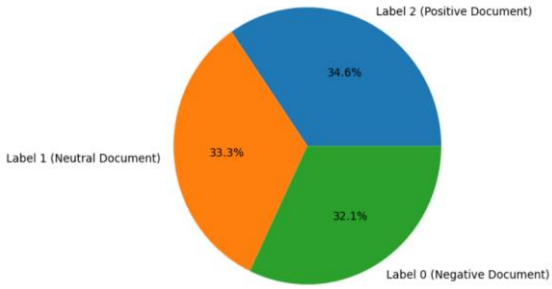

Figure 5. Balance Dataset after prosessed method

The hybrid over-sampling with ADASYN and under-sampling with Tomek Links holds significant potential for improving classification performance in imbalanced datasets. With the use of ADASYN, the minority-class sample is increased in a selective and adaptable manner, and with Tomek Links, the distribution of the majority class is refined to remove redundancy in the data that could impair model performance. The hybrid approach is focused on improving data quality, not quantity, for enhancing model predictive performance for the minority class, a critical consideration in the classification of documents for halal certification. In the analysis of documents for halal certification, an imbalance in information can affect the performance of a model in predicting accurately regarding a product's or a process's state of being halal. With the use of over-under sampling with the assistance of ADASYN and Tomek Link’s techniques, it is hoped that a model will be in a position to detect meaningful trends in information, counteract inaccuracies produced through a state of imbalance in classes, and produce a more effective system for supporting management in halal certification in the long-term future. In conclusion, such a practice has proven successful in enhancing accuracy and reliability in classification models in working with real-life information imbalance, in this case, in analyzing documents for halal certification, critical in safeguarding consumers and ensuring a long-term survival in an industry.

DOI: https://doi.org/10.54216/FPA.190215
Received: December 23, 2024 Revised: February 15, 2025 Accepted: March 05, 2025

4.2 Model Evaluation with Resampling Techniques

Table 1 shows the performance evaluation of over-sampling and under-sampling techniques in Logistic Regression with imbalanced datasets. In this, Accuracy, F1-Score, Recall, and Precision have been taken as performance metrics for evaluating the model performance post-applying these techniques.

In the over-sampling group, the algorithm ADASYN displays the best performance with an accuracy of 0.759, an F1-Score of 0.748, a Recall of 0.759, and a Precision of 0.768. This confirms that ADASYN performs best in balancing between both the majority and minority classes and, therefore, best in detecting the minority-class at a loss in overall accuracy. High performance is displayed by both SMOTE with an accuracy of 0.655, but less than that of ADASYN, and other algorithms such as Borderline SMOTE and SMOTENC, but less than that of ADASYN.

On the under-sampling, Random under Sampler generated an accuracy of 0.552, below most over-sampling algorithms. Much worse, at an accuracy of 0.276, was Near Miss, an extreme under-sampling algorithm, and it seems that removing information in the dominant class can result in loss of important information.

The Instance Hardness Threshold algorithm fared reasonably well with accuracy at 0.724 and an F1-Score at 0.69, outperforming a couple of under-sampling algorithms. Condensed Nearest Neighbour, Edited Nearest Neighbours, and Repeated Edited Nearest Neighbours all performed in a similar range, with accuracy at 0.655 consistently, but with less Precision.

Table 1 presents the evaluation results from various sampling methods applied to the Naive Bayes Classifier to handle data imbalance. The evaluation was conducted using four key metrics: Accuracy, F1-Score, Recall, and Precision, providing insights into how well the model can predict classes in imbalanced data.

Table 1: Evaluation of Testing

Category	Method	Logistic Regression Method				Naive Bayes Classifier Method			
		Accuracy	F1-Score	Recall	Precision	Accuracy	F1-Score	Recall	Precision
Over sampling	SMOTE	0.655	0.607	0.655	0.652	0.69	0.669	0.69	0.654
	Borderline SMOTE	0.69	0.655	0.69	0.645	0.724	0.716	0.724	0.71
	ADASYN	0.759	0.748	0.759	0.768	0.724	0.728	0.724	0.756
	SMOTENC	0.69	0.656	0.69	0.671	0.724	0.702	0.724	0.686
	Random Over Sampler	0.69	0.655	0.69	0.645	0.655	0.633	0.655	0.629
Under sampling	Random Under Sampler	0.552	0.547	0.552	0.566	0.552	0.554	0.552	0.566
	Near Miss	0.276	0.299	0.276	0.384	0.517	0.529	0.517	0.558
	Instance Hardness Threshold	0.724	0.69	0.724	0.706	0.655	0.564	0.655	0.54
	Condensed Nearest Neighbour	0.655	0.564	0.655	0.54	0.655	0.519	0.655	0.429
	Edited Nearest Neighbours	0.655	0.564	0.655	0.54	0.655	0.564	0.655	0.54
	Repeated Edited Nearest Neighbours	0.655	0.564	0.655	0.54	0.655	0.564	0.655	0.54
	AllKNN	0.655	0.564	0.655	0.54	0.655	0.564	0.655	0.54

DOI: https://doi.org/10.54216/FPA.190215
Received: December 23, 2024 Revised: February 15, 2025 Accepted: March 05, 2025

	Neighbourhood Cleaning Rule	0.655	0.564	0.655	0.54	0.655	0.564	0.655	0.54
	One Sided Selection	0.655	0.564	0.655	0.54	0.655	0.564	0.655	0.54
	Tomek Links	0.655	0.564	0.655	0.54	0.655	0.564	0.655	0.54
Our Approach		0.759	0.748	0.759	0.768	0.724	0.728	0.724	0.756

In the over-sampling category, the ADASYN method showed the best results with an Accuracy of 0.724, F1-Score of 0.728, Recall of 0.724, and Precision of 0.756. These results indicate that ADASYN is highly effective in balancing the classes and improving model performance, with a high Precision, indicating the model's ability to more accurately identify the positive class. Other methods like Borderline SMOTE and SMOTENC also provided good results, with an Accuracy of 0.724 and similarly high F1-Scores, though slightly lower than ADASYN. The Random over Sampler method yielded lower Accuracy (0.655) and lower Precision (0.629), suggesting that this technique may not be as effective in handling class imbalance compared to the others.

On the under-sampling, Random under Sampler showed an accuracy of 0.552, below that for over-sampling approaches. Near Miss fared badly with an accuracy of 0.517, and it seems that removing samples in the dominant class can actually impair model performance. Methods such as Instance Hardness Threshold and Condensed nearest Neighbour showed relatively poor Precision, but with a few having high Recall. Table 3 presents a comparison with a variety of types of sampling techniques adopted in balancing the Random Forest Classifier in handling imbalanced datasets. All processes of evaluation have been conducted using Accuracy, F1-Score, Recall, and Precision, providing a complete analysis of model performance in handling imbalanced datasets.

In the over-sampling category, the best results were achieved using the SMOTENC method, which yielded an Accuracy of 0.759, F1-Score of 0.728, Recall of 0.759, and Precision of 0.784. This indicates that SMOTENC is effective in improving the balance between the minority and majority classes, with high Precision, showing that the model can effectively identify the positive class.

The ADASYN and SMOTE methods also show good results, with an Accuracy of 0.724 and F1-Score of 0.66. Although these results are slightly lower than SMOTENC, they still demonstrate solid performance in handling class imbalance. On the under-sampling side, the Random under Sampler and Near Miss methods show very low Accuracy, with values of 0.414 and 0.379, respectively. This indicates that reducing the majority class samples significantly decreases the model's performance, particularly in terms of Recall and Precision. Other methods, such as Instance Hardness Threshold, yield relatively high Precision (0.793) but have lower Accuracy (0.517), suggesting that while this method can improve the prediction accuracy for the positive class, the model is less effective overall in predicting the minority class.

Table 2: Evaluation of Testing

Category	Method	Random Forest Classifier Method				Extreme Gradient Boosting Classifier Method with finetuning parameters objective='binary:logistic', eval_metric='logloss'			
		Accuracy	F1-Score	Recall	Precision	Accuracy	F1-Score	Recall	Precision
Over sampling	SMOTE	0.724	0.66	0.724	0.627	0.655	0.607	0.655	0.565
	Borderline SMOTE	0.655	0.612	0.655	0.58	0.69	0.655	0.69	0.645
	ADASYN	0.724	0.66	0.724	0.627	0.759	0.748	0.759	0.768
	SMOTENC	0.759	0.728	0.759	0.784	0.655	0.642	0.655	0.695

DOI: https://doi.org/10.54216/FPA.190215
Received: December 23, 2024 Revised: February 15, 2025 Accepted: March 05, 2025

	Random Over Sampler	0.69	0.631	0.69	0.588	0.69	0.634	0.69	0.588
Under sampling	Random Under Sampler	0.414	0.394	0.414	0.521	0.552	0.543	0.552	0.726
	Near Miss	0.379	0.353	0.379	0.631	0.448	0.498	0.448	0.587
	Instance Hardness Threshold	0.517	0.559	0.517	0.793	0.552	0.593	0.552	0.759
	Condensed Nearest Neighbour	0.69	0.631	0.69	0.588	0.586	0.558	0.586	0.557
	Edited Nearest Neighbours	0.655	0.564	0.655	0.54	0.655	0.564	0.655	0.54
	Repeated Edited Nearest Neighbours	0.655	0.564	0.655	0.54	0.655	0.564	0.655	0.54
	AllKNN	0.655	0.564	0.655	0.54	0.655	0.564	0.655	0.54
	Neighbourhood Cleaning Rule	0.69	0.631	0.69	0.695	0.655	0.564	0.655	0.54
	One Sided Selection	0.655	0.564	0.655	0.54	0.69	0.634	0.69	0.588
	Tomek Links	0.655	0.564	0.655	0.54	0.724	0.698	0.724	0.746
Our Approach		0.759	0.728	0.759	0.784	0.759	0.748	0.759	0.768

Table 2 presents the evaluation results for the Extreme Gradient Boosting Classifier with fine-tuning parameters such as objective='binary:logistic' and eval_metric='logloss' to handle data imbalance. The evaluation was conducted using Accuracy, F1-Score, Recall, and Precision metrics, which provide insights into the model's performance in handling both minority and majority classes. The Borderline SMOTE method has a lower Accuracy (0.621), with also lower F1-Score and Precision, indicating that while the model prioritizes data points closer to the decision boundary between classes, the results are not optimal in identifying the minority class.

In the under-sampling category, the Random under Sampler method yields a lower Accuracy (0.483) with similarly low Recall (0.483), but with high Precision (0.699), indicating that although the number of minority-class predictions is lower, the quality of predictions for that class is better. The Instance Hardness Threshold method performs quite well, with an F1-Score of 0.658 and Precision of 0.822, making it a highly effective method for improving the model's ability to accurately identify and predict the minority class. Overall, the Instance Hardness Threshold method stands out with high Precision, while SMOTE and ADASYN show good performance in terms of Recall and Accuracy, but face challenges in improving Precision. In the over-sampling category, the ADASYN method shows the best results with an Accuracy of 0.759, F1-Score of 0.748, Recall of 0.759, and Precision of 0.768. This indicates that ADASYN is highly effective in enhancing model performance by providing a better representation of the minority class. The Borderline SMOTE and SMOTE methods also yield good performance, with Accuracy scores of 0.69 and 0.655, respectively, but their F1-Score, Recall, and Precision are lower compared to ADASYN. The SMOTENC method provides relatively high Precision (0.695), but its Accuracy and F1-Score are slightly lower than ADASYN.

In the under-sampling category, Tomek Links performs the best, with an Accuracy of 0.724, F1-Score of 0.698, Recall of 0.724, and Precision of 0.746. This shows that the method is effective in improving data balance by removing neighboring sample pairs from different classes, thereby reducing redundancy in the majority class. Table 3 shows the testing evaluation results for the Random Forest Classifier, which has been fine-tuned with parameters such as criterion='entropy', max_samples=0.8, min_samples_split=10, and random_state=0 to address the issue of data imbalance. The evaluation is conducted using metrics such as Accuracy, F1-Score, Recall, and Precision. In the over-sampling category, the SMOTE and ADASYN methods give the same results, with Accuracy of 0.724, F1-Score of 0.66, and Recall of 0.724. Both have slightly lower Precision (0.627), indicating that the model predicts the majority class had better than the minority class.

Table 3: Evaluation of Testing

DOI: https://doi.org/10.54216/FPA.190215
Received: December 23, 2024 Revised: February 15, 2025 Accepted: March 05, 2025

Category	Method	Random Forest Classifier finetuning method with parameters criterion='entropy', max_samples=0.8, min_samples_split=10, random_state=0				Extreme Gradient Boosting Classifier finetuning method with parameters objective='binary:logistic', eval_metric='logloss', learning_rate=0.8, max_depth=20, gamma=0.6			
		Accuracy	F1-Score	Recall	Precision	Accuracy	F1-Score	Recall	Precision
Over sampling	SMOTE	0.724	0.66	0.724	0.627	0.655	0.607	0.655	0.652
	Borderline SMOTE	0.69	0.631	0.69	0.588	0.621	0.575	0.621	0.535
	ADASYN	0.724	0.66	0.724	0.627	0.655	0.603	0.655	0.559
	SMOTENC	0.69	0.656	0.69	0.671	0.655	0.648	0.655	0.707
	Random Over Sampler	0.69	0.634	0.69	0.588	0.621	0.593	0.621	0.575
Under sampling	Random Under Sampler	0.552	0.543	0.552	0.726	0.483	0.462	0.483	0.699
	Near Miss	0.448	0.498	0.448	0.587	0.483	0.539	0.483	0.637
	Instance Hardness Threshold	0.552	0.593	0.552	0.759	0.621	0.658	0.621	0.822
	Condensed Nearest Neighbour	0.586	0.558	0.586	0.557	0.586	0.558	0.586	0.557
	Edited Nearest Neighbours	0.655	0.564	0.655	0.54	0.69	0.631	0.69	0.695
	Repeated Edited Nearest Neighbours	0.655	0.564	0.655	0.54	0.69	0.631	0.69	0.695
	AllKNN	0.655	0.564	0.655	0.54	0.655	0.564	0.655	0.54
	Neighbourhood Cleaning Rule	0.655	0.564	0.655	0.54	0.69	0.631	0.69	0.695
	One Sided Selection	0.69	0.634	0.69	0.588	0.69	0.634	0.69	0.588
	Tomek Links	0.724	0.698	0.724	0.746	0.655	0.615	0.655	0.58
Our Approach		0.724	0.66	0.724	0.627	0.621	0.575	0.621	0.535

The Borderline SMOTE method yields an Accuracy of 0.69. However, it is F1-Score, Recall, and Precision are comparatively poorer, which indicates that there is an improvement of the balance in data distribution; however, the model performs poorly when compared to SMOTE and ADASYN approaches. Within the under-sampling domain, Tomek Links performs excellently with an Accuracy of 0.724, F1-Score of 0.698, Recall of 0.724, and Precision of 0.746, showing that after eliminating neighbor sample pairs of different classes the model, in higher-quality metrics such as Precision, improves. All in all, the joint use of ADASYN together with Tomek Links helps to achieve the same level of Accuracy and Recall that is achieved through the use of SMOTE in combination with ADASYN, which, however, shows lower F1-Score and Precision. This indicates that this combined approach, while achieving

DOI: https://doi.org/10.54216/FPA.190215
Received: December 23, 2024 Revised: February 15, 2025 Accepted: March 05, 2025

improvement in data balance, may require further work on the model to enhance its predictive performance on the minority class.

The findings reveal that Tomek Links can greatly improve Precision for the model, while SMOTE and ADASYN greatly improve the performance of the model regarding the minority class. These results are compared against the evaluation results of the Extreme Gradient Boosting Classifier set in Table 3, which has been trained on the parameters objective='binary:logistic', eval_metric='logloss', and learning rate of 0.8, maximum depth of 20, gamma of 0.6, reg_lambda of 0.1, and reg_alpha of 0.1. The measurements aquired include Accuracy, F1-Score, Recall, and Precision. For the category of over-sampling, the method SMOTE achieved Accuracy of 0.655, F1-Score of 0.607, Recall of 0.655, and Precision of 0.652. Generally this model performs great in terms of class imbalance and class Recall, however, claims can be made that F1 Score and Precision are indicators of lower overall accuracy and greater issues in identifying the minority class.

5. Conclusion

This research entails employing an over-under sampling technique with Adaptive Synthetic Sampling (ADASYN) and Tomek Links in a quest to address datasets imbalance in sentence classification for documents of halal certificates. In most scenarios, in machine learning, one of the classes overrepresents, and its counterpart, the minority-class, will not receive enough consideration. In such a scenario, one ends up with poor-classification models, especially in scenarios of official documents such as in halal certificates, in which sentences under a specific category must be accurately determined.

In this research, a variety of techniques for sampling have been tried in terms of performance in improving the model for classification, such as SMOTE, Borderline SMOTE, ADASYN, and SMOTENC, and a few under-sampling techniques such as Tomek Links, Random Under Sampler, and Near Miss. On performance evaluation over a variety of metrics, namely, Accuracy, F1-Score, Recall, and Precision, the following observations can be stated:

1. ADASYN demonstrated to have performed best in handling data imbalance. It performed well in generating synthetic samples for the minority-class, balancing out the distribution of the data and enhancing the detection of sentences regarding halal certificates. There were significant improvements in terms of Accuracy (0.759), F1-Score (0.748), Recall (0.759), and Precision (0.768) in testing evaluations. ADASYN performed well in adding diversity in terms of data, and the model could detect patterns regarding the minority-class with ease at little loss in accuracy for the majority-class.
2. Tomek Links, an under-sampling algorithm, fared better in terms of Precision. Despite a marginally compromised Accuracy and Recall when contrasted with ADASYN, the algorithm functioned well in erasing uncertain or high-risk examples with a probability of misclassification. By erasing neighboring pairs of examples in vastly different classes, Tomek Links boosted overall model performance, even at a loss in terms of other performance factors.
3. A combined model with Tomek Links and ADASYN generated acceptable performance, most notably in Recall improvement. Over-sampling and under-sampling balanced each other and maintained the model's ability to identify useful sentences in documents of halal certificates. The two approaches complemented each other, with diversity in the minority class being increased through ADASYN, and overfitting reduced through filtering out less indicative information with Tomek Links.
4. Despite some Precision and Accuracy loss in employing such a mixed model, Recall remained most important in the case of sentence classification in documents for halal certificates. That is, Recall will ensure that sentences actually relevant to the matter of halal will not escape undetected in the classification, and such is significant in ensuring information integrity in such documents of record.

Overall, the ADASYN and Tomek Links approaches handled data imbalance in certificate documents of halal sentence classification effectively. The approaches can be relied upon for model performance improvement in identifying sentences for halal certification, such that critical information is not overlooked and accuracy in classification is maintained. In such a manner, such an approach is promising for application in information systems for supporting evaluation and verification of halal in the food and halal product industries, with a view towards increased efficiency and accuracy in the halal certification process.

Acknowledgment: "We would like to extend our deepest gratitude to the Food and Drug Monitoring Agency (BPOM) and the Indonesian Ulema Council (MUI) DIY for your kind collaboration in this study. Most of the success in this study can be attributed to your collaboration and beneficial contribution of BPOM and MUI DIY. With your collaboration, in terms of information, expertise, and infrastructure, we have been successful in conducting this study and generating meaningful output. The authors express their gratitude to Alma Ata University for funding this study."

Funding: "This research received no external funding"

DOI: https://doi.org/10.54216/FPA.190215
Received: December 23, 2024 Revised: February 15, 2025 Accepted: March 05, 2025

Conflicts of Interest: "The authors declare no conflict of interest."

References

[1] Y. X. He, D. X. Liu, S. H. Lyu, C. Qian, and Z. H. Zhou, "Multi-class imbalance problem: A multi-objective solution," *Inf. Sci. (Ny)*, vol. 680, p. 121156, 2024, doi: 10.1016/j.ins.2024.121156.

[2] C. Jian, H. Chen, Y. Ao, and X. Zhang, "A two-stage learning framework for imbalanced semi-supervised domain generalization fault diagnosis under unknown operating conditions," *Adv. Eng. Informatics*, vol. 62, p. 102878, 2024, doi: 10.1016/j.aei.2024.102878.

[3] M. Mustapha et al., "A hybrid machine learning approach for imbalanced irrigation water quality classification," *Desalin. Water Treat.*, vol. 321, p. 100910, 2025, doi: 10.1016/j.dwt.2024.100910.

[4] N. Sakib, T. Paul, N. Anwari, and M. Hadiuzzaman, "Ensemble-based model to investigate factors influencing road crash fatality for imbalanced data," *Transp. Eng.*, vol. 18, p. 100284, 2024, doi: 10.1016/j.treng.2024.100284.

[5] L. Zhang, C. S. Oh, and Y. S. Choi, "Improved phase prediction of high-entropy alloys assisted by imbalance learning," *Mater. Des*, vol. 246, p. 113310, 2024, doi: 10.1016/j.matdes.2024.113310.

[6] M. Vasconcelos and L. Cavique, "Mitigating false negatives in imbalanced datasets: An ensemble approach," *Expert Syst. Appl.*, vol. 262, p. 125674, 2025, doi: 10.1016/j.eswa.2024.125674.

[7] B. Alabduallah et al., "Class imbalanced data handling with cyberattack classification using Hybrid Salp Swarm Algorithm with deep learning approach," *Alexandria Eng. J.*, vol. 106, pp. 654–663, 2024, doi: 10.1016/j.aej.2024.08.061.

[8] W. Q. Wang, R. Q. Ye, B. J. Tang, and Y. Y. Qi, "MultiThal-classifier, a machine learning-based multi-class model for thalassemia diagnosis and classification," *Clin. Chim. Acta*, vol. 567, p. 120025, 2025, doi: 10.1016/j.cca.2024.120025.

[9] Q. Zhou and B. Sun, "Adaptive K-means clustering based under-sampling methods to solve the class imbalance problem," *Data Inf. Manag.*, vol. 8, p. 100064, 2024, doi: 10.1016/j.dim.2023.100064.

[10] M. M. Chowdhury, R. S. Ayon, and M. S. Hossain, "An investigation of machine learning algorithms and data augmentation techniques for diabetes diagnosis using class imbalanced BRFSS dataset," *Healthc. Anal.*, vol. 5, p. 100297, 2024, doi: 10.1016/j.health.2023.100297.

[11] T. Hu et al., "Improved classification of soil As contamination at continental scale: Resolving class imbalances using machine learning approach," *Chemosphere*, vol. 363, p. 142697, 2024, doi: 10.1016/j.chemosphere.2024.142697.

[12] L. Zhao, W. Pu, R. Zhou, and Q. Shi, "A third-order majorization algorithm for logistic regression with convergence rate guarantees," *IEEE Signal Process. Lett*, vol. 31, pp. 1700–1704, 2024, doi: 10.1109/LSP.2024.3413306.

[13] Y. Ai et al., "A real-time road boundary detection approach in surface mine based on meta random forest," *IEEE Trans. Intell. Veh*, vol. 9, no. 1, pp. 1989–2001, 2024, doi: 10.1109/TIV.2023.3296767.

[14] T. Zhang et al., "A novel random forest variant based on intervention correlation ratio," *IEEE Trans. Emerg. Top. Comput. Intell*, vol. 8, no. 3, pp. 2541–2553, 2024, doi: 10.1109/TETCI.2024.3369320.

[15] Y. Yin et al., "A novel remaining useful life prediction approach combined eXtreme gradient boosting and multi-quantile recurrent neural network," *IEEE Access*, vol. 12, pp. 44648–44658, 2024, doi: 10.1109/ACCESS.2024.3381492.

[16] J. Deng et al., "SGO: An innovative oversampling approach for imbalanced datasets using SVM and genetic algorithms," *Inf. Sci. (Ny)*, vol. 690, p. 121584, 2025, doi: 10.1016/j.ins.2024.121584.

[17] J. Tu, S. Gu, and C. Hou, "Online imbalance learning with unpredictable feature evolution and label scarcity," *Neurocomputing*, vol. 610, p. 128476, 2024, doi: 10.1016/j.neucom.2024.128476.

DOI: https://doi.org/10.54216/FPA.190215
Received: December 23, 2024 Revised: February 15, 2025 Accepted: March 05, 2025

[18] C. Qi et al., "Leveraging visible-near-infrared spectroscopy and machine learning to detect nickel contamination in soil: Addressing class imbalances for environmental management," *J. Hazard. Mater. Adv.*, vol. 16, p. 100489, 2024, doi: 10.1016/j.hazadv.2024.100489.

[19] Z. Hou et al., "MVQS: Robust multi-view instance-level cost-sensitive learning method for imbalanced data classification," *Inf. Sci. (Ny).*, vol. 675, p. 120467, 2024, doi: 10.1016/j.ins.2024.120467.

[20] K. Zhong et al., "Prediction of slope failure probability based on machine learning with genetic-ADASYN algorithm," *Eng. Geol.*, vol. 346, 2025, doi: 10.1016/j.enggeo.2024.107885.

[21] X. Li, "Evaluation of optimization strategies for cross-border e-commerce logistics network based on ADASYN algorithm," *Procedia Comput. Sci.*, vol. 247, pp. 1036–1043, 2023, doi: 10.1016/j.procs.2024.10.125.

[22] M. Song et al., "Credit risk prediction based on improved ADASYN sampling and optimized LightGBM," *J. Soc. Comput.*, vol. 5, no. 3, pp. 232–241, 2024, doi: 10.23919/JSC.2024.0019.

[23] Q. Ning et al., "A novel method for identification of glutarylation sites combining borderline-SMOTE with Tomek Links technique in imbalanced data," *IEEE/ACM Trans. Comput. Biol. Bioinforma.*, vol. 19, no. 5, pp. 2632–2641, 2022, doi: 10.1109/TCBB.2021.3095482.

[24] F. Nhita, Adiwijaya, and I. Kurniawan, "Improvement of imbalanced data handling: A hybrid sampling approach using adaptive synthetic sampling and Tomek links," *Proc. 8th Int. Conf. Informatics Comput. ICIC 2023*, pp. 1–5, 2023, doi: 10.1109/ICIC60109.2023.10381929.

[25] R. Kaur and P. Singh, "Enhanced imbalanced data classification using hybrid SMOTE-ENN and optimized random forest," *Neural Comput. Appl.*, vol. 36, pp. 9827–9843, 2024, doi: 10.1007/s00521-023-08609-5.

[26] J. A. Pérez-Ortega et al., "A novel hybrid approach using SMOTE and neighborhood cleaning rule for imbalanced classification problems," *Expert Syst. Appl.*, vol. 229, p. 120694, 2023, doi: 10.1016/j.eswa.2023.120694.

[27] P. Wang et al., "A hybrid sampling approach using SMOTE and Tomek Links for financial fraud detection," *Comput. Ind. Eng.*, vol. 188, p. 109996, 2024, doi: 10.1016/j.cie.2023.109996.

[28] M. H. Mirza, I. Ahmad, and S. P. Singh, "A comparative study of ensemble methods for imbalanced text classification," *Inf. Process. Manag.*, vol. 61, p. 103419, 2024, doi: 10.1016/j.ipm.2023.103419.

[29] F. U. Rehman, A. A. Anwar, and A. R. Butt, "Cost-sensitive deep learning model for class-imbalanced medical diagnosis," *Biomed. Signal Process. Control*, vol. 95, p. 105002, 2024, doi: 10.1016/j.bspc.2023.105002.

[30] R. C. Prasad and A. S. Rao, "An effective deep learning framework for handling imbalanced big data classification," *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 36, no. 1, pp. 1121–1132, 2024, doi: 10.1016/j.jksuci.2023.01.005.

DOI: https://doi.org/10.54216/FPA.190215
Received: December 23, 2024 Revised: February 15, 2025 Accepted: March 05, 2025