Borderline Active Learning: Transactional Records in Alert-Feedback System

Bokyung Amy Kwon and Kyungtae Kang*

Department of Artificial Intelligence, Hanyang University, Ansan, Korea
Email: amykwon@hanyang.ac.kr(B.A.K.); ktkang@hanyang.ac.kr(K.K.)
*Corresponding author

Manuscript received November 11, 2024; revised November 27, 2024; accepted December 6, 2024; published December 27, 2024.

***Abstract*—Transactional records often exhibit highly imbalanced patterns, which can hinder the performance of data-driven models in alert-feedback systems. While oversampling techniques are commonly used to address this imbalance, they increase the total number of instances, leading to higher computational costs. Although the Active Learning (AL) approach is computationally expensive, it focuses only on the most informative samples, which can be more efficient for transactional records. Our experiments show that AL outperforms SMOTE and Borderline-SMOTE in terms of accuracy and AUPRC. Therefore, AL presents a promising approach for addressing the class imbalance problem in transactional records, without the added computational burden of synthetic samples.**

***Keywords*—active learning, oversampling, transactional records, precision-recall, machine learning**

I. Introduction

Although advancements in machine learning (ML) and artificial intelligence (AI) have significantly improved information retrieval and analysis across various fields, these techniques often fail to achieve satisfactory performance due to generic problems specific to the data. Transactional records serve as typical examples. In practice, many real-time transactional records are monitored by alert-feedback systems that trigger alarms when abnormal transactional patterns are detected based on data-driven models, to prevent significant damage from fraud. It is crucial to select the right data-driven model for the system to achieve satisfactory performance in identifying abnormal patterns. However, even with the right models, there are still generic challenges specific to the data. Transactional records are affected by *concept drift*, leading to continuously evolving abnormal patterns over time [1], and there is a latency issue in verifying the ground truth for the label information, which is critical in a supervised learning framework by hindering prompt knowledge acquisition for the data-driven model [2].

Moreover, abnormal patterns resulting from illegal activities are often not accessible, leading to a lack of positive information that results in a severe class imbalance problem. This skewed distribution in the class often causes a hard mining problem when detecting anomaly patterns, which ultimately exacerbates the performance of the given model and makes it harder for ML or AI based models to learn from positive events [3, 4]. Since all of these problems either explicitly or implicitly impact the overall performance of the data-driven model on transactional records, improving the model's performance would benefit from measuring any one of these issues. Our study specifically focuses on addressing imbalanced class distribution in transactional records for a given classifier.

There have been studies addressing class imbalance problems using sampling approaches. These studies aimed to balance the data by either oversampling observations in the minority class or undersampling observations in the majority class [5–7]. While these methods have shown improvements in performance in certain cases, they generally yield inconsistent results, and no specific studies have demonstrated improved performance when applying these approaches to transactional records. Additionally, increasing the data volume to force an even ratio could reduce the effectiveness of the alert-feedback system by increasing training time.

Our study proposes active learning (AL) as a solution to alleviate the challenges of learning from imbalanced data. AL focuses on selecting informative instances during model training, enabling effective learning without necessitating a balance in the data distribution. Moreover, to the best of our knowledge, there has been no direct application of AL as a solution for imbalanced data distribution.

After the introduction, we briefly describe related studies regarding studies specifically focusing on class imbalance problems as well as ones comparatively selecting the right models on transactional data. Next, we present our AL approach in detail, along with a brief explanation of the background needed in the context. In the Results section, we will present the experimental results based on standard performance measures. Finally, we summarize the strengths and weaknesses of our approach and highlight areas for future research.

II. Literature Review

The ML community generally addresses the class imbalance problem by either penalizing classification errors [5] or using sampling techniques to balance the distribution [6, 7, 9]. The former approach typically assigns cost matrices to the predicted classes during the selection process to minimize classification errors based on inductive learning. However, these methods may not be effective for noisy data, as they often require an increased number of rules. Therefore, we reviewed the literature focusing on the latter approach.

The Synthetic Minority Over-sampling Technique (SMOTE) specifically addresses the class imbalance problem in supervised learning [6]. This method generates synthetic instances of the minority class in feature space under the over-sampling framework. Synthetic instances are created along the line segments joining $N\%$ of the k nearest minority class neighbors depending on the necessary amounts of over-sampling. Less commonly, this approach creates a random

doi: 10.18178/ijml.2024.14.4.1170

point by multiplying the random number between 0 and 1 by the difference between the feature vector and its neighbors then add this result, to the original feature vector, along the line segment between two specific features. In the experiment using C4.5 as the base classifier, SMOTE demonstrated an improved area under the receiver operating characteristic curve (AUC). Borderline-SMOTE was proposed to address the same issue as SMOTE [7]. This method oversamples only the minority instances near the decision boundary, considering the relative importance of instances in that region. It constructs a 'danger set' to generate synthetic instances, similar to the process in SMOTE, where minority class instances are included in the danger set if more than half of their nearest neighbors belong to the majority class. Comparisons show that Borderline-SMOTE performs relatively better in terms of F-score [8]. Meanwhile, one-sided selection (OSS) [9] leaves the minority class instances untouched while eliminating noisy and unreliable instances from the majority class based on Tomek links [10]. This approach moves all misclassified instances in the training set to the C set, which consists of a randomly chosen negative instance and all positive instances, using 1-nearest neighbor (1-nn). It then removes instances belonging to Tomek links from the C set. This method has relatively low costs for learning, and evaluates performance using geometric mean of accuracy as a single measurement.

Apart from this problem, several comparative studies have been conducted to identify the most effective data-driven models using ML or neural network (NN) techniques on transactional records for an alert-feedback system [11–15]. On the comparison study among decision tree, k-nn, logistic regression (LR), Naïve Bayes and random forest (RF), it recommended the decision tree model considering prediction time as well as accuracy to detect fraudulent events [11]. Decision-tree based models are also independently compared according to different criteria, entropy and GINI index, with another combination of ML models in [12], the study recommended NN instead in terms of both accuracy and sensitivity. One study highlights the imbalanced characteristics of transactional records and examines whether model performance varies with different loss functions [13]. It suggests that the focal loss function may improve performance on imbalanced transactional records compared to standard cross-entropy [14]. However, adapting the focal loss function for training on transactional records is challenging, as it remains unclear how to set the weights for new data given the evolving characteristics. Considering the sequential procedures of the alert-feedback system, one study compared various classifiers under different parameter settings and selected the multi-layer perceptron (MLP) as the best classifier to minimize false positive rates, since reducing false positives decreases the time required for post-processing [15]. Additionally, both over-sampling and under-sampling techniques are applied to various ML models to address the inherent characteristics of transactional records and compare their performance [16]. The results indicate that oversampling approaches generally outperform undersampling approaches, recommending RF as the best option. However, it also cautions that results based on sampling techniques may not always be optimal.

III. Background

A. Alert-feedback System

The alert-feedback system consists of three sequential procedures to process real-time transactional records, as illustrated in Fig. 1. When a transaction occurs, it passes through the control gate, which authorizes normal transactions by checking authentication using various encryption methods within a short time frame. Once authorized, a data-driven model predicts whether the transaction exhibits an abnormal pattern, based on a set of labeled data and utilizing ML or AI techniques. Therefore, an effective classifier is crucial in this phase. If a transaction is flagged as abnormal, the verification process is initiated, triggering alarms and confirming the transaction's validity through expert review. During this phase, an annotation delay often occurs due to the latency in obtaining true label information, which hinders prompt knowledge acquisition.

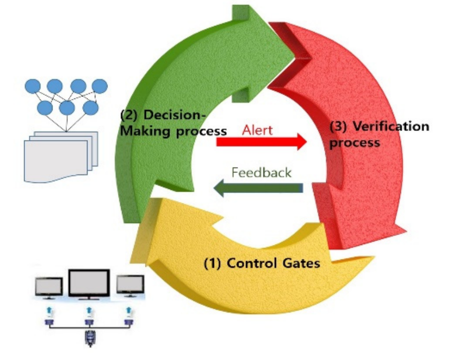


Fig 1. The process of the alert-feedback system.

B. Data-driven models

As data-driven models, we consider the following five models that have shown relatively better performance in previous studies [11–16].

- Logistic regression (LR)
  LR serves as a base classifier in our context, consistent with previous literature.

- Random Forest (RF) [17]
  RF is an ensemble-based approach that makes predictions by aggregating multiple trees built in parallel from independently chosen random subsets of the data. In terms of generalization error, it demonstrates robustness against overfitting, making it an effective strategy for building individual trees with higher strength and lower correlation. Generally, larger trees tend to provide better predictions as the number of features increases.

- *LightGBM* (L-GBM) [18]
  L-GBM is implemented as a gradient boosting decision tree (GBDT), but it uses a subset of instances with larger gradient values to compute information gain. By randomly dropping instances with small gradients, it achieves accurate estimation of information gain with a much smaller subset, avoiding the need to compute it at all split points. Additionally, L-GBM employs a strategy to bundle mutually exclusive features by treating two features as vertices and linking them if they

are not mutually exclusive. This problem is then approached as a graph coloring problem, solved in a greedy manner with a constant approximation ratio. These two main strategies lead to faster computation during training while maintaining accuracy and efficient memory consumption compared to other approaches.

- *XGboost* [19]
  XGBoost is a scalable end-to-end tree boosting algorithm based on an approximate greedy approach. It incorporates column sub-sampling, same to RF, and is designed to be sparsity-aware. Additionally, it regularizes the ensemble tree model to prevent overfitting. For approximation, XGBoost generates a set of split point candidates by deriving the ranks of feature quantiles, resulting in weighted quantiles that provide a theoretical guarantee. Furthermore, it handles all sparsity patterns uniformly, whether dealing with categorical or dense data, which speeds up computation by only visiting non-missing entries in the default direction of the branches.

- Neural network (NN)[20]
  NN is constructed with a standard architecture having two hidden layers illustrated in Fig 2, and the backpropagation is implemented by a perceptron converge procedure, which is simpler and maintains locality in weight space.

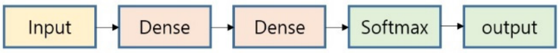


Fig. 2. Standard architecture of NNs in the context.

C. SMOTE technique

Based on previous literature indicating that over-sampling approaches generally outperform under-sampling methods, we selected SMOTE as our benchmark algorithm to address the class imbalance problem. Suppose that $\{x_i: i = 1, \cdots, N_p\}$ is a set of minority samples, and $N\%$ of the samples needs to be synthesized by SMOTE. (For simplicity, we convert $N\%$ to N^* using $(N/100)$ in the context.) For each instance of x_i, *k*-nearest neighbors are computed to generate synthetic samples as many as $N^* \cdot T$ as a total. The synthetic samples are generated at each x_i through the procedure called `Populate ($N^*$, i, *nn_ind*)', where *nn_ind* indicates the indices of *k*- nearest neighbors of $x_i$ . (The pseudocode of `PROCEDURE: Populate' is summarized in Algorithm 1.)

Algorithm 1: The pseudo code of SMOTE

```
PROCEDURE: Populate ($N^*$, i, nn_ind)
While ($N^*$ > 0) do
    Select randomly $j$ in the nn_ind
    Do linear interpolation between $x_i$ and $x_j$
    Generate synthetic samples of $z_{ij}$
        $\omega \sim Unif(0,1)$
        $z_{ij} = x_i + \omega \cdot (x_j - x_i)$
        $N^* \leftarrow N^* - 1$
End while
```

D. Borderline-SMOTE Technique

Borderline-SMOTE is also implemented within the over-sampling framework. While standard SMOTE utilizes all nearest neighbors, Borderline-SMOTE specifically targets instances near the decision boundary. This distinction motivated our study, despite differences in implementation.

Given a value of *k*, Borderline-SMOTE categorizes nearest neighbors based on their corresponding label information into three categories: If the number of their labels belonging to the majority is equal to *k*, x_i is regarded as a noisy sample. If it is less than half of *k*, x_i is regarded as a safe sample. Otherwise, x_i is regarded as danger, and assigned to a danger set. (The procedure for generating Danger samples, distinct from SMOTE, is summarized in Algorithm 2.)

Algorithm 2: The pseudo code of Danger

```
danger = {}                         \\* creating Danger(k,i)
For $i = 1, \cdots, N_p$:
    Compute k-nn, and count nn having the negative labels
    (nn_i ← sum(I($y_{nn} = -1$))
    If ($\frac{k}{2}$ ≤ nn_i ≤ $k$) do:
        danger ← danger ∪ {$x\_i$}
    End If
End For
```

Note: y_{nn} indicates the label information of *k* nearest neighbors.

The synthetic samples are generated in the same way to SMOTE. Our study identifies the initial sets for active learning by modifying the Danger procedure.

E. Evaluation Metrics

The most common setting is binary classification within the framework of supervised learning, particularly when addressing class imbalance problems. By convention, the prevalent performance measure is accuracy, defined as Eq. (1), where TP, TN, FP, and FN denote the numbers of true positives, true negatives, false positives, and false negatives, respectively.

$$\text{Accuracy} = \frac{TP+TN}{TP+FN+FP+TN} \tag{1}$$

However, the predictive accuracy does not often convey performance properly in the context of imbalanced data [6, 9, 21]. Area under the receiver operating characteristic curve (AUROC) assigns lower score to random or those that predict only one class [22], making it a more reliable evaluation metric than accuracy. When AUROC curves intersect, the overall AUROC serves as an average value for comparing models [23]. Recently, the area under the precision-recall curve (AUPRC) has garnered considerable attention in settings with class imbalance, according to recent literature [24–27]. Let f be a trained model for a binary classification in an imbalanced context, represented as $f: X \rightarrow Y$, where $(x, y) \in X \times Y$. The model, f outputs continuous probability scores over the space, and both AUROC and AUPRC can be expressed as shown in Eq. (2) and Eq. (3).

$$\text{AUROC}_\text{f} = 1 - E_{z\,\sim \text{f(x)}|y=1}[\text{FPR}(f, z)] \tag{2}$$

$$\text{AUPRC}_\text{f} = 1 - \text{p}_\text{y}[0] \cdot E_{z\,\sim \text{f(x)}|y=1}[\frac{\text{FPR(f,z)}}{P(f(x)>z)}] \tag{3}$$

As noted in [24], these two metrics differ in how they optimize model-dependent parameters. While AUROC minimizes the expected false positive rate (FPR) across all positive samples evenly, AUPRC minimizes the expected FPR over positive samples, weighted by the inverse of the firing rate $P(f(x) > z)$, at a given positive sample score. Consequently, AUPRC places greater emphasis on high-score misclassification errors, so it should be interpreted cautiously in general situations. However, it is particularly well-suited for identifying abnormal patterns in transactional records, since high-score misclassifications can lead to significant potential damages in alert-feedback systems due to the costs incurred by the verification process. Therefore, our study prioritizes the AUPRC score over AUROC and accuracy as reference measures.

IV. Methods

A. Active Learning

AL is a machine learning technique that aims to select informative instances based on a given acquisition function, allowing for some control over the input space. This is especially useful when part of the data is unlabeled within the supervised learning framework. Traditionally, supervised learning assumes balanced data, but few studies have addressed the class imbalance problem in the context of active learning. (In the domain of computer vision, the imbalance problem is typically defined by the imbalance ratio, which is calculated as the mean divided by the standard deviation for each class, particularly in multi-label settings [28]. This definition deviates from ours, so it is not considered in this context.) One study directly tackles the class imbalance problem in active learning within a support vector machine (SVM) framework [29]. This study selects a small, constant number of random sets, independent of the training set size, and chooses the instance closest to the hyperplane, assuming that the instance is among the top p% closest instances in the training set with a probability of 1 - η. If η is set to 0.05, the number of samples in a random set is fixed at 59, regardless of the size of the training set. This approach has an advantage to reduce the version space faster by local searching, but it is specifically designed for SVM. Based on the literature on the performance of data-driven models, SVMs do not perform well on transactional records, so this approach is not preferred as reference in the context.

B. Borderline-Active Learning

The previous literature shows that AL can be advantageous for training on imbalanced data due to its inherent property of not retaining its entire input. In particular, the distribution near the decision boundary is more balanced than that of the entire dataset [29]. Hence, we hypothesize that the performance with AL, focusing solely on informative samples near the decision boundary, would be at least as good as methods that either retain the entire dataset or enforce balance through sampling approaches.

The AL can be implemented under either stream-based or pool-based framework depending on how to choose the instances for learning, and we assumes pool-based framework where the data is denoted as D consisting of labeled observations and unlabeled observations (For simplicity, they are denoted as $\mathcal{L}$ set and $\mathcal{U}$ set, respectively).

A. Initial Setting

Motivated by the Danger procedure in Borderline-SMOTE, we construct a seed set S based on the L set for learning. The key difference in our approach is that S includes both minority and majority samples near the borderline, with approximately balanced quantities, to proceed with AL. In contrast, the danger set in Algorithm 2 only includes minority samples near the borderline.

Let $L_P = \{x_i: i = 1, \cdots, \mathrm{N}_p\}$ and $L_N = \{x_i^*: i = 1, \cdots, \mathrm{N}_n\}$ be a set of minority samples, and majority samples, respectively. The procedure to construct S, is as follows:

Step 1. For $x_i \in L_P$, compute k-nn, and count the negative samples, as $\mathrm{nn_i} \leftarrow \mathrm{sum}(\mathrm{I}(\mathrm{y_{nn}} = -1))$.

Step 2. If $(\frac{k}{2} \le \mathrm{nn_i} \le \mathrm{k})$ do:

$$S \leftarrow \mathrm{S} \cup \{x_i,\ \mathrm{x}_i^{(nn-)}\}$$

where $\mathrm{x}_i^{(nn-)}$ indicates the *nn* with negative labels

Step 3. Repeat for all x_i in L_p

If k is set to 5, the number of negative samples typically becomes about twice as large as the number of positive samples, at most. This ratio can be adjusted by controlling both k and the criteria for the sets. (For example, if we randomly choose a single negative instance, it creates a perfectly balanced seed set.

B. Implementation

As the data-driven model is trained with S, AL can be incorporated by adding the most informative samples, x_i, from the unlabeled set U according to a predefined strategy. One common strategy is uncertainty-based selection [30, 31], where entropy is a well-known measure, as shown in Eq (4).

$$\mathrm{H}(p_j) = -\sum_{j=1}^{n_i} p(x_j) \cdot \log p(x_j) \tag{4}$$

Given the characteristics of entropy, the instance with the highest entropy is selected. Higher entropy indicates greater uncertainty, which naturally aligns with the instances in S in our study. This entropy measure can be re-expressed in Eq (5) where $P_L(y|x)$ indicates the predicted value by the model for x_i.

$$H\big(y_i, P_L(y|x)\big) = -\sum_{y_i} \mathrm{y_i} \cdot \log P_L(y_i|x_i) \tag{5}$$

This measure is often incorporated into more refined formulas, but its fundamental characteristic is inherently maintained. Based on this measure, active learning (AL) can be implemented as follows (the pseudocode follows).

Algorithm 3: The pseudo code of Borderline-AL

```
Algorithm. AL
Input:   $x_i \in U,\ (x_i, y_i) \in S$ as input
Output:  $S'$
Initialization: $S' = S$
Loop while adding new instance into S'
    Train the pre-defined model with $S'$
    Use the model to probabilistically label $x_i$
    Compute Eq (5)
    Choose the instance satisfying
        $\mathrm{x_i} = arg \max_{\{\mathrm{x_i}\}} H\big(y_i, P_L(y|x)\big)$
    $\mathrm{S}' = S' \cup \{x_i\}$
Until the predefined stopping condition is met.
Return $S'$
```

Note: S indicates the seed set.

V. Experiments

A. Datasets

The original dataset consists of 284,807 credit card transactions made by European cardholders in September 2013. The data were collected through a collaboration between Worldline and the Machine Learning Group at Université Libre de Bruxelles [32]. Out of the 26 features, 24 were transformed using principal component analysis (PCA) to maintain confidentiality. The two features that were not transformed are 'time' and `amount.' 'Time' represents the number of seconds elapsed from the first transaction to each subsequent transaction, while `amount' denotes the total amount paid in each transaction. Each transactional record belongs to either `normal' or `abnormal'.

B. Exploration & Pre-processing

Before classification, the 'amount' feature was log-transformed to prevent performance deterioration due to skewness. Additionally, the density kernels of all features were analyzed by class, and some of these results are randomly selected and displayed in Fig. 3.

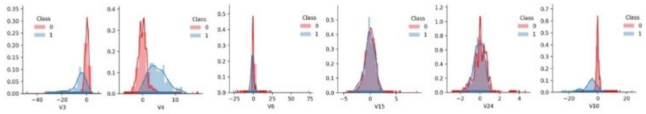


Fig. 3. Kernel densities of randomly selected input features.

C. Experiment: Implementation

SMOTE and Borderline-SMOTE are applied with 100% synthesis based on the number of positive instances, creating a perfectly balanced dataset with a 1:1 ratio of positive to negative instances. For AL, all negative samples from the danger set are added to the seed set, *S* for initial learning, resulting in an approximately 1:2 positive-to-negative ratio. Here, *k* is fixed at 5, and the predefined stopping condition is set such that the difference in accuracy between iterations is less than 0.0001. This value is set as a stopping criterion, with the assumption that increments smaller than 0.0001 are negligible, particularly in terms of training efficiency.

(As mentioned in the previous section, balance can be achieved by selecting a nearest negative neighbor in AL, though we believe this does not significantly impact performance given the characteristics of AL.)

Under this setup, all 24 features are used as input, and the experiment is conducted independently for each condition.

VI. Results

The performance of AL is evaluated based on accuracy, AUROC, and AUPRC for each model, with the selected models demonstrating relatively better performance on transactional records compared to combinations of other models in previous literature [11-15]. The AL approach achieved the highest accuracy in 4 out of 5 models except NN and the best AUPRC in 3 out of 5 models. Overall, AL demonstrated fair performance; however, it exhibited lower accuracy compared to SMOTE in the NN model. The reasons for the NN's performance deviation are not entirely clear, but several factors, such as the number of layers or nodes, could influence its performance. These factors should be explored further in future studies.

In contrast, SMOTE delivered the best AUROC in 4 out of 5 models, while Borderline-SMOTE showed similar results to SMOTE, but with the best AUROC in only 2 out of 5 models, which is relatively lower (The comparison results are summarized in Table 1).

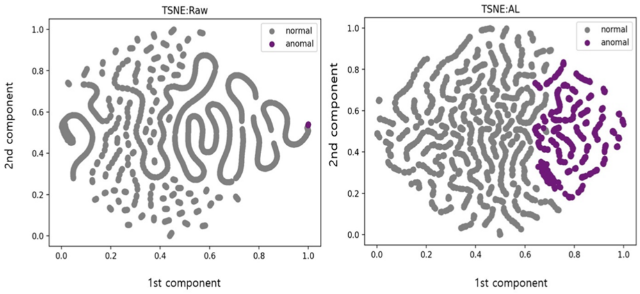
Table 1. Performance comparison by the different processing

Models	Processing	Performance		
		ACCU.	AUROC	AUPRC
LR	SMOTE	0.977	0.958*	0.845*
	Borderline-SMOTE	0.992	0.935	0.841
	Active learning	0.997*	0.925	0.715
RF	SMOTE	0.997	0.964	0.818
	Borderline-SMOTE	0.990	0.966*	0.806
	Active Learning	0.999*	0.939	0.836*
XGboost	SMOTE	0.987	0.959*	0.861
	Borderline-SMOTE	0.997	0.959*	0.877
	Active Learning	0.999*	0.952	0.878*
L-GBM	SMOTE	0.980	0.966*	0.840
	Borderline-SMOTE	0.989	0.962	0.819
	Active Learning	0.999*	0.919	0.858*
Neural Networks	SMOTE	0.999*	0.923*	0.885
	Borderline-SMOTE	0.999*	0.920	0.896*
	Active Learning	0.995	0.914	0.838

[1] LR: Logistic regression (base).; * indicates the best accuracy

A. Additional analysis results

The t-SNE (Stochastic Neighbor Embedding) [33] is illustrated in Fig. 4, comparing SMOTE and AL after training the NN classifier where the X and Y axes represent the first and second t-SNE components, respectively. As shown in Fig. 4, the synthetic samples from SMOTE are less distinguishable, as the total number of instances nearly doubles in the t-SNE plot. In contrast, AL exhibits more distinct clustering, making it easier to identify positive samples, with the total number of instances averaging below 20k.



The x and y axes indicate the first and the second t-SNE components.
Fig. 4. t-SNE comparison between SMOTE vs. AL

Additionally, we identified the important features selected by the NN classifier using AL based on SHAP values described in Eq (6). The SHAP value originally represents a fair distribution of a total reward among participants based on their contributions, and it can be interpreted as the contribution of each feature to the prediction by a given model in machine learning context. According to results, we illustrated the top 10 features as a descending order based on SHAP value in Fig. 5. The 4th, 10th, 14th, and 16th features were prioritized according to Eq. (6), though no single feature emerged as distinctly dominant.

$$\phi_i(\upsilon) = \sum_{S\subseteq N\{i\}} \frac{|S|!(|N|-|S|-1)!}{|N|!}[\upsilon(S\cup\{i\})-\upsilon(S)] \quad (6)$$

where $\phi_i(\upsilon)$ represents the SHAP value for each feature i, and N is the set of all features. Here, $\upsilon(S\cup\{i\})-\upsilon(S)$ is marginal contribution of feature i to the coalition S when |S| and |N| are the size of the subset S and the total number of features, respectively.

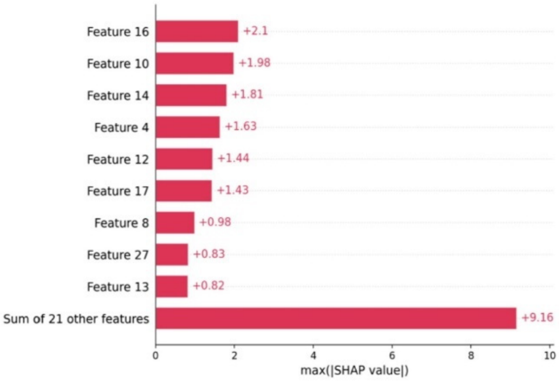


Fig. 5. Features by SHAP values.

VII. Discussion

It is well-known that sampling techniques are commonly used to address the class imbalance problem, yet few studies have explored the use of Active Learning (AL). Oversampling techniques are typically applied before training the model, requiring minimal time. In contrast, AL is more computationally expensive, as it is performed during the model's training process. However, AL offers distinct advantages over methods like oversampling and undersampling. The oversampling techniques generally encounter computational burdens for training because of excessive data volume while undersampling techniques are exposed to the risk of losing valuable information at the cost of balance. Since AL focuses on informative instances for training according to a pre-defined informative measure, it can avoid those risks. However, AL does not originally aim to make balance unlike sampling techniques, we leverage the fact that the instances near decision boundary are relatively balanced in class distribution, which drives us to propose Borderline-AL approach. Based on our approach, initial set of instances are collected at the boundary, which reduce skewness in distribution and is faster than standard AL techniques. In alert-feedback systems, for example, positive instances in transactional records are often less than 1-2% of the total data, which can lead to high computational costs. Although over-sampling can help by generating synthetic instances, it may not always be efficient. Additionally, false positives in alert-feedback systems can result in significant damage. In our experiments, AL techniques outperformed sampling methods, demonstrating higher accuracy and better AUPRC (Area Under the Precision-Recall Curve), even though they do not produce a perfectly balanced dataset. Recent research has also highlighted that synthetic samples generated by methods like SMOTE may not always accurately represent the minority class distribution, potentially harming classification performance. [34] Given these limitations, AL algorithms present a promising alternative to over-sampling techniques for handling class imbalance in transactional records. We also plan to further develop AL techniques to solve the imbalance issue in the near future.

VIII. Conclusion

Unlike sampling techniques, our proposed approach based on AL, focuses solely on informative samples near the decision boundary for effective training. This avoids risks of losing valuable information and doubling the training volume associated with sampling techniques, which also addressing the imbalance issue. Given these facts, it is noteworthy that AL demonstrates its potential for addressing the class imbalance problem, especially when compared to traditional sampling techniques.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

AMK: conceptualization, analysis and writing. KK: conceptualization, review.

Funding

This research was supported by Basic Science Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (RS-2023-00237241).

References

[1] C. Alippi, G. Boracchi, and M. Roveri, "Just-in-time classifiers for recurrent concepts." *IEEE Trans. Neurala. Netw. Learn. Syst.*, vol. 24, pp. 620–634, 2013.

[2] G. Krempl and V. Hofer, "Classification in presence of drift and latency," in *Proc the 11th Data Mining Workshops*, 2011, pp. 596–603. Vancouver, Canada.

[3] K. Leonard, "Detecting credit card fraud using expert systems," *Computers and Industrial Engineering*, vol. 57, pp. 103–106, 1993.

[4] H. He and E. Garcia, "Learning from imbalanced data," *IEEE Transactions on Knowledge and Data Engineering*, vol. 21, pp. 1263–1284, 2009.

[5] M. Pazzani, C. Merz, P. Murphy, K. Ali, T. Hume, and C. Brunk, "Reducing misclassification costs," *Machine Learning Proceedings*, July 10–13, pp. 217–225, 1994.

[6] N. Chawla, K. Bowyer, L. Hall, and W. Kegelmeyer, "SMOTE: synthetic minority over-sampling technique," *Journal of Artificial Intelligence Research*, vol. 16, pp. 321–357, 2002.

[7] H. Han, W. Wang, and B. Mao, "Borderline-smote: A new over-sampling method in imbalanced data sets learning," in *Advances in Intelligent Computing,* D. Huang, X. Zhang, G. Huang, Eds, vol. 3644. Springer, Berlin, Heidelberg, pp. 878–887, Springer, 2005.

[8] C. Rijsbergen, *Information Retrieval*, Butterworths, London, 1979.

[9] M. Kubat and S. Matwin, "Addressing the curse of imbalanced training sets: One-sided selection," in *Proc. the 14th International Conference on Machine Learning*, pp. 179–186, 1997.

[10] I. Tomek, "A generalization of the K-NN rule," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 2, pp. 121–126, 1976.

[11] S. Khatri, A. Arora, and A. Agrawal, "Supervised machine learning algorithms for credit card fraud detection: A comparison," in *Proc. The 10th International Conference on Cloud Computing, Data Science & Engineering*, 2020, pp. 680–683.

[12] L. Duan, "Performance evaluation and practical use of supervised data mining algorithms for credit card approval," in *Proc. International Conference on Computing and Data Science (CDS)*, pp. 251–254, 2020.

[13] X. Yu, X. Li, Y. Dong, and R. Zheng, "Deep neural network algorithm for detecting credit card fraud," in *Proc. International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE)*, 2009, pp. 181–183.

[14] T. Lin, P. Goyal, R. Girshick *et al.*, "Focal loss for dense object detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, 318–327, 2020.

[15] R. Carrasco and M. Sicilia-Urban, 2009, "Evaluation of deep neural networks for reduction of credit card fraud alerts," *IEEE Access*, vol. 8, pp. 186421–186432.

[16] R. Bhuiyan, M. Khatun, M. Taslim, and M. Hossain, "Handling Class Imbalance in Credit Card Fraud Using Various Sampling Techniques," *Am. J. Multidisciplinary. Res. Inno*, vol. 1, pp. 160–168, 2022.

[17] L. Breiman, "Random forests," *Machine Learning*, vol. 45, pp. 5–32, 2001. https://doi.org/10.1023/A:1010933404324

[18] G. Ke, Q. Meng, T. Finley *et al.*, "LightGBM: A highly efficient gradient boosting decision tree," *Advances in Neural Information Processing Systems*, vol. 30, pp. 3146–3154, 2017.

[19] T. Chen and C. Guestrin "XGboost: A scalable tree boosting system," in *Proc. the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 785–794. New York, NY, USA: ACM, 2016.

[20] D. Rumelhart, G. Hinton, and R. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, p. 6088, 533–536, 1986. https://doi.org/10.1038/323533a0

[21] C. Liang and C. Li, "Data mining for direct marketing: Problems and solutions," in *Proc. the Fourth International Conference on Knowledge Discovery and Data Mining (KDD-98)*, New York, NY, AAAI Press, 1998.

[22] A. Bradley, "The use of the area under the ROC curves in the evaluation of machine learning algorithms," *Pattern Recognition*, vol. 30, no. 6, pp. 1145–1159, 1997.

[23] S. Lee, "Noisy replication in skewed binary classification," *Computational Statistics and Data Analysis*, vol. 34, pp. 165–191, 2000.

[24] M. McDermott, L. Hansen, H. Zhang, G. Angelotti, J. Gallifant, "A Closer Look at AUROC and AUPRC under Class Imbalance, *arXvid: 2401.06091v3*, pp. 1–32, 2024.

[25] D. Rosenberg, *Imbalanced Data? Stop Using ROC-AUC and Use AUPRC Instead*, June 2022.

[26] T. Satio and M. Rehmsmeier, "The precision-recall plot is more informative than the roc plot when evaluating binary classifiers on imbalanced datasets," *PloS One*, vol. 10, e0118432, 2015.

[27] P. Flach and M. Kull, "Precision-recall-gain-curves: Pr analysis done right," *Advances in Neural Information Processing Systems*, vol. 28, 2015.

[28] U. Aggarwal, A. Popescu, and C. Hudelot, "Minority class oriented active learning for imbalanced datasets," arXiv:2202.00390v1, pp. 1–11, 2022.

[29] S. Ertekin, J. Huang, L. Bottou, and C. Giles, "Learning on the border: Active learning in imbalanced data classification," *CIKM*, Lisbon, Portugal, pp. 1–10, 2007.

[30] C. Zhang and T. Chen, "An active learning framework for content-based information retrieval," *IEEE Transactions on Multimedia*, vol. 4, no. 2, pp. 260–268, 2002.

[31] J. Zhu and E. Hovy, "Active learning for word sense disambiguation with methods for addressing the class imbalance problem," in *Proc. Joint. Conf. Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pp. 783–790, 2007.

[32] C. Fabrizio, L. Yann-Ael, C. Olivier *et al.*, "Combining Unsupervised and Supervised Learning in Credit Card Fraud Detection," *Information Sciences*, vol. 4, pp. 317–331, 2021.

[33] G. Hinton and S. Roweis, "Stochastic neighbor embedding," *Advances in Neural Information Processing Systems*, vol. 15, pp. 833–840, 2002.

[34] D. Elreedy, A. Atiya, and F. Kamalov, "A theoretical distribution analysis of synthetic minority oversampling technique (SMOTE) for imbalanced learning," *Machine Learning*, pp. 1–21, 2023.