BAREKENG: Journal of Mathematics and Its Applications
December 2026 Volume 20 Issue 4 Page 2839-2850
P-ISSN: 1978-7227 E-ISSN: 2615-3017
https://doi.org/10.30598/barekengvol20iss4pp2839-2850

INTEGRATING STRUCTURED AND UNSTRUCTURED FEATURES FOR E-TICKETING CLASSIFICATION: A MACHINE LEARNING AND ENSEMBLE-BASED APPROACH

Siti Zulaikha Mohd Jamaludin[1], Majid Khan Majahar Ali[2]*, Eric Wong Vun Shiung[3], Mohd Tahir Ismail[4], Noor Farizah Ibrahim[5], Nur Ezlin Zamri[6]

[1,2,3,4]*School of Mathematical Sciences, Universiti Sains Malaysia*
[5]*School of Computer Sciences, Universiti Sains Malaysia*
USM, Penang, 11800, Malaysia

[3]*Keysight Technologies, Bayan Lepas Free Industrial Zone Phase 3*
Bayan Lepas, Penang, 11900, Malaysia

[6]*Department of Mathematics and Statistics, Faculty of Science, Universiti Putra Malaysia*
UPM, Serdang, Selangor, 43400, Malaysia

*Corresponding author's e-mail: * **majidkhanmajaharali@usm.my***

Article Info

Article History:

Received: 10th September 2025
Revised: 02nd April 2026
Accepted: 22nd May 2026
Available online: 01st July 2026

Keywords:

Classification;
Ensemble;
Feature selection;
Hybrid data;
Machine learning;
Ticketing system.

ABSTRACT

E-ticketing systems (ETS) generate mixed information from categorical ticket attributes and free-text descriptions, yet many classification studies still treat these sources separately, which can limit routing accuracy. This study aims (i) to develop an effective preprocessing pipeline for hybrid data, (ii) to develop a two-stage feature selection (2-FS) pipeline for hybrid data, (iii) to design an ensemble classification framework that improves hybrid data performance, and (iv) to benchmark all implemented and proposed models using real-world ETS data. The methodology builds a unified preprocessing workflow by combining Natural Language Processing for textual feature with categorical encoding for categorical features, followed by feature concatenation and vector-space integration to form hybrid representations. Five baseline classifiers (LR, SVM, MNB, RF, and KNN) are evaluated and extended with majority-vote ensembles that pair LR with other classifiers (LR-S, LR-M, LR-R, LR-K, and LR-ALL). Model performance is assessed using accuracy, precision, recall, and F1-score, and the Friedman test is applied to examine statistical consistency across datasets. Results show that hybrid data consistently outperforms single-type datasets, while categorical-only data yields the lowest scores. The best hybrid ensemble of LR-K achieves up to 93% classification accuracy, with strong overall performance also observed for RF on hybrid data. The Friedman test indicates minimal rank differences across models, suggesting that several classifiers are competitive under this ETS setting. Limitations include possible noise and overfitting effects that reduce separation between dataset types. Future work will explore feature ranking-based selection, data-driven segmentation during preprocessing, multi-level classification, imbalance handling, and deeper ensemble strategies to strengthen robustness and generalization.

How to cite this article:

S. Z. M. Jamaludin, M. K. M. Ali, E. W. V. Shiung, M. T. Ismail, N. F. Ibrahim and N. E. Zamri., "INTEGRATING STRUCTURED AND UNSTRUCTURED FEATURES FOR E-TICKETING CLASSIFICATION: A MACHINE LEARNING AND ENSEMBLE-BASED APPROACH", *BAREKENG: J. Math. & App.*, vol. 20, no. 4, pp. 2839-2852, Dec, 2026.

Journal homepage: *https://ojs3.unpatti.ac.id/index.php/barekeng/*
Journal e-mail: *barekeng.math@yahoo.com; barekeng.journal@mail.unpatti.ac.id*

Research Article · Open Access

1. INTRODUCTION

The rapid digitization of service management has led to the global adoption of e-ticketing systems (ETS) in domains such as IT support, public services, telecommunications, and customer care. These systems are crucial for automating the process of creating, classifying, and resolving service tickets, which enhances response times, accountability, and efficiency [1], [2], [3]. However, growing volume and complexity of ticket data, which frequently consists of both structured and unstructured features, make challenge to classify ETS accurately and effectively [4]. Due to this matter, a common operational issue is that IT experts may still incorrectly assign tickets to the incorrect resolver group [5].

In automating ticket classification, machine learning (ML) techniques have demonstrated significant success, providing scalable solutions that outperform human or rule-based approaches [6], [7], [8]. Nonetheless, most previous studies train classification models using only structured features [8] or only unstructured text [9], [10], which underutilizes the complementary information available in hybrid data. Although a minority of works combine both structured and unstructured features for classification [11], [12], [13], they often do not describe clearly how hybrid data are constructed and managed across preprocessing and classification. In another word, preprocessing workflows tailored to hybrid data remain limited and are not well documented. This motivates the need for an effective hybrid data preprocessing technique, specifically to support reliable classification in complex ETS settings. Noted that hybrid data in this context is referring to the data that joins and integrates the structured and unstructured features.

In comparison with [11], [12], [13], [14], [15], the novelty of this work is a hybrid data-first ETS framework that links a two-stage feature selection (2-FS), a unified preprocessing pipeline, and majority-vote ensemble modelling into a repeatable methodology. Prior studies either address trouble-ticket classification in domain-specific settings without a unified hybrid preprocessing workflow [11], [15], focus on classification strategies outside the ETS hybrid-routing context [12], [13], or apply ensembles for ticket identification without integrating staged hybrid feature selection with unified preprocessing [14]. In this study, 2-FS first screens and validates hybrid predictors through exploratory analysis and visualization, then refines a compact hybrid subset for modelling. In parallel, the unified pipeline standardizes text processing [16] and categorical encoding before merging both feature types into a single vector space for consistent training and evaluation across baseline models. In addition, this work implements a majority-vote technique to ensemble the LR-centered models [7], [14], which supports robust decision-making in the final classification.

From a mathematical standpoint, this work offers an opportunity to develop a unified modeling framework that combines feature selection, classification, and data preprocessing under a hybrid data perspective. One of the main challenges is developing efficient feature engineering methods that maintain discriminating power while reducing dimensionality [17], particularly when working with diverse data sources [11], [17] . Hence, this motivates the development of strong ensemble models that can combine the advantages of several classifiers in a way that is both interpretable and mathematically sound. Accordingly, this study addresses the following research question, how can a unified hybrid data pipeline that integrates preprocessing, staged feature selection, and voting-based ensemble modelling improve ETS ticket classification performance on real-world data? Therefore, the objectives of this work are: (1) to develop effective feature selection pipeline for hybrid data, (2) to develop effective preprocessing pipeline for hybrid data, (3) to design an ensemble classification framework for hybrid data to improve performance and (4) to evaluate the performance of implemented and proposed models with benchmarking on e-Ticketing System data.

The rest structure of this work is as follows: Section 2 outlines the methodology including data presentation, model classification, and evaluation metrics. Section 3 describes experimental setup and results for our proposed method that is used to achieve our objectives. Also, this section offers a critical discussion on our results and findings. Finally, Section 4 provides a conclusion and recommendation for future work.

2. RESEARCH METHODS

This section covers the research method that is implemented in our work. It is good to note that our work will approach the real-world e-ticketing system (ETS) data, which will be detailed in Section 2.1. The

illustration of proposed methodology is illustrated in Fig. 1, which will be explained in detail in Section 2.2 until 2.4.

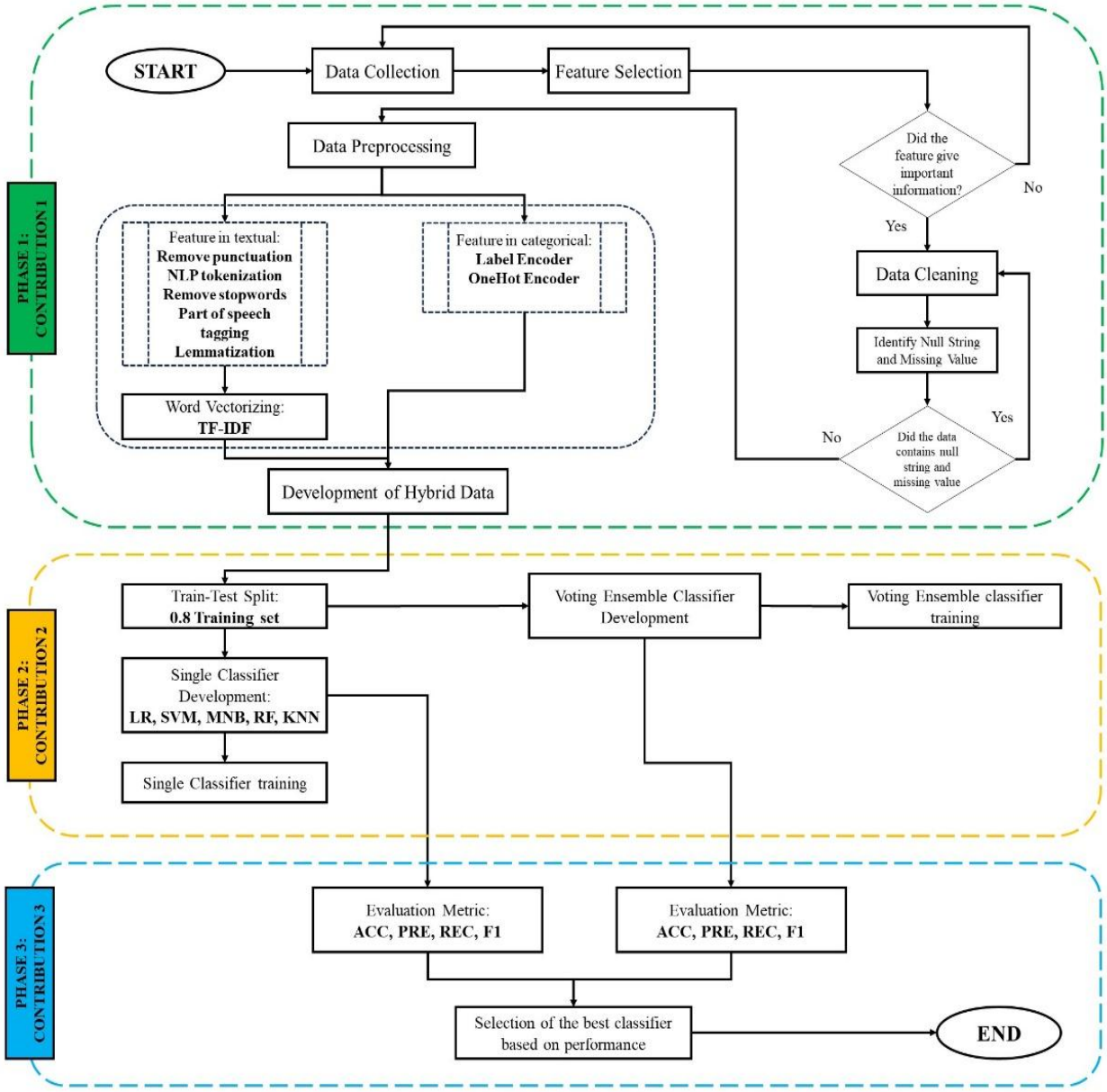


Figure 1. Illustration of the Proposed Methodology

2.1 Data Research

Our work will focus on the field of e-ticketing service desk system in a worldwide hardware industry company located at North Peninsular in Malaysia. The dataset is provided by a company, and we limited our selection to the methods tested on tasking data. The dataset contained both types of data, which are categorical and textual. Initially, the dataset consists of 37532 records with 12 features in total that are generated from customer reports from which the tickets are automatically generated. The summary of features of raw data with their respective description as shown in the following Table 1. The final data that contains all relevant features will be further discussed in Section 3. To ensure the consistency and fairness of model performance, we consider the features in dataset must have both types of structured and unstructured.

Table 1. Raw Data and Description Each Feature

No.	Code	Features	Description	Type of Features
1	*Description*	sct_short_description	Description of the technical issue stated by the customer.	Unstructured
2	*Configuration*	sct_cmdb_ci	Known as Configuration Item, that specify issue.	Structured
3	*Priority*	sct_priority	Urgency level based on user preference	Structured

No.	Code	Features	Description	Type of Features
4	*State*	sct_state	Current state of ticket	Structured
5	*Escalation*	sct_escalation	The type of current escalation of issue.	Structured
6	*Issue*	mi_definition	The definition of current issue.	Structured
7	*Group*	sct_assignment_group	The IT team who resolved this issue/ticket.	Structured

2.2 Phase 1: Development of Hybrid Data

In this sub-section, we will discuss in depth how hybrid data is developed such as feature selection and data preprocessing steps towards textual and categorical features. **Algorithm 1** described the algorithm process for our hybrid data preprocessing. To better classification performance, we approach the two-stage feature selection (2-FS) as an effective feature selection (FS) for hybrid data.

Algorithm 1. Development of hybrid data

Input: TD: the set of data; F_i: the list of features in TD, where $(1 \leq i \leq I)$; I: the total number of features.

1 **while** $(\boldsymbol{n(Q) = 1})$ **do**
2 **for each** $\boldsymbol{i = 1, \dots, I}$ **do**
3 Identify the type of each F_i;
4 Explore the information in each F_i;
5 Perform data visualization in each F_i;
6 Identify which the features are most relevant;
7 Remove the irrelevant features;
8 **end for**
9 Remove the null strings and missing values in each relevant F_i;
10 Define FC_j as categorical-string feature, where $(1 \leq j \leq J)$ and FT_k as textual-string feature, where$(1 \leq k \leq K)$;
11 **for** $\boldsymbol{j = 1, \dots, J}$ **do**
12 **if (**$\boldsymbol{FC}$ **is ordinal)**
13 Apply LabelEncoder to the feature;
14 **else**
15 Apply OneHotEncoder to the feature;
16 **end if**
17 **end for**
18 Initialize empty list FT;
19 **for** $\boldsymbol{k = 1, \dots, K}$ **do**
20 Remove punctuation from the document;
21 Tokenize the document into words;
22 Remove stop words from the tokens;
23 Perform part-of-speech tagging on the tokens;
24 Perform lemmatization on the tokens based on their part-of-speech tags;
25 Add processed tokens to FT;
26 Vectorize FT using TF-IDF;
27 **end for**
28 Concatenate FC and FT to form final data that define as new TD that called as TR;
29 **end while**
30 **return**

Output: TR: The set of data that contains relevant features.

In Lines 1-7, we first inject the raw data into Microsoft 365 Excel to view the dataset accordingly in the form of rows (cases) and columns (features). During this stage, we explored the specific details and identify the type of each feature, (F). Next stage, we performed visualization analysis to verify that the features is relevant or not to proceed perform classification. The irrelevant features will be removed, while the relevant features will remain. The final relevant data will be injected into Python programming to perform further data preprocessing. In Line 9-10, we perform data cleaning to remove the null strings and missing values after we finalize the relevant features. Before we proceed to the next step, we make sure to define the FC as categorical-string feature and FT as textual-string feature.

In Lines 11-17, an encoding technique is applied to preprocess FC. We use a Label Encoder to assign an integer value to each category when FC is ordinal. Else, we will apply for One-Hot Encoder. To preprocessing the FT, we implement NLP because ticket texts are often short, noisy, and domain-specific, which can degrade classification if the text is not normalized [14], [15]. The NLP steps include removing

punctuation, tokenizing words, removing stop words, performing part-of-speech tagging and lemmatization, and vectorizing using TF-IDF, as shown in Lines 18-27. TF-IDF was selected because it down-weights common terms and better emphasizes discriminative words in short ticket descriptions [14], [15]. Line 28 then shows that we integrate textual and categorical features using concatenation to form hybrid data before proceeding with classification.

2.3 Phase 2: Development of Ensemble Classification Model

In purpose to classify the hybrid data, we will implement the common five ML classification models, which are Logistic Regression (LR) [8], Support Vector Machine (SVM) [18], Multinomial Naïve Bayes (MNB) [14] , Random Forest [19], and K Nearest Neighbors (KNN) [20]. The selection of these implemented model is based on most related literature of this work. Afterwards, we perform ensemble classification model by embedded baseline ML models, which is LR, with SVM, MNB, RF, and KNN, respectively.

2.3.1 Fundamental Theory of Single Classification

According to [8], [20], LR is one of the most widely used classification methods in learning analytics. In addition, [8] mentioned that LR is a common technique for classification problems, especially for binary classification. Based on this, LR is selected as the baseline classifier in this work. However, there are several limitations of difficulty interpreting LR when dealing with complex and imbalanced data [21] such as linearity assumption, lack of automatic feature selection and limited predictive power. Therefore, to improve classification performance for ETS tasks, LR is combined with other machine learning models through ensemble modelling to enhance efficiency in ETS support processes.

Incorporating the SVM model into a ticketing system proves beneficial for efficient issue resolution [18]. In this approaching, we trained each class. After that, we treat that class as the positive class and all other classes as the negative class. In the context of this work, we used SVM to classifying tickets effectively based on their categories. Despite its strengths, SVM can face difficulties when handling very large datasets. It is also sensitive to class imbalance, which may lead to biased predictions if not addressed properly. In context of our work, MNB is beneficial for efficient issue categorization in large datasets and easy to implement [22]. According to [9], MNB performs well when handling text data, which makes it suitable for classifying and routing tickets based on their content. MNB is a probabilistic classifier grounded in Bayes' theorem. However, strong independence assumption may limit its predictive accuracy in some practical cases, especially when feature dependencies are important.

According to [19], RF can be defined as a tree-based system for classification that consists of many decision trees. They also mentioned that RF can handle both categorical and numerical features. In ETS classification, RF can provide robust performance by reducing overfitting through averaging multiple decision trees, and it can also offer insights into relevant predictors and handle missing values effectively [23]. However, [23] mentioned that RF might be less interpretable compared to simpler models like LR and requires significant computational resources for large datasets. KNN is a non-parametric method that classifies a sample based on the majority class among its nearest neighbours [19], [20]. In ETS, KNN can support ticket categorization by comparing an incoming ticket with previously resolved tickets and assigning a label based on similarity. This makes KNN practical for problems where similar issues tend to share similar categories. However, KNN is sensitive to irrelevant or redundant features. Therefore, feature selection and careful preprocessing are important to improve its reliability and classification accuracy.

2.3.2 Ensemble Classification Model

In our context, we adopt LR as base classifier on the ETS classifier. This is due to its interpretability, cheap computing complexity, and good generalization ability [24], [25], [26]. According to [25], LR is an appropriate and reliable option for acting as a fundamental model in ensemble classification as it performs well in high-dimensional and sparse datasets, such those including text-based ticket descriptions. Additionally, the probabilistic outputs of LR offer a solid foundation for incorporation into ensemble procedures like voting.

Due to this concern, we embed the LR base learner in a heterogeneous ensemble that also consists of SVM, MNB, RF, and KNN as there is no single model that can fully represent the data. The approach for ensemble classification can be summarized in Table 2. At the first stage, we study the limitations of LR and other implemented single ML model analyzing the complex data in ETS. To combine the baseline model

with each single ML model in this study to obtain the final output, we apply voting ensemble techniques that are inspired by the studies of [7], [18]. We consider an ensemble approach by employing a voting strategy, which selects the alternative that obtains the most votes [18].

Table 2. List of Ensemble Models by Voting Strategy

Code	Ensemble Model
LR-S	LR, SVM
LR-M	LR, MNB
LR-R	LR, RF
LR-K	LR, KNN
LRALL	LR, SVM, MNB, RF, KNN

2.4 Phase 3: Evaluation Comparison

We evaluate the performance of the data classification models using standard classification metrics **[5], [13]**: ACC and $F1$. The ACC metric is used to measure the proportion of correctly classified tickets across all categories. ACC can be expressed as the ratio of the sum of true positives for all categories to the total number of tickets. The $F1$ metric is the harmonic means of precision and recall metrics. The metrics are formulated as in Eq. (1) and Eq. (2), where TP is true positive, TN is true negative, FP is false positive, and $FN$is false negative.

$$ACC = \frac{TP + TN}{TP + TN + FP + FN}, \tag{1}$$

$$F1 = \frac{2 \times \frac{TP}{TP + FP} \times \frac{TP}{TP + FN}}{\frac{TP}{TP + FP} + \frac{TP}{TP + FN}}. \tag{2}$$

To compare the influence in the performance, the analysis will divide the data into three categories. The text data (TXTD) will contain only textual features. Then, the categorical data (CATD) will contain only categorical features. Lastly, hybrid data (HYBD) will contain both textual and categorical features. The summary of the data for baseline comparison can be found in Table 3.

Table 3. Category of Data for the Analysis

Code	Category data	Description	Features Involved
TXTD	Textual	Data that contains only textual features	*Description*
CATD	Categorical	Data that contains only categorical features	*Priority*, *State*
HYBD	Hybrid	Data that contains both textual and categorical features	*Description*, *Priority*, *State*

2.5 Experimental Research

The experiment was executed on a single personal computer equipped with an Intel Core i5 10th Generation 1.60GHz and 16.0GB of RAM. We recorded the raw data in Microsoft 365 Excel. The algorithm in this paper was developed using the Python Anaconda distribution within Visual Studio Code version 1.89. We import scikit-learn library in Python programming to perform data preprocessing, model classification and metrics evaluation. In the model training environment, we utilize sklearn.linear_model for LR model, sklearn.svm for SVM model, sklearn.naive_bayes for MNB model, sklearn.ensemble for RF model and voting ensemble model, and sklearn.neighbors for KNN model.

3. RESULTS AND DISCUSSION

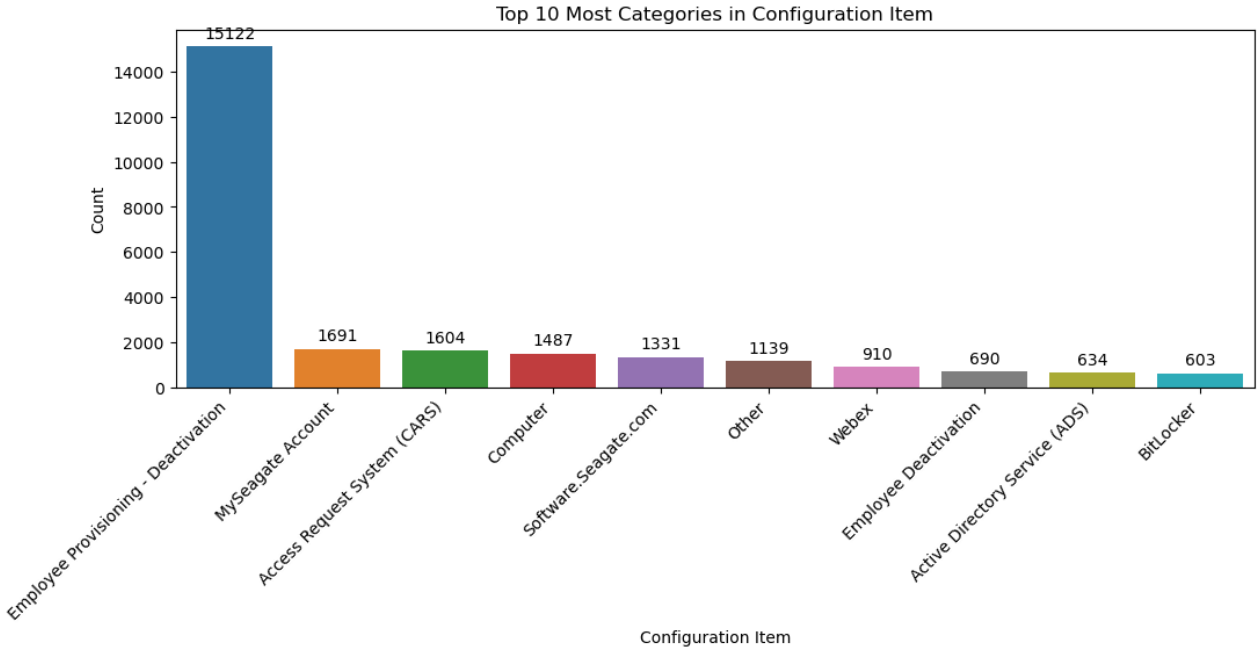
In this section, we will discuss the results of the performance of the five ML models (LR, SVM, MNB, RF, and KNN) and five ensemble ML models (LR-S, LR-M, LR-R, LR-K, and LRALL) to define the optimum classification of the ETS assignment group. Section 3.1 discusses the result of relevant features by performing the feature selection. Section 3.2 discusses the performance analysis of the models in terms of types of data (TXTD, CATD, and HYBD). To ensure consistency and fairness in result analysis and interpretation, we performed experiments with the same device and programming tools. Moreover, this study

using default parameter settings to ensure a fair and consistent baseline comparison across classifiers and datasets. The data were split into training and testing sets using an 80/20 ratio [17].

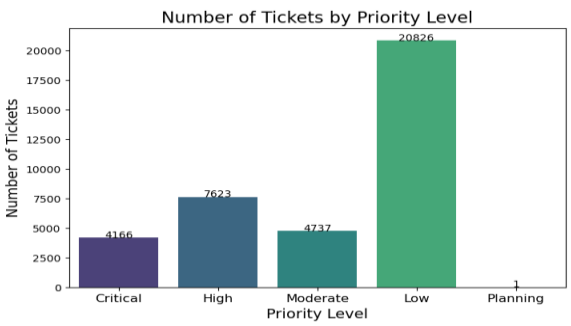
3.1 Analysis for Relevant Features

After we injected the raw data of ETS into Python programming, we performed feature selection by data exploration and visualization. Throughout the data exploration phase, we found that all features did provide significant important information. Thus, there are no elimination features in this phase. Next, we further extended by visualizing technique on all features. To perform data visualization, we split the data into categorical and textual. For textual, we only found that *Description* is the only textual feature among the features. With this, we consider *Description* to be a relevant feature to our ETS data due to the important information that is contained in the feature.

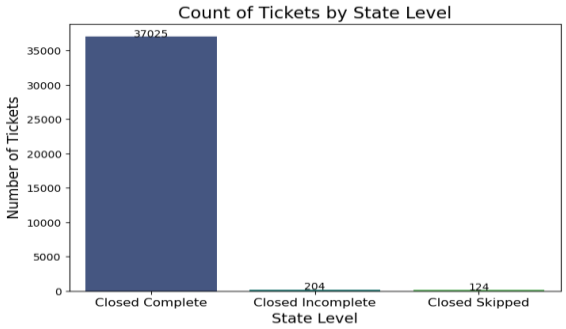
In this work, we perform one-hot encoder to label the non-ordinal type features such as *Configuration Item* and *Group* as shown in Fig. 2 (a) and Fig. 2 (f), respectively. The initial visual of *Priority* in Fig. 2 (b) presented no huge difference. Thus, *Priority* does not need to undergo cleaning process by removing least number. We just performed numerical transformation to proceed further analysis using label encoder. Unfortunately, *Escalation*, and *Issue* as in Fig. 2 (c) – Fig. 2 (e), respectively, showed there is no difference counts and no trend. Hence, we assume these two features are not relevant and are being removed from the data. During the phase of visualization analysis, we finalize only 4 features are relevant to consider our classification problem. The relevant features are *Description*, *Priority*, *Configuration Item*, and *State*. As claimed in study of [18], [24], we found that identifying the most relevant features and eliminating irrelevant can greatly improve model performance. In this work, the implementation of TF–IDF text representation is limited to 10,000 features. After concatenating the encoded categorical features, the final hybrid data vector has 10,023 dimensions.



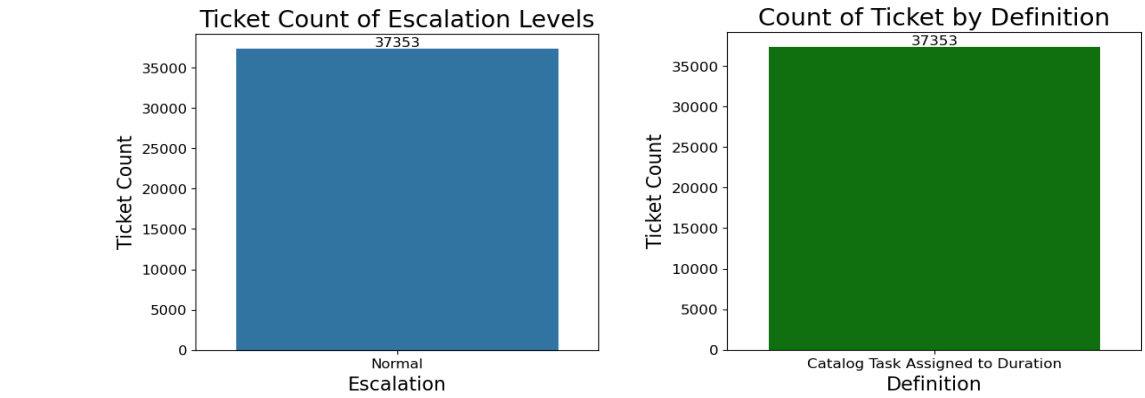
(a)



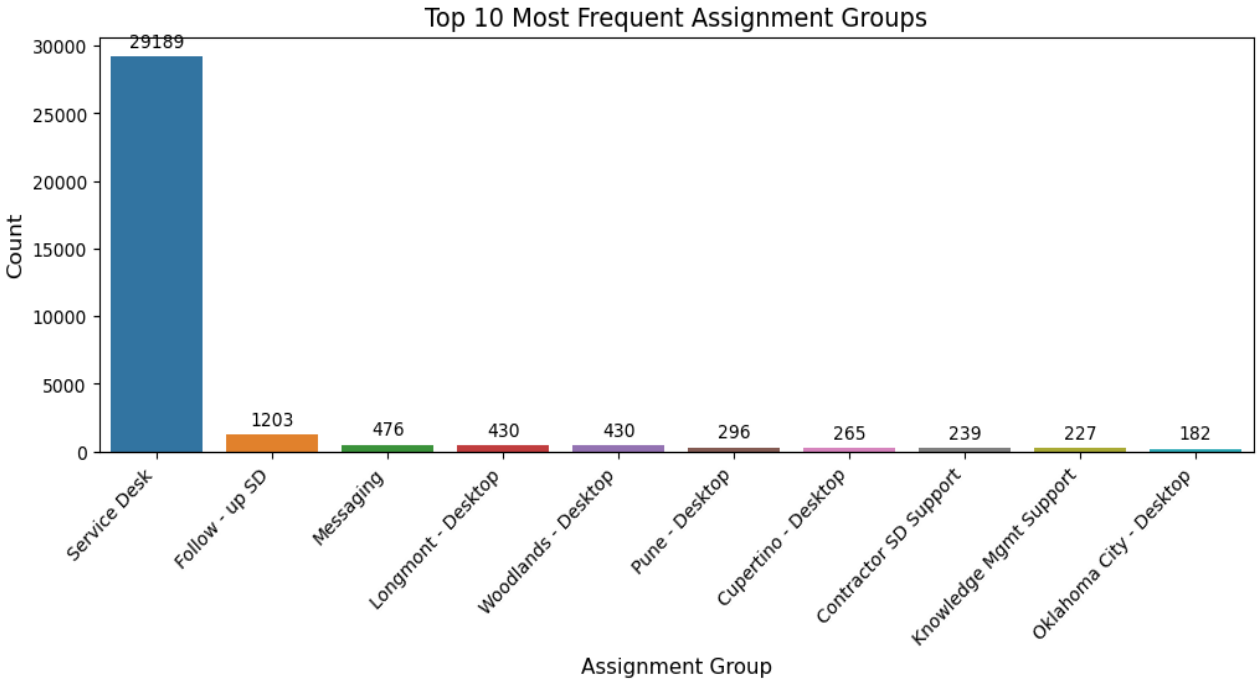
(b)



(c)



(d) (e)



(f)

Figure 2. Initial Visualization
(a) Configuration, (b) Priority, (c) State, (d) Escalation, (e) Issue, (f) Group

3.2 Analysis of Model Classification for Data

The main purpose of this experiment is to analyses the performance of proposed ensemble model with the baseline model leads to better classification. In this subsection, the performance of proposed ensemble classification model will be examined by comparing it with existing ML classification model. All the results performance of ACC and $F1$ in the context of classification models are presented in Table 4 and Table 5, respectively. Note that bold entries in the tables for each metric signifies the best performance of model in terms of data. To test the significance of the model performance towards datasets, this experiment conducts non-parametric statistical test of Friedman test rank (χ^2) [27].

Table 4. Analysis of $\boldsymbol{ACC}$

Dataset	Ensemble Classification					ML Classification				
	LR-S	LR-M	LR-R	LR-K	LR-ALL	SVM	MNB	RF	KNN	LR
HYBD	0.927	0.919	**0.929**	0.918	0.924	0.915	0.883	**0.930**	0.913	0.921
TXTD	0.927	0.919	**0.928**	0.918	0.924	0.927	0.910	**0.928**	0.912	0.924
CATD	**0.868**	**0.868**	0.867	0.830	**0.868**	**0.868**	**0.868**	0.867	0.831	0.868

Table 5. Analysis of $F1$

Dataset	Ensemble Classification					ML Classification				
	LR-S	LR-M	LR-R	LR-K	LR-ALL	SVM	MNB	RF	KNN	LR
HYBD	0.911	0.896	0.846	0.920	0.908	0.917	0.906	0.921	0.913	0.912
TXTD	0.914	0.916	0.889	0.918	0.903	0.917	0.906	0.920	0.913	0.911
CATD	0.807	0.806	0.807	0.807	0.812	0.807	0.807	0.807	0.812	0.807

From Table 4, the analysis in the context of ACC is presented as follows:

1. The score of ACC in all implemented models range from 0.830 to 0.930, indicating that all models are showed strong overall classification performance in the ETS. Thus, this demonstrates the ability of models to accurately assign tickets to the appropriate IT support groups.
2. For ML classification, RF achieved the highest ACC with 0.930 for HYBD. This demonstrates that RF is able to provide robust and reliable performance of ETS classification when implemented the complex data that integrate both textual and categorical data.
3. Ensemble models of LR-R ($ACC = 0.929$) showed the highest performance is close with RF on HYBD. This indicates that ensemble models can efficiently integrate the combination of diverse data types to provide highly accurate IT group classification of ETS.
4. Friedman test rank has been conducted for all the datasets with $\alpha = 0.05$ and degree of freedom, $df = 9$. The p-value for ACC is $> 0.05 (\chi^2 = 14.487)$. Thus, the null hypothesis of equal performance for all models was accepted. Although the value of average rank of all models varied slightly, the differences were very minimal. Therefore, it indicates that the result has no significant statistically different performance among the models.

While the analysis in the context of $F1$ in from Table 5 is presented as follows:

1. The score of $F1$ in all implemented models achieved in range from 0.806 to 0.921. This demonstrates balanced performance of all models in terms both of precision and recall. This advocates that all models are effective at accurately identifying and classifying tickets while minimizing the FP and FN. In view of ETS, this ensures that tickets are correctly assigned and consistently captured across IT groups to solve the tickets.
2. For ML classification, RF achieved the highest $F1 = 0.921$ for HYBD, indicating that RF has a strong ability to balance precision and recall. In the context of ETS, RF is the most preferred ML model that reliable in ETS classification into the correct IT group by minimizing both FP and FN.
3. Ensemble models of LR-K showed the highest performance on HYBD, indicating that LR-K has a strong ability to effectively balance precision and recall. In point of view ETS, this suggests that the LR-K is the best ensemble model that reliable in ETS classification into the correct IT group by ensuring precise and reliable assignment to the correct IT groups.
4. Friedman test rank has been conducted for all the datasets with $\alpha = 0.05$ and degree of freedom, $df = 9$. The p-value for $F1$ is $> 0.05 (\chi^2 = 13.397)$. Thus, the null hypothesis of equal performance for all models was accepted. Although the value of average rank of all models varied slightly, the differences were very minimal. Therefore, it indicates that the result has no significant statistically different performance among the models.

From the overall score, it can be concluded that the CATD dataset that contains only categorical features is in lower performance. Although [28] mentioned that categorical data is practical, it can affect the performance of the model due to often lacking information. Yet, the overall scores showed that baseline model of LR as a simpler model that is able effectively handle the CATD alone without needing more complex models. The overall metric scores of ACC and $F1$ in TXTD and HYBD showed the implemented models are well performing in EST classifications. This indicates that the rich contextual information within textual features provides significant benefits for ETS classifications tasks. This is supported by [15], [29] who highlighted the efficacy of description in the tickets can maximize the model performance due to relevant information they provide. Whilst the TXTD showed higher performance than CATD, still TXTD not able to compete the performance score in HYBD. From the results as shown in Tables 4 and 5, it is considered that the combination of textual and categorical features obtained the best performance scores. In compliance with [11], the hybrid approach by combining the variety of features can increase the information of data. Also, it indirectly improved the accuracy of ETS classification and robustness.

In the context of classification, Tables 4 and 5 showed that ensemble models, especially LR-R and LR-K, achieved high performance across all metrics when using HYBD dataset. In most situations with ETS, the systems usually deal with complex and variety types of data. Ensemble techniques can reduce overfitting in classification, lead to more reliable performance and be able to handle real-world case classification more effectively [29]. Thus, we believe that our ETS classification system can better manage the diversity of incoming tickets, ensuring accurate ticketing classification, routing for prompt resolution and increase user satisfaction by approaching an ensemble technique. Our findings align with [14] by highlighting the effectiveness of an ensemble classification can improve the performance of ETS classifications.

4. CONCLUSION

This study aimed to develop an effective preprocessing pipeline and a two-stage feature selection (2-FS) approach for hybrid ETS data, design LR-centered majority-vote ensemble models, and benchmark their performance on real-world e-ticketing data. To meet these objectives, five ML classifiers (LR, SVM, MNB, RF, and KNN) and five voting ensembles (LR-S, LR-M, LR-R, LR-K, and LR-ALL) were implemented and evaluated across three dataset settings: textual-only (TXTD), categorical-only (CATD), and hybrid data (HYBD). The results show that CATD produces the lowest performance, indicating that categorical features alone often provide limited information for reliable ticket assignment. In contrast, the inclusion of textual descriptions improves performance, and the best results are obtained when textual and categorical features are integrated as HYBD. In particular, the proposed voting ensembles improve performance in the hybrid setting, achieving up to 93% accuracy, which highlights the value of combining complementary feature types for ETS group classification and better resource allocation.

Although HYBD generally performs better than single-type datasets, the overall differences between TXTD, CATD, and HYBD are not large in some cases, which may be influenced by noise or overfitting in the data. Therefore, future work will explore feature ranking methods for feature selection [30] and data-driven segmentation strategies in preprocessing to improve robustness [20]. We also plan to extend the task to multi-level classification [2], address class imbalance [15], further fine-tune the models, and investigate deeper ensemble approaches to enhance generalization [18], [31].

Author Contributions

Siti Zulaikha Mohd Jamaludin: Writing - Original Draft and Project Administration. Majid Khan Majahar Ali: Resources and Funding Acquisition. Eric Wong Vun Shiung: Formal Analysis, Software. Mohd Tahir Ismail: Supervision and Conceptualization. Noor Farizah Ibrahim: Methodology and Writing - Review and Editing. Nur Ezlin Zamri: Visualization and Validation. All authors discussed the results and contributed to the final manuscript.

Funding Statement

This work was supported by a Universiti Sains Malaysia, Short-Term Grant with Project No: R501-LR-RND002-0000001828-0000.

Acknowledgment

The author would like to thank his/her mentor and all helpers for their continued support.

Declarations

The authors declare that has no conflicts of interest in reporting study.

Declaration of Generative AI and AI-assisted Technologies

The authors declare that no generative AI or AI-assisted technologies were used in the preparation of this manuscript, including for writing, editing, data analysis, or the creation of tables and figures.

REFERENCES

[1] S. Heras *et al*., "MULTI-DOMAIN CASE-BASED MODULE FOR CUSTOMER SUPPORT," *Expert Syst. Appl.*, vol. 36, no. 3 PART 2, pp. 6866–6873, 2009, doi: https://doi.org/10.1016/j.eswa.2008.08.003.

[2] A. Zangari, M. Marcuzzo, M. Schiavinato, A. Gasparetto, and A. Albarelli, "TICKET AUTOMATION: AN INSIGHT INTO CURRENT RESEARCH WITH APPLICATIONS TO MULTI-LEVEL CLASSIFICATION SCENARIOS," Sep. 01, 2023, *Elsevier Ltd*. doi: https://doi.org/10.1016/j.eswa.2023.119984.

[3] F. Al-Hawari and H. Barham, "A MACHINE LEARNING BASED HELP DESK SYSTEM FOR IT SERVICE MANAGEMENT," *Journal of King Saud University - Computer and Information Sciences*, vol. 33, no. 6, pp. 702–718, Jul. 2021, doi: https://doi.org/10.1016/j.jksuci.2019.04.001.

[4] M. Gupta, A. Asadullah, S. Padmanabhuni, and A. Serebrenik, "REDUCING USER INPUT REQUESTS TO IMPROVE IT SUPPORT TICKET RESOLUTION PROCESS," *Empir. Softw. Eng.*, vol. 23, no. 3, pp. 1664–1703, Jun. 2018, doi: https://doi.org/10.1007/s10664-017-9532-2.

[5] S. Z. Mohd Jamaludin, M. K. Majahar Ali, E. S. W. Vun, M. T. Ismail, and N. F. Ibrahim, "SOLVING COMPLEXITY DATASET IN E-TICKETING USING MACHINE LEARNING TO DETERMINE OPTIMUM FEATURE," *Malaysian Journal of Fundamental and Applied Sciences*, vol. 19, no. 6, pp. 1068–1081, Dec. 2023, doi: https://doi.org/10.11113/mjfas.v19n6.2921.

[6] C. M. Rump, "MINING A LOTTERY FOR A FAVORABLE JACKPOT[FORMULA PRESENTED]," *Expert Syst. Appl.*, vol. 238, Mar. 2024, doi: https://doi.org/10.1016/j.eswa.2023.122063.

[7] N. A. Azit *et al*., "PREDICTION OF HEPATOCELLULAR CARCINOMA RISK IN PATIENTS WITH TYPE-2 DIABETES USING SUPERVISED MACHINE LEARNING CLASSIFICATION MODEL," *Heliyon*, vol. 8, no. 10, Oct. 2022, doi: https://doi.org/10.1016/j.heliyon.2022.e10772.

[8] M. Azzone, E. Barucci, G. Giuffra Moncayo, and D. Marazzina, "A MACHINE LEARNING MODEL FOR LAPSE PREDICTION IN LIFE INSURANCE CONTRACTS," *Expert Syst. Appl.*, vol. 191, Apr. 2022, doi: https://doi.org/10.1016/j.eswa.2021.116261.

[9] S. Gan, S. Shao, L. Chen, L. Yu, and L. Jiang, "ADAPTING HIDDEN NAIVE BAYES FOR TEXT CLASSIFICATION," *Mathematics*, vol. 9, no. 19, p. 2378, Sep. 2021, doi: https://doi.org/10.3390/math9192378.

[10] M. Qorib, T. Oladunni, M. Denis, E. Ososanya, and P. Cotae, "COVID-19 VACCINE HESITANCY: TEXT MINING, SENTIMENT ANALYSIS AND MACHINE LEARNING ON COVID-19 VACCINATION TWITTER DATASET," *Expert Syst. Appl.*, vol. 212, p. 118715, Feb. 2023, doi: https://doi.org/10.1016/j.eswa.2022.118715.

[11] M. W. Asres *et al*., "SUPPORTING TELECOMMUNICATION ALARM MANAGEMENT SYSTEM WITH TROUBLE TICKET PREDICTION," *IEEE Trans. Industr. Inform.*, vol. 17, no. 2, pp. 1459–1469, Feb. 2021, doi: https://doi.org/10.1109/TII.2020.2996942.

[12] T. Kim and J. S. Lee, "MAXIMIZING AUC TO LEARN WEIGHTED NAIVE BAYES FOR IMBALANCED DATA CLASSIFICATION," *Expert Syst. Appl.*, vol. 217, May 2023, doi: https://doi.org/10.1016/j.eswa.2023.119564.

[13] S. G. Ozkaya *et al*., "AN AUTOMATED EARTHQUAKE CLASSIFICATION MODEL BASED ON A NEW BUTTERFLY PATTERN USING SEISMIC SIGNALS," *Expert Syst. Appl.*, vol. 238, Mar. 2024, doi: https://doi.org/10.1016/j.eswa.2023.122079.

[14] J. Xu, L. Tang, and T. Li, "SYSTEM SITUATION TICKET IDENTIFICATION USING SVMS ENSEMBLE," *Expert Syst. Appl.*, vol. 60, pp. 130–140, Oct. 2016, doi: https://doi.org/10.1016/j.eswa.2016.04.017.

[15] J. Xu, H. Zhang, W. Zhou, R. He, and T. Li, "SIGNATURE BASED TROUBLE TICKET CLASSIFICATION," *Future Generation Computer Systems*, vol. 78, pp. 41–58, Jan. 2018, doi: https://doi.org/10.1016/j.future.2017.07.054.

[16] Q. Li *et al*., "A SURVEY ON TEXT CLASSIFICATION: FROM TRADITIONAL TO DEEP LEARNING," Apr. 01, 2022, *Association for Computing Machinery*. doi: https://doi.org/10.1145/3495162.

[17] M. A. Bouke and A. Abdullah, "AN EMPIRICAL STUDY OF PATTERN LEAKAGE IMPACT DURING DATA PREPROCESSING ON MACHINE LEARNING-BASED INTRUSION DETECTION MODELS RELIABILITY," *Expert Syst. Appl.*, vol. 230, Nov. 2023, doi: https://doi.org/10.1016/j.eswa.2023.120715.

[18] J. He, L. Qu, P. Wang, and Z. Li, "AN OSCILLATORY PARTICLE SWARM OPTIMIZATION FEATURE SELECTION ALGORITHM FOR HYBRID DATA BASED ON MUTUAL INFORMATION ENTROPY," *Appl. Soft Comput.*, vol. 152, Feb. 2024, doi: https://doi.org/10.1016/j.asoc.2024.111261.

[19] S. Das, D. R. Khanwelkar, and J. Maiti, "A SEMI-AUTOMATED CODING SCHEME FOR OCCUPATIONAL INJURY DATA: AN APPROACH USING BAYESIAN DECISION SUPPORT SYSTEM," *Expert Syst. Appl.*, vol. 237, Mar. 2024, doi: https://doi.org/10.1016/j.eswa.2023.121610.

[20] A. K. Biswas, R. Seethalakshmi, P. Mariappan, and D. Bhattacharjee, "AN ENSEMBLE LEARNING MODEL FOR PREDICTING THE INTENTION TO QUIT AMONG EMPLOYEES USING CLASSIFICATION ALGORITHMS," *Decision Analytics Journal*, vol. 9, Dec. 2023, doi: https://doi.org/10.1016/j.dajour.2023.100335.

[21] M. Kiguchi, W. Saeed, and I. Medi, "CHURN PREDICTION IN DIGITAL GAME-BASED LEARNING USING DATA MINING TECHNIQUES: LOGISTIC REGRESSION, DECISION TREE, AND RANDOM FOREST," *Appl. Soft Comput.*, vol. 118, Mar. 2022, doi: https://doi.org/10.1016/j.asoc.2022.108491

[22] R. Slikboer, S. D. Muir, S. S. M. Silva, and D. Meyer, "A SYSTEMATIC REVIEW OF STATISTICAL MODELS AND OUTCOMES OF PREDICTING FATAL AND SERIOUS INJURY CRASHES FROM DRIVER CRASH AND OFFENSE HISTORY DATA," *Syst. Rev.*, vol. 9, no. 1, Sep. 2020, doi: https://doi.org/10.1186/s13643-020-01475-7.

[23] J. Tanha, Y. Abdi, N. Samadi, N. Razzaghi, and M. Asadpour, "BOOSTING METHODS FOR MULTI-CLASS IMBALANCED DATA CLASSIFICATION: AN EXPERIMENTAL REVIEW," *J. Big Data*, vol. 7, no. 1, p. 70, Dec. 2020, doi: https://doi.org/10.1186/s40537-020-00349-y.

[24] M. Phan, A. De Caigny, and K. Coussement, "A DECISION SUPPORT FRAMEWORK TO INCORPORATE TEXTUAL DATA FOR EARLY STUDENT DROPOUT PREDICTION IN HIGHER EDUCATION," *Decis. Support Syst.*, vol. 168, May 2023, doi: https://doi.org/10.1016/j.dss.2023.113940

[25] F. K. Nakano, K. Pliakos, and C. Vens, "DEEP TREE-ENSEMBLES FOR MULTI-OUTPUT PREDICTION," *Pattern Recognit.*, vol. 121, Jan. 2021, doi: https://doi.org/10.1016/j.patcog.2021.108211.

[26] G. Manita, A. Chhabra, and O. Korbaa, "EFFICIENT E-MAIL SPAM FILTERING APPROACH COMBINING LOGISTIC REGRESSION MODEL AND ORTHOGONAL ATOMIC ORBITAL SEARCH ALGORITHM," *Appl. Soft Comput.*, vol. 144, Sep. 2023, doi: https://doi.org/10.1016/j.asoc.2023.110478.

[27] N. E. Zamri *et al.*, "A MODIFIED REVERSE-BASED ANALYSIS LOGIC MINING MODEL WITH WEIGHTED RANDOM 2 SATISFIABILITY LOGIC IN DISCRETE HOPFIELD NEURAL NETWORK AND MULTI-OBJECTIVE TRAINING OF MODIFIED NICHED GENETIC ALGORITHM," *Expert Syst. Appl.*, vol. 240, p. 122307, Apr. 2024, doi: https://doi.org/10.1016/j.eswa.2023.122307.

[28] S. Z. M. Jamaludin *et al*., "NOVEL LOGIC MINING INCORPORATING LOG LINEAR APPROACH," *Journal of King Saud University - Computer and Information Sciences*, vol. 34, no. 10, pp. 9011–9027, Nov. 2022, doi: https://doi.org/10.1016/j.jksuci.2022.08.026.

[29] P. Supsermpol, V. N. Huynh, S. Thajchayapong, and N. Chiadamrong, "PREDICTING FINANCIAL PERFORMANCE FOR LISTED COMPANIES IN THAILAND DURING THE TRANSITION PERIOD: A CLASS-BASED APPROACH USING LOGISTIC REGRESSION AND RANDOM FOREST ALGORITHM," *Journal of Open Innovation: Technology, Market, and Complexity*, vol. 9, no. 3, Sep. 2023, doi: https://doi.org/10.1016/j.joitmc.2023.100130.

[30] H. Chen *et al*., "TEXTCNN-BASED ENSEMBLE LEARNING MODEL FOR JAPANESE TEXT MULTI-CLASSIFICATION," *Computers and Electrical Engineering*, vol. 109, Aug. 2023, doi: https://doi.org/10.1016/j.compeleceng.2023.108751.

[31] A. Rehman, K. Javed, H. A. Babri, and N. Asim, "SELECTION OF THE MOST RELEVANT TERMS BASED ON A MAX-MIN RATIO METRIC FOR TEXT CLASSIFICATION," *Expert Syst. Appl.*, vol. 114, pp. 78–96, Dec. 2018, doi: https://doi.org/10.1016/j.eswa.2018.07.028.