Research Article

AUTOMATIC CLASSIFICATION OF BANKING BRANCH REQUESTS AND ERRORS WITH NATURAL LANGUAGE PROCESSING AND MACHINE LEARNING

Authors: Evren KARAKOÇ, Metin TURAN

To cite to this article: Karakoç, E. & Turan, M. (2025). Automatic Classification Of Banking Branch Requests And Errors With Natural Language Processing And Machine Learning, International Journal of Engineering and Innovative Research, 7(1), p 1-15.

DOI: 10.47933/ijeir.1597039

To link to this artice: https://dergipark.org.tr/tr/pub/ijeir/archive

International Journal of Engineering and Innovative Research

http://dergipark.gov.tr/ijeir

AUTOMATIC CLASSIFICATION OF BANKING BRANCH REQUESTS AND ERRORS WITH NATURAL LANGUAGE PROCESSING AND MACHINE LEARNING

Evren KARAKOÇ*, Metin TURAN[2]

[1] İstanbul Ticaret University, Department of Computer Engineering, Maltepe, Türkiye.
[2] İstanbul Ticaret University, Department of Software Engineering, Maltepe, Türkiye

*Corresponding Author: evren.karakoc@istanbulticaret.edu.tr

(**Received:** 05.12.2024; **Accepted:** 19.01.2025)

https://doi.org/10.47933/ijeir.1597039

ABSTRACT: In today's world, the service sector is undergoing rapid technological development. Keeping pace with this transformation is not only a necessity for institutions but also essential for gaining a competitive advantage. Technological advancements have made it crucial to respond to customer demands quickly and accurately. This study emphasizes the importance of efficiently classifying customer demands and software errors in the branches of a bank operating in the finance sector. Real-world data was divided into three categories: Desktop Support, Software Support, and Field Support, with a total of 4,500 samples equally distributed among the categories. Eighty percent of the data was used for training and 20% for testing machine learning algorithms such as Bidirectional Encoder Representations from Transformers (BERT), Naive Bayes, Random Forest, and Artificial Neural Networks (ANN). The models were trained separately using CountVectorizer and Term Frequency–Inverse Document Frequency (TF-IDF) metrics. The dataset was also analyzed using two sample sizes: 3,000 and 4,500. The best results were obtained with BERT and ANN models using 4,500 samples and the TF-IDF metric, achieving accuracy rates above 92%. The positive effects of increased data size and the TF-IDF metric were evident. Additionally, ANN-based models proved more effective for this type of classification problem.

Keywords: Machine Learning, Classification , Natural Language Processing (NLP)

1. INTRODUCTION

The service sector, as an area where technology is rapidly developing and competition is increasing, requires constant innovation and adaptation. In this context, it is imperative for the service sector to use technology effectively in order to increase efficiency and maximize customer satisfaction. In this study, the banking sector is analyzed within the scope of the service sector. The impact of technological development in the banking sector is used to optimize business processes and improve service quality by automating manual processes.

In this study, it is seen that the requests and errors coming from bank branches are classified by employees in the current system and this process is open to human errors. The manual classification process not only leads to incorrect or incomplete classifications, but also causes the process to proceed slowly. As a result, the slow progress of requests causes customer dissatisfaction and makes it difficult to accurately determine the workload and resource allocation of the teams. This leads to customer dissatisfaction and loss of business. In this study, we investigate how machine learning techniques can be used to automatically classify requests from bank branches as a solution to the problem. In this process, classical algorithms such as Naive Bayes, Random Forest, ANN and advanced algorithms such as BERT are used. The algorithms used in this study show that by providing accurate classification of requests, it will help to reveal the workload of the teams more clearly and help to distribute employee resources more efficiently.

The study presents the automated classification of errors and requests in the banking sector by using machine learning techniques. A total of 4500 data were used in the model training process. The success results of the models were first obtained with 3000 data, and then the number of data was increased to 4500 and it was observed whether there was an increase or decrease in model success. As a textual feature, the widely used, easily applicable CountVectorizer and TF-IDF metrics were extracted from the data and their effects were examined.

The contribution of this study to the literature is that the classification and routing of requests can be automated, especially in service providers such as banking, thus reducing the workload of the teams and increasing customer satisfaction. It is a first in the literature, especially because it is a study conducted on real data and, as far as we know, it is the first time it has been applied in the banking sector. Since the results obtained are much better than other studies in this field, it also offers a good problem-specific model proposal to the literature. Since the available data is obtained from the banking sector, it has been demonstrated that it can achieve achievements that can meet the sensitivity and expectations of financial institutions at a high rate. In addition, by keeping the technical level of this study low, it has been demonstrated that models with high efficiency can be obtained with fast and simple applications. Finally, it provides a roadmap on how service sector organizations such as banks can benefit from technology to increase business efficiency. Machine learning algorithms are effectively applied in classification processes as in many other fields. In particular, studies on the classification of data such as requests, comments and complaints reveal innovations in the field of social media and text mining and reveal the potential offered by these technologies. Various studies in the literature emphasize processes such as sentiment analysis and automatic classification , and extracting meaningful information from these data with machine learning algorithms. Below are examples of studies on this topic. İlhan and Sağaltıcı, emphasize that the rapidly increasing masses of data on social media should be analyzed to understand the emotions and thoughts of individuals. They aimed to classify tweets as positive, negative and neutral using Naive Bayes and Support Vector Machines (SVM) on 1,578,627 tweets obtained from Twitter. The accuracy of the classification was increased with the N-gram method. This data provides important information that can be used for commercial and political purposes. Naive Bayes and SVM methods were compared and the results were improved with the N-gram method. Support Vector Machines achieved high success rates. The resulting sentiment analysis provided data that can be used in marketing and commercial areas. The need to extract meaningful information from social media data has been effective in performing this analysis [1]. Kumas, performed sentiment analysis on Turkish tweets retrieved from the social media platform Twitter. The study aimed to apply text mining methods using a dataset of 32,000 tweets labeled as positive and negative. The main objective is to structure these data with text mining techniques and determine the effectiveness of different classification algorithms. Five different classification algorithms such as Naive Bayes, K Nearest Neighbor (KNN), Support Vector Machines (SVM), Logistic Regression and Decision Tree were used. The TF-IDF term weighting method was applied on the tweets and the data was split into 80% training and 20% testing and tested on the classifiers. In addition, normalization and tokenization operations were performed as data preprocessing steps. In the classification tests, the most successful result was obtained with SVM with an F1 score of 73%. Other algorithms gave similar results; however, SVM showed the best success in general [2]. Yildiz and Yildirim aimed to compare the performance of both traditional and modern methods in text classification to determine which approach might be more appropriate. Two datasets, TTC-3600 and T-4900, were used in the study. The performance of the Bag-of-Words (BoW) method and artificial neural network (ANN) based approaches in text classification is compared. The increase in textual data and the need for automatic labeling in text classification has increased the importance of this study. The advantage of ANN-based models is that they can generate word and text representations more efficiently. This results in shorter and denser vectors than traditional methods. An F1 score of 89.0% was achieved with the Logistic Regression classifier on the T-4900 dataset. On the same dataset, the m-NB approach with traditional bag-of-words based Information Gain feature selection achieved a similar success with an F1 score of 90.0%. On the TTC-3600 dataset, the PV-DM model achieved an F1 score of 92.3%, while the traditional bag-of-words method with Information Gain feature selection was more successful with an F1 score of 93.1%. As a result, ANN-based word and text representations were found to be more effective and efficient than BoW methods [3].

Mesut et al. conducted a study to detect fake news spread in social and digital media. In this study, they aimed to determine whether the news is real or fake by using artificial intelligence and Natural Language Processing methods. An artificial intelligence-based approach is proposed to detect fake news that spreads rapidly on social media and can misinform the public. The dataset consists of 6,335 news headlines and content, of which 3,171 are classified as real news and 3,164 as fake news. In the analysis using Natural Language Processing and Long Short Term Memory (LTM) model, 99.83% accuracy is achieved for training data and 91.48% accuracy is achieved for test data. These results show that the proposed method is very effective in detecting fake news [4]. Toğaçar et al. conducted a study to evaluate the effectiveness of the methods used in the classification process of R&D projects and to improve this process. The objective of this study is to speed up and improve the process of supporting R&D projects by matching them with appropriate referees. By using deep learning methods and natural language processing techniques, it is aimed to increase the classification accuracy and make the processes more efficient. Classification management was performed using Convolutional Neural Networks (CNN) and other traditional classification methods (Naive Bayes, Support Vector Machines (SVM), etc.). Approximately 80% success was achieved with Naive Bayes and SVM, while over 90% success was achieved with Word2Vec and CNN methods. The results show that Word2Vec and CNN methods provide high accuracy rates in the classification of R&D projects and perform better than traditional methods [5]. Görentaş et al. performed automatic classification of dispute court judgements into two different classes. Using data preprocessing and TF-IDF feature extraction methods, court decisions were classified by Logistic Regression, Support Vector Machines, Decision Trees and Random Forest algorithms. A success rate of 87% was achieved in the modelling. The aim of the study is to develop applications that can support the decision-making processes of legal actors and to propose algorithms for potential "virtual judge" artificial intelligence applications [6]. Songpan focuses on automating review predictions by analysing customer reviews. In order to resolve inconsistencies between customer reviews and textual reviews, Naive Bayes and decision tree models are used to classify reviews (positive or negative). In the study, hotel customer reviews were analysed and it was found that the Naive Bayes model outperformed the decision tree with an accuracy rate of 94.37%. This model successfully predicted reviews by analysing trends in reviews more consistently. [7]. In their study, Kaşıkçı and Gökçen aimed to determine whether these pages are e-commerce sites by analysing the content of the websites specified by the user and using text mining methods. This study aims to make it easier for users to find e-commerce websites. Data were collected from different sources to be used in text classification and the results obtained by using KNN (K-Nearest Neighbour) and Naive Bayes classification algorithms were compared. The most successful result was obtained with the k-NN algorithm. For k values "1, 3, 5, 7, 9, 11, 13 and 15", the model success rate was 83%. The algorithm with more successful results was selected and integrated into the desktop application developed with Java programming language [8]. Arslan et al. studied the classification of customer requests of Detay Soft company using text mining and machine learning methods. The data were processed using various preprocessing techniques and vectorisation methods and tested with different machine learning algorithms. The study provides an effective model for routing the correct requests to the relevant departments. TFIDF, Word2Vec and BERT vectorisation methods were also used in this process. The TF-IDF vectorisation method gave the most successful results, while the removal of unimportant words increased the success. However, word smoothing and stemming operations generally resulted in poor performance. The prediction performance was lower because the requests of sub-departments contained similar words. In the test results, 79% success was achieved [9]. Kazan and Karakoca, Internet usage is increasing and with it the amount of storage and daily data is also increasing significantly. This increase usually brings with it unstructured data stacks and increases the need for effective automatic text categorisation systems for the management and organisation of these data. Methods such as Naive Bayes, Logistic Regression, Random Forest, Decision Tree, Support Vector Machines (SVM), ANN were used. Among the methods used, the best result was obtained with SVM with 97% success [10]. In his research, Koruyan focused on the ability of businesses to make the right decisions by analysing numerical and verbal data to increase their profit margins. The study aims to analyse the 2020 customer complaint data of Turkey's three major consumer electronics retail companies. Complaint data were obtained from sikayetvar.com. Four categories, Returns/Exchanges and Refunds, Delivery/Shipping, Customer Relations and Services, and Warranty and Service, were identified and their accuracy was compared using three machine learning algorithms (Logistic Regression, Stochastic Gradient Descent and Linear Support Vector Machines). The highest accuracy among the machine learning results was obtained with the Logistic Regression algorithm with a rate of 80% [11]. In this study, Tekin and Tunalı tried to predict priorities by classifying not only defects but also various enhancement requests with text mining methods. Taking into account the structural difficulties of the Turkish language, this study utilised real demand records from the demand management system of a company operating in Turkey. After cleaning and preprocessing the data, a document-record matrix was created with the TF-IDF method and various classification algorithms were tested. As a result, the Random Forest algorithm achieved the highest success with an F1-score of 74.5% [12]. In this study, Aydemir et al. aimed to categorise Turkish news texts into eight categories (life, world, economy, culture and arts, magazine, automobile, sports, technology) using text mining techniques. The targeted success rate was set as at least 90%. In the study, Turkish news texts were collected and cleaned from unnecessary texts. The texts were analysed using Weka software. Multinomial Naive Bayes Algorithm (MNBA) and Random Forest (RF) algorithms were applied. For the reliability of the data, 10-fold cross-validation technique and kappa coefficient were calculated. The results show that 99.86% success was achieved with 2248 data. Multinomial Naive Bayes Algorithm and Random Forest showed high success in categorising Turkish news texts into correct categories [13].

Sevimli aimed to compare and analyse the performance of rule-based classification algorithms. The rule-based algorithms analysed include fuzzy unordered rule algorithm (FURIA), decision trees and repeated incremental pruning algorithm for error reduction. The aim of the study was to compare these algorithms according to criteria such as correct classification rates, mean absolute error and root mean square error. In addition, it is determined which algorithm performs better on certain data sets and which algorithm works more efficiently on large data sets. The results showed that the FURIA algorithm performed the best compared to other algorithms in terms of correct classification. It was observed that the efficiency of FURIA increased as the data set size increased. Although no specific underperformance was reported for the other algorithms, it was emphasised that FURIA gave the best results in general. The aim of the research is to make a comparative analysis between rule-based algorithms to determine which algorithm is more successful on specific data sets. The FURIA algorithm gave the most successful result [14]. In his study, Binici focused on the automation of standard file plan codes with artificial intelligence and machine learning applications in document management systems. Public institutions organise documents in accordance with the file plan and manual processes can lead to erroneous results due to incorrect coding. In order to eliminate this margin of error, it is aimed to minimise the error rate by automating with machine learning. The Support Vector Machines algorithm achieved a high success rate of 87.72% by automating the standard file plan codes of documents [15].

2. MATERIAL

This section outlines the dataset and the methodologies applied in this study to process and analyze the data effectively. The materials include a detailed description of the dataset, preprocessing techniques, and feature extraction methods, forming the foundation for subsequent classification tasks.

2.1. Data Set

This study uses a dataset of requests and errors recorded in the branches of a bank. The dataset is derived from natural operational processes and reflects real-world scenarios. Requests are created manually through screens accessible to users, while errors are recorded in the system when momentary problems on the screens turn into automatic call logs.

The data set consists of 4,500 data in total and is divided into three main categories: Desktop Support, Field Support and Software Support. The categories are evenly distributed, with each category containing 1,500 data. The Desktop Support category covers requests and problems that can be solved by remote intervention without the need for physical intervention in branches. The Field Support category includes requests for physical devices, branch infrastructure and technical equipment that can be solved through on-site support with physical intervention. The Software Support category includes technical problems and requests that can be solved by application and software development used within the organization.

Each record consists of text-based descriptions provided by the user and the category determined in accordance with these descriptions. The dataset provides textual data to be analyzed with natural language processing techniques. Moreover, the categorical structure of the dataset provides a convenient basis for analyzing semantic differences between different classes and comparing the performance of classification algorithms.Characteristics of the data set:

Source: Requests and bug reports from different branches of the company.

Categories:

- **Desktop Support**: Problems and support requests related to desktop computers.
- **Field Support:** Problems and support requests related to field work.
- **Software Support:** Problems and support requests related to software and applications.

Usage Areas of the Data Set:

- Automatic Categorisation: Automatic assignment of bugs and tickets to the correct categories.
- Resource Allocation: Improving support processes by ensuring accurate and efficient resource allocation.
- Performance Analyses: Process improvements by analysing the performance of support requests and solutions.

This data set constitutes an important resource, especially in terms of the applicability of artificial intelligence and machine learning algorithms and will be used in the improvement and optimisation of support processes.

2.2. Text Preprocessing

Text preprocessing is a set of techniques applied in natural language processing (NLP) projects to make text data suitable for analysis or modelling. This process aims to transform raw text data into a more organised and meaningful form.

- Punctuation marks and special characters were removed from the texts.
- **Removal of stop words**: Frequently used but non-informative stop words (e.g. 'and', 'one', 'or', etc.) were removed from the texts.

2.3. Feature Extraction

They are criteria or variables that represent important and meaningful information in a dataset. In machine learning and data analysis processes, attributes are inputs used for the model to learn and make predictions.

2.3.1. TF-IDF (Term Frequency - Inverse Document Frequency)

TF-IDF is a feature extraction method used in text mining. It is used to determine how important each word is in a collection of documents.

The following steps were followed to extract TF-IDF values from the texts:

- **Calculation of Term Frequency (TF)**: The frequency of each term in all documents was calculated.
- **Calculation of the Inverse Document Frequency (IDF)**: Calculated how rare the term is among the documents in the collection.
- **Calculation of TF-IDF Values**: TF-IDF value was obtained by multiplying TF and IDF values.

2.3.2. Count Vectorizer

CountVectorizer, a basic feature extraction method used in text mining and natural language processing, converts texts into numerical data and makes them usable in machine learning algorithms. This method creates a vocabulary of words in the entire text collection and represents each document with a vector by counting the frequency of each word in the document. The resulting vectors are organised into a numerical matrix of documents according to their word frequencies. However, CountVectorizer only considers the frequency of words and does not take into account grammatical relations or the meaning of words. Therefore, it is often used in combination with other methods such as TF-IDF.

The data set used in the study will be subjected to classification processes using machine learning techniques according to the preprocessing and attribute steps given above. The aim is to ensure that each error is automatically assigned to the relevant team correctly through correct classification. For this purpose, various algorithms (Naive bayes, Random Forest, ANN, BERT) will be used to perform the process.

3. METHODS

In this study, the errors or requests generated from the screens used in bank branches are subjected to classification with machine learning models to determine which class they belong to. Python programming language was used for classification analysis. The flow diagram of the study is shown in Figure 1.

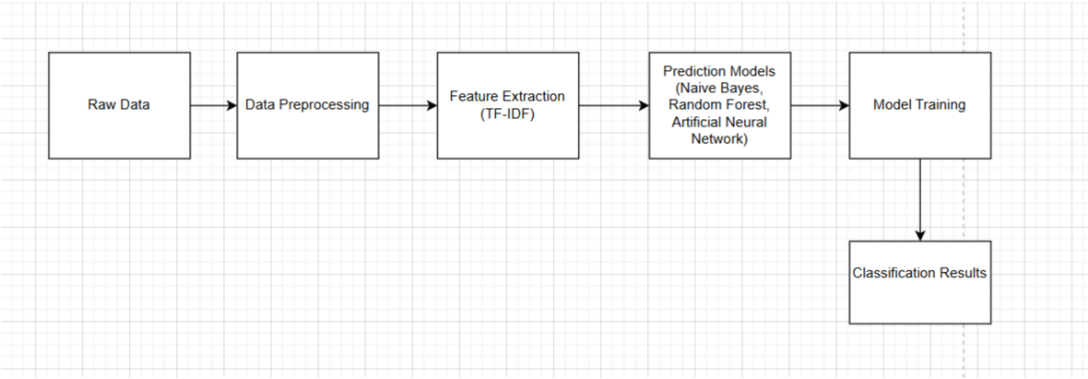
Raw Data
Data Preprocessing
Feature Extraction (TF-IDF)
Prediction Models (Naive Bayes, Random Forest, Artificial Neural Network)
Model Training
Classification Results

Figure 1. Flow Chart

3.1. Model Setup, Evaluation, Training and Testing

The data set, which consists of errors and requests obtained from the screens used by the branches, contains 1500 randomly selected data for each category: The data received by the Field Support, Desktop Support and Software Support teams were received in raw form without interpretation. The data was subjected to pre-processing steps. For each category, the data was randomly split into 80% training and 20% testing sets. Model setup, training, testing and prediction were performed using Python programming language. Spyder in Anaconda, a popular application development environment for machine learning and data science, was used. Table 1 presents examples for each category. The data in the table are the original versions without word correction and preprocessing steps.

Table 1. Requests and Errors from Branches

Desktop Support	merhabalar, sicil numaralı ayşe arkadaşımız doğum sonrası mayıs itibariyle çalışmaya başlayacaktır, kullanıcısının aktif hale getirilmesi konusunda desteğinizi rica ederiz.
Field-Support	iyi çalışmalar ; merter ticari şubede göreve başlamam sebebiyle notebool gönderilmesi konusunda yardımlarınızı arz rica ederim.
Software Support	merhaba, müşterimiz dosya yüklerken hata almaktadır. konu husususnda derğerli yardımlarınızı arz ederim.

3.2. Methods Used

In the study, Naive Bayes, Random Forest, ANN, BERT machine learning algorithms were used for classification.

3.2.1. Naive Bayes

Naive Bayes (NB) classification algorithm is a simple but powerful method that estimates the probability that an instance in a dataset belongs to a particular class using Bayes' Theorem. Bayes' Theorem is expressed as follows (Equation 1) [16].

$$P(c \mid x) = \frac{P(x \mid c) \cdot P(c)}{P(x)} \tag{1}$$

In this equation:

- **c**: The class to be predicted,
- **x**: Data point or feature vector,
- **P(c\x)**: The probability (posterior probability) that class c occurs when data point or feature vector x occurs,

- **P(x\c)**: The likelihood of a data point or feature vector x occurring when class c occurs,
- **P(c)**: The probability of class c occurring,
- **P(x)**: The probability (normalisation constant) that the data point or feature vector x occurs.

3.2.2. Random Forest

Random Forest is a supervised learning algorithm and can be used in both classification and regression problems. It is characterised by its flexibility and ease of use. This algorithm builds decision trees on randomly selected data samples, takes predictions from each tree and determines the best result by combining these predictions through voting [17]. Random samples are taken from the data set. A decision tree is created for each sample and predictions are made from these trees. For each predicted result, a voting process is performed and the result with the most votes is selected. This process helps to increase the overall success of the model because different trees make different predictions, making the model more robust and reliable.

3.3.3. Artificial Neural Networks

ANNs are machine learning models that process data by mimicking the working principle of nerve cells in the human brain. It usually learns from inputs using multilayer structures and makes classification or prediction based on these inputs.

There are many different types of ANNs. In this study, Multi-Layer Perceptron (MLP) structure is used as ANN. MLP is a widely used model in classification problems. The MLP network consists of an input layer, a hidden layer and an output layer. The hidden layer consists of 4 neurons and each neuron processes the input data with the help of non-linear activation functions. During the training of the model, the Downslope and Back Propagation algorithms are used to optimise the weighted connections of the network.

In this model, the text data is first converted into a numerical form with CountVectorizer, and then this converted data is given to the MLP classifier. MLP processes the input vectors and produces classification output. The output is compared with the target classes and evaluated with metrics such as model accuracy, precision, recall rate and F1 score. The error between the target values and the predicted values of the model is minimised by the Mean Squared Error approach [18].

3.3.4. BERT

The BERT model [19] is a transducer model introduced by Google in 2018, designed to pre-train pairwise representations from unlabelled text and then fine-tune them using labelled text for different NLP tasks. With BERT, the translation success of the Google Translate application has also increased significantly, and now this application can translate even very long sentences between different languages with a high degree of accuracy without loss of meaning [20][21]. The translators have a self-attention mechanism based on a queueing structure. The BERT model maps a query and a set of key-value pairs to an output. Here, vectors are formed to express the correlation between the query, keys, values and output. The output is calculated as a weighted sum of the values. The weight assigned to a value is calculated by its compatibility with the key corresponding to the query [22]. The BERT model processes a text both from right to left and from left to right, so that it can learn the relationships between the elements in the text. In the training phase, MLM (Masked Language Modelling) and NSP (Next Sentence Prediction) techniques are used. In the MLM technique, masked words are predicted by using open (unmasked) words. With this technique, analysis and prediction are made over the words in a sentence. In the NSP technique, the relationship of sentences with each other is analysed by looking at the relationship of a sentence with the following sentence. In the structures established with the BERT model, a pre-trained model is required.

4. EXPERIMENTS

The experiments performed within the scope of the study are listed below. The experiments were performed on 3000 and 4500 data samples. For each dataset, CountVectorizer and TF-IDF features were used separately to examine the performance metrics of the models. The ROC curve in Figure 2 compares the performance of Naive Bayes, Random Forest and Artificial Neural Network (ANN) models.

4.1. Naive Bayes

The results obtained using the Naive Bayes algorithm are given in Table 2.

Table 2. Naive Bayes Performance Values Table

	Tf-Idf		CountVectorizer	
	3000 data	4500 data	3000 data	4500 data
Accuracy	0.902	0.915	0.903	0.905
Certainty	0.900	0.913	0.903	0.905
Sensitivity	0.900	0.913	0.901	0.904
F1-Score	0.900	0.913	0.901	0.904
Loss of education	0.278	0.286	0.110	0.100

Since Naive Bayes algorithms do not have an epoch-based training process, the training loss refers only to a certain point.

4.2. Random Forest

The Random Forest algorithm does not directly provide a training loss metric, because this classifier model does not train using the loss function in the optimisation process. However, in order to monitor the training process, training and test accuracy values were used as evaluation criteria and the model was configured with 50 trees.

The results obtained using the Random Forest algorithm are given in Table 3.

Table 3. Random Forest Performance Values Table

	Tf-Idf		CountVectorizer	
	3000 data	4500 data	3000 data	4500 data
Accuracy	0.831	0.861	0.821	0.856
Certainty	0.830	0.862	0.822	0.857
Responsiveness	0.830	0.859	0.820	0.855
F1-Score	0.830	0.859	0.820	0.855

The most successful training and test accuracy graphs obtained by applying the Random Forest algorithm are given in Figure 2 below. These graphs are prepared in order to visualise the performance of the model in the training process and test phase.

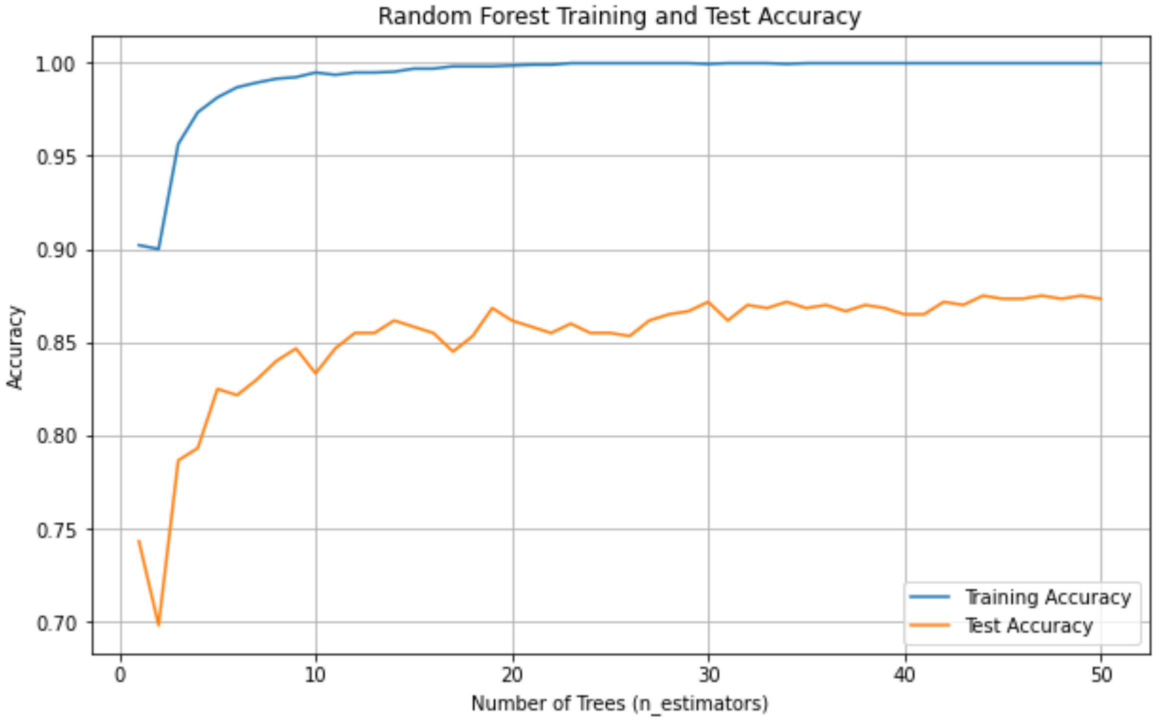

Figure 2. TF-IDF approach with 4500 data

4.3. Artificial Neural Network

Using the neural network model, comprehensive evaluations were performed on 4500 and 3000 data samples. The performance of the model was analysed with TF-IDF and CountVectorizer feature extraction methods and the results for both data sets are given in Table 4 below.

Table 4. Artificial Neural Network Performance Values Table

	Tf-Idf		CountVectorizer	
	3000 data	4500 data	3000 data	4500 data
Accuracy	0.870	0.926	0.890	0.901
Certainty	0.868	0.921	0.886	0.909
Responsiveness	0.868	0.921	0.886	0.909
F1-Score	0.868	0.921	0.886	0.909

The most successful training and test accuracy graph obtained using the artificial neural network is given in Figure 3 below. This graph is prepared to visualise the performance of the model in the training process and in the test phase.

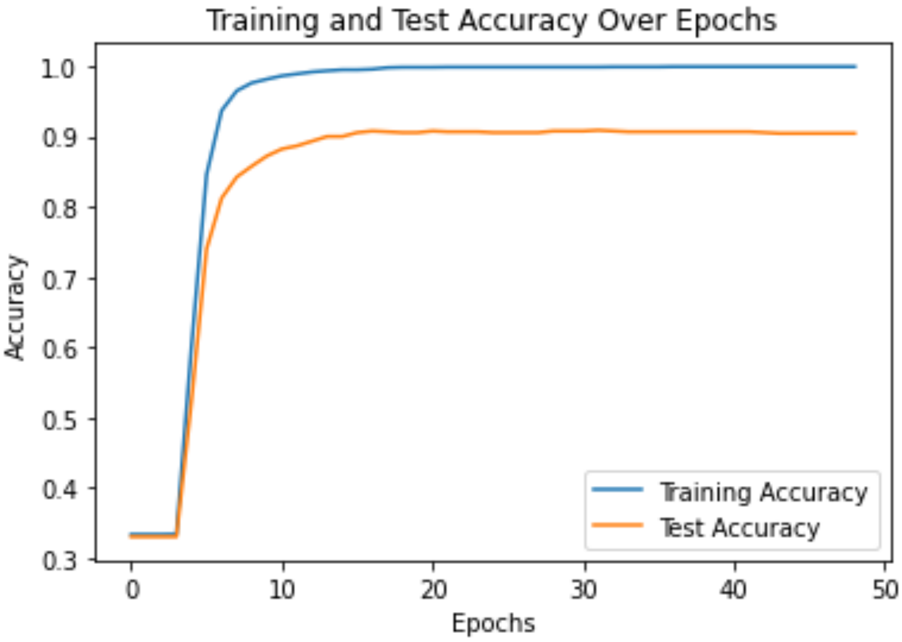

Figure 3. TF-IDF approach with 4500 data

4.4. BERT

Using the BERT model, extensive evaluations were made on 4500 and 3000 data samples. In the BERT model, feature extraction is performed within the architecture. While 89% success was achieved with 3000 data samples, the success rate increased to 91% when the number of data was increased to 4500.

4.5. ROC-AUC Curve Evaluation

Figure 4 shows the ROC curves and AUC (Area Under the Curve) values of Naive Bayes, Random Forest and ANN models in comparison. The Naive Bayes and ANN models perform close to or slightly better than each other, while the Random Forest model is inferior to the other two models.

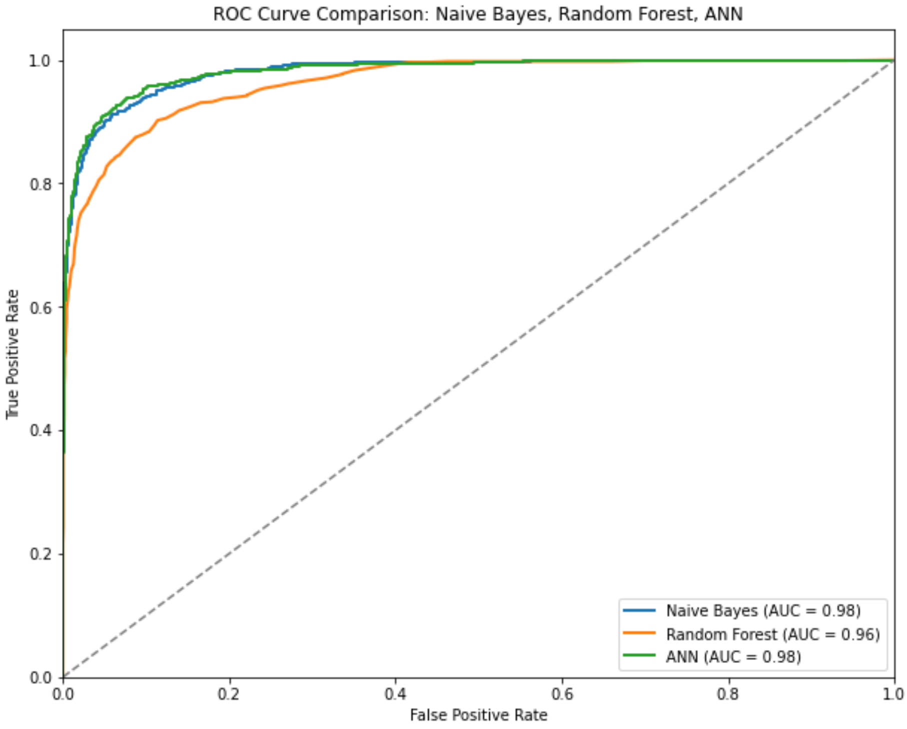

Figure 4. ROC-AUC Curve

5. RESULTS

This study emphasizes the importance of automatic classification in the banking sector and aims to provide a basis for future research. As a result of the literature review, there is no previous study on demand and error classification in the banking sector. In this respect, the study is the first of its kind and offers a problem-specific sector standard to the literature.

When different studies on similar demand, complaint and other similar classifications are examined, it is seen that 79% accuracy is obtained in the study [9], which was carried out in a different sector, and an average of 80% accuracy is obtained in other classification studies. In this study, the results obtained on real data showed an acceptable classification success of over 92%. In this respect, the study presents a successful model to the literature. When the experiments are analysed, it is concluded that increasing the number of data increases the success, so the model performance can be improved with more data. The TF-IDF feature was found to be slightly more discriminative than the CountVectorizer method. The best result was obtained using ANN (Artificial Neural Networks) with an accuracy of 92%. The ANN algorithm showed a high performance with TF-IDF, reaching up to 92.6% accuracy, especially in experiments with 4500 data. Accuracy, sensitivity and F1-Score metrics also exceeded 92%, indicating superior performance of TF-IDF. ANN achieved the highest accuracy rate with TF-IDF and outperformed the other algorithms. A high accuracy rate of 91% was also achieved with the deep learning model BERT.

The high success rate on real data shows that requests can be automated and directed to teams with high accuracy through the automatic classification process. For the bank, requests or errors can be automated to reach the relevant people more quickly instead of being sent to the relevant people through personnel. In this way, customer satisfaction and productivity can be increased by taking action on requests faster. The correct classification process can play an active role in the correct calculation of the man/daytime allocated to the teams at the end of the year, resulting in more effective use of personnel and personnel savings.

These results show that there are significant performance differences between the algorithms. In particular, ANN and BERT models achieve high accuracy rates, suggesting that more complex models are effective in the classification task. However, in order to address the lack of statistical significance discussion, it would be useful to use statistical tests to evaluate the performance differences between the models. For example, statistical analysis could be performed to determine whether the differences in accuracy rates are random or significant. Such an analysis would provide a stronger argument for algorithm selection and implementation. However, in the current study, the primary goal was to emphasize the overall performance of the algorithms.

In the future, by increasing the number of data and conducting more extensive experiments, the generalization capabilities of the models can be better understood, and more robust classification systems can be developed. Furthermore, data augmentation techniques can be used to obtain artificially diversified data sets and improve the adaptability of the models to different conditions. With transfer learning approaches, the training process can be shortened, and performance can be improved by using large language models that are pre-trained for similar tasks or pre-trained models from different sectors. In this way, automatic classification systems specific to the banking industry can be made more robust, scalable and efficient. These approaches can be applied not only in the banking sector, but also in other industries with similar structures, providing a wider range of automation and productivity gains.

REFERENCES

[1] İlhan, N., & Sağaltıcı, D. (2020). Sentiment Analysis on Twitter. Harran University Engineering Journal, 5(2), 146-156. https://doi.org/10.46

[2] E. Kumaş, "Comparison of Classifiers in Sentiment Analysis from Turkish Twitter Data", ESTUDAM Bilişim, vol. 2, pp. 2, pp. 1-5, 2021.

[3] Yıldırım, S., & Yıldız, T. (2018). Comparative text classification analysis for Turkish. Pamukkale University Journal of Engineering Sciences, 24(5), 879-886.

[4] Toğaçar, M., Eşidir, K. A., & Ergen, B. (2022). Detection of Fake News Published on the Internet Using Artificial Intelligence-Based Natural Language Processing Approach. Journal of Intelligent Systems: Theory and Applications, 5(1), 1-8. https://doi.org/10.38016/jista.950713

[5] Kocak, S., İç, Y. T., Sert, M., Dengiz, B. (2023). Natural Turkish language processing based method for classification of R&D projects. Gazi University Journal of Engineering and Architecture Faculty, 38(3), 1375-1388. https://doi.org/10.17341/gazimmfd.889395

[6] Görentaş, M. B., Uçkan, T., & Bayram Arlı, N. (2023). Classification of Court of Dispute Decisions with Machine Learning Methods. Journal of Yüzüncü Yıl University Graduate School of Science and Technology, 28(3), 947-961. https://doi.org/10.53433/yyufbed.1292275

[7] W. Songpan, "The analysis and prediction of customer review rating using opinion mining," *2017 IEEE 15th International Conference on Software Engineering Research, Management and Applications (SERA)*, London, UK, 2017, pp. 71-77, doi: 10.1109/SERA.2017.7965709.

[8]Kaşıkçı, T., & Gökçen, H. (2014). Identification of E-Commerce Sites with Text Mining. Journal of Information Technologies, 7(1). https://doi.org/10.12973/bid.2014]

[9] Arslan, H., Dadaş, I. E., & Işık, Y. E. (2022). Demand Classification with Different Vectorisation and Preprocessing Methods. Düzce University Journal of Science and Technology, 10(3), 1433-1442. https://doi.org/10.29130/dubited.1017422

[10]S. Kazan and H. Karakoca, "Product Category Classification with Machine Learning", *SAUCIS*, vol. 2, pp. 1, pp. 18-27, 2019, doi: 10.35377/saucis.02.01.523139.

[11] Classification of Customer Complaints with Machine Learning* Kutan KORUYAN, Dokuz Eylül University, Faculty of Economics and Administrative Sciences, Department of Management Information Systems, Ph. Member, 0000-0002-3115-5676

[12] Tekin, M. C., & Tunalı, V. (2019). Prioritisation of software development requests with text mining methods. Pamukkale University Journal of Engineering Sciences, 25(5), 615-620.

[13] Aydemir, E., Işık, M., & Tuncer, T. (2021). Classification of Turkish News Texts by Using Multinomial Naive Bayes Algorithm. Fırat University Journal of Engineering Sciences, 33(2), 519-526. https://doi.org/10.35234/fumbd.871986

[14] Sevimli Deniz, S. (2021). Comparison of Rule-Based Classification Algorithms. Data Science, 4(3), 72-80.

[15] Binici, K. (2019). A Study on Automatic Assignment of Standard File Plan Numbers to e-Documents by Machine Learning Approach. Knowledge Management, 2(2), 116-126. https://doi.org/10.33721/by.654464

[16] H. Deng, Y. Sun, Y. Chang, J. Han, "Probabilistic Models for Classification" in C.C. Aggarwal (Eds.), Data Classification Algorithms and Applications (pp. 67-70), CRC Press, New York, USA, 2015.

[17] G. Louppe, "Understanding Random Forest", PhD thesis, University of Liege, 2015.

[18] J.M. Zurada, "Introduction to Artificial Neural Systems", West Publishing Company, 1992.

[19] Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K., "Bert: Pre-trainingof deep bidirectional transformers for language understanding," arXivpreprint arXiv:1810.04805, 2018.

[20]Gupta, Shashij, et al. "Machine translation testing via pathological invariance." Proceedings of the 28th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering. 2020.

[21] Do, Quang-Minh, Kungan Zeng, and Incheon Paik. "Resolving Lexical Ambiguity in English-Japanese Neural Machine Translation." 2020 3rd Artificial Intelligence and Cloud Computing Conference. 2020.

[22]Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez,A. N., Kaiser, L., and Polosukhin, I., "Attention is all you need," arXivpreprint arXiv:1706.03762, 2017