Çukurova Üniversitesi
Mühendislik Fakültesi Dergisi
Çukurova University
Journal of the Faculty of Engineering
ISSN: 2757-9255 CİLT/VOLUME: 39 SAYI/ISSUE: 4 ARALIK/DECEMBER 2024

Crime Prediction with DistilBERT-based Feature Extraction and Machine Learning

Emel ÇOLAKOĞLU [1,a], **Serhat HIZLISOY** [2,b], **Recep Sinan ARSLAN** [2,c]

[1]*Kayseri University, Rectorate, Kayseri, Türkiye*
[2]*Kayseri University, Faculty of Engineering, Architecture and Design, Department of Computer Engineering, Kayseri, Türkiye*

[a]***ORCID**: 0000-0003-1755-3130; [b]**ORCID**: 0000-0001-8440-5539; [c]**ORCID**: 0000-0002-3028-0416*

Article Info
Received : 26.07.2024
Accepted : 23.12.2024
DOI: 10.21605/cukurovaumfd.1606169

Corresponding Author
Emel ÇOLAKOĞLU
emelcolakoglu@kayseri.edu.tr

Keywords
Google BERT
Natural language processing
Crime analysis
Machine learning

***How to cite:** ÇOLAKOĞLU, E., HIZLISOY, S., ARSLAN, R.S., (2024). Crime Prediction with DistilBERT-based Feature Extraction and Machine Learning. Cukurova University, Journal of the Faculty of Engineering, 39(4), 1067-1079.*

ABSTRACT
Crime is all actions and behaviors that harm societies and have a legal and criminal counterpart. Although the fight against crime is basically interpreted as the duty of the state, practices similar to this study are important in order to support the struggle. Because it can create situations that can be interpreted with different analyzes made on crime data. From this point of view, additional measures taken will be an auxiliary element in the fight against crime. Being able to predict the crime that may occur ensures that it is prevented before the crime situation occurs. Therefore, the analysis and prediction of crimes is important in identifying and reducing future crimes. In this research, a model in which features are obtained with DistilBERT and 8 different machine learning algorithms are used as classifiers is proposed. The San Francisco crime dataset, which was used for an online competition managed by Kaggle Inc, was used as the dataset. Unlike the literature, all crime categories (39 categories) in the dataset were included in the study. In addition, obtaining features with DistilBERT is another point that differentiates the study. GridSearchCV was preferred for parameter optimization and a general improvement was observed in the range of 1-2% compared to the default parameters. The highest accuracy rate was accomplished with the Support Vector Machine (SVM) with 99.78%. In addition, with 10-fold cross-validation, higher accuracy values were achieved in SVM and Logistic Regression (LR) classifiers.

DistilBERT Tabanlı Özellik Çıkarma ve Makine Öğrenimi ile Suç Tahmini

Makale Bilgileri
Geliş : 26.07.2024
Kabul : 23.12.2024
DOI: 10.21605/cukurovaumfd.1606169

Sorumlu Yazar
Emel ÇOLAKOĞLU
emelcolakoglu@kayseri.edu.tr

Anahtar Kelimeler
Google BERT
Doğal dil işleme
Suç analizi
Makine öğrenmesi

***Atıf şekli:** ÇOLAKOĞLU, E., HIZLISOY, S., ARSLAN, R.S., (2024). Crime Prediction with DistilBERT-based Feature Extraction and Machine Learning. Cukurova University, Journal of the Faculty of Engineering, 39(4), 1067-1079.*

ÖZ
Suç toplumlara zarar veren yasal olarak da cezai bir karşılığı da olan tüm eylem ve davranışlardır. Suçla mücadele temelde devletin görevi olarak yorumlanmakla birlikte bu çalışmaya benzer uygulamalar mücadeleyi destekleyebilmek adına önemlidir. Çünkü suç verileri üzerinden yapılan farklı analizler ile yorumlanabilir durumlar ortaya çıkarabilir. Buradan hareketle alınan ek tedbirler suç ile mücadele de yardımcı öge olmuş olur. Oluşabilecek suçun tahmin edilebilmesi suç durumu oluşmadan önlenmesini sağlar. Bu nedenle suçların analizi ve tahmini gelecekteki suçları belirlemede ve azaltmada önemlidir. Bu çalışmada DistilBERT ile özniteliklerin elde edildiği ve 8 farklı makine öğrenim algoritmasının sınıflandırıcı olarak kullanıldığı bir model önerilmiştir. Veriseti olarak Kaggle Inc. Tarafından yönetilen çevrimiçi bir yarışma için kullanılan San Francisco suç veriseti kullanılmıştır. Literatürden farklı olarak verisetindeki tüm suç kategorileri (39 kategori) çalışmaya dâhil edilmiştir. Ayrıca DistilBERT ile özniteliklerin elde edilmesi de çalışmayı farklılaştıran diğer bir noktadır. Parametre optimizasyonu için GridSearchCV tercih edilmiş ve default parametrelere göre 1-2% aralığında genel iyileşme gözlemlenmiştir. En yüksek doğruluk oranı 99.78% ile Destek Vektör Makinesi (DVM) ile elde edilmiştir. Ayrıca 10 kat çapraz doğrulama ile de yine DVM ve Lojistik Regresyon (LR) sınıflandırıcılarında daha yüksek doğruluk değerlerine ulaşılmıştır.

1. INTRODUCTION

Crime refers to actions defined as legally determined and socially harmful [1]. The most distinctive feature of crime is that it can occur anywhere or at any time. This makes predictability difficult [2].

Crime is one of the most common and worrying situations around the world. The frequency of crime is rising daily and this negatively affects people's lives. Of course, crime analysis and prevention before it occurs is also important in this process [3]. It is fundamentally the responsibility of the security units to manage and mitigate this issue. However, in order to fulfill this task, the crimes committed must be analyzed in detail and the threat levels must be determined. In order to carry out these analyzes, crime data are prepared in many countries and cities and shared for studies [4]. Based on these data, studies have been carried out and continue to be carried out on crime analysis and prediction of crime [3] [5]

Criminology is the scientific study of the scope, causes, management, control, results and prevention of criminal behavior [6]. Criminologists and statisticians conduct studies to analyze crime data and achieve a certain degree of success. However, the increase in the volume of crime and the differences in modern crime make this analysis difficult [7]. In addition, processing this data requires significant human and time resources; since the human being is the controller of the process, it may not be possible to obtain all the relationships/qualities [6]. In this case, it is necessary to involve new techniques in the process in order to analyze the crime [8].

Technological developments in every field have led to analytical approaches to crime [9]. Thus, systematic analysis of crime with approaches such as machine learning, deep learning, and data mining has been included in the process. The introduction of the projects developed in this context will help the security units to use their resources more effectively, to predict crime to some extent, and to fight crime effectively [2]. It is not possible to completely prevent crime with these analyzes. However, additional measures can be taken in sensitive areas and according to the crime group [10].

The use of analytical, especially quantitative techniques, to identify targets, prevent or solve crime is actually the concept of predictive policing. The most important task of this field is to proactively predict criminal activity. In fact, predictive policing is a holistic structure that includes data analysts, developers, and law enforcement [11].

Analysis and prediction of crime is a process that helps to reduce and deter crime [8] .The results of our literature review on crime analysis are as follows:

Khan et al. [2], used San Francisco crime data as a dataset in their research in 2022. They included the top 10 crimes, which make up 97% of the dataset. In their study, they used Gradient Boosting Decision Tree, Naive Bayes (NB) and Random Forest (RF) algorithms to predict and classify crimes into violent and non-violent crimes. A two-class output is intended. They extracted the features from the original dataset. They achieved the best result with Gradient Boosting with 98.5%.

Abouelnaga [12], conducted an analysis on San Francisco crime data in the year 2016. Along with the 3 components of PCA that maintain the highest variance, Hour, Month, County, Day of the Week, Longitude, Latitude, Street No, Block attributes were used. The classifiers were XGBoost (XGB), Bayesian, Decision Tree (DT), RF and K Nearest Neighbor (KNN). The best result was obtained with Random Forest, yielding a log-loss of 2.39031.

Arslan et al. [4], used the San Francisco crime dataset in their 2023 study. The distinguishing feature of this study compared to similar works is the processing of attributes as text and converting them into vectors. Using 10-fold cross-validation, they achieved 99.80% accuracy for the 15 crime categories with the highest incident rates.The study also includes a stacking ensemble model comprising eight machine learning models.

Chandrasekar et al. [13], used Gradient Boosted Decision Trees, NB, SVM and RF classification algorithms in their study. Three different classification analyses were conducted: 39 crime categories, white-collar and blue-collar crimes and non-violent and violent crimes. The dataset was San Francisco crime data. Pre-processing was carried out before classification. United States Census data were also used to develop the

feature set. This data includes demographic data such as the average income level of the neighborhoods and racial diversity. Low recognition values were obtained for 39 classes. The best result for Blue/White Collar Crime was 96.3% (Gradient Boosted Decision Trees) and 75.02% for Severe/Non-Violent Crime classification.

Arslan et al. [14], used the San Francisco dataset and reached an accuracy value of 86.5% with the Random Forest classifier.

Pradhan [15], used the San Francisco crime dataset (2003-2018) in her master's thesis. Data cleaning, data transformation, data reduction pre-operations had been implemented. Five different classifiers were used: Random Forest, Logistic Regression, Naive Bayes, K-Nearest Neighbor and Decision Tree. The best result was obtained in the Decision Tree with 2.3928 log-losses.

Abdulrahman et al. [10], used the San Francisco crime dataset as part of the study. KNN and Naive Bayes were the classifiers whose results were examined. The model was designed using 8 features, including dates, category, description, day of the week, police department area, solution, address, and X&Y coordinates. The best result was obtained with multinomial Naive Bayes using cross validation with 2.611 log-loss.

Bilen et al. [16], examined different regression models in their study in 2022, including Linear, Polynomial, Ridge, and Lasso regression models. The dataset used is the Elazığ cyber dataset. They achieved 79% success with Polynomial Regression.

Arslan et al. [3], used machine learning algorithms to predict where crimes will occur. The dataset contains 49030 samples, 62 different crime categories, and 12 characteristics related to different crimes that took place in New York State in 2019. 99.9% accuracy was achieved with the decision tree classifier.

Sarzaeim et al. [17], used the Random Forest algorithm on the ML side and BART, GPT-3, and GPT-4 on the LLM side. In all prompting or fine-tuning scenarios, the corresponding OpenAI API for GPT models and the Hugging Face API were used to interact with the BART model. San Francisco and Los Angeles were the datasets used. In both datasets, crime groups were combined and grouped into 10 common classes. Since the datasets were unbalanced, precision, accuracy, recall, and weighted average of the F1 score were taken into account. The best result in the SF dataset was obtained with fine-Tuned GPT3 and the weighted average is 97%. The best result in the LA dataset was obtained with Few-Shot GPT4 and is around 60%.

Selvakumari ve Peter [18], used the real dataset, The DCRB (District Crime Record Bureau) in their study. The dataset consists of six classes. Tokenization, lower casing, lemmatization, stop word removal and stemming preprocessing processes were applied on the dataset. In classification, 1D CNN, GRU and autoencoder techniques were preferred. The highest accuracy value was achieved in the autoencoder technique with 97.4%.

Bharath et al. [19], used more than 6000000 records in the Chicago Police Department's CLEAR (Citizen Law Enforcement Analysis and Reporting) system as crime data. Various pre-processing steps had been applied to the data. 3 different models had been created. These were analysis and time series forecasting according to the type of crime and the location of the crime. The problem was analyzed with four different classifiers including Decision Tree, Random Forest, Logistic Regression and Support Vector Machine. The highest accuracy value was obtained in Random Forest with 98% according to the crime type and 97% KNN according to the location of the crime. The proposed technique went beyond simple annual averages; it had given a deeper understanding of future crime trends. The years 2018 and 2019 were estimated from the model and it was observed that they were close compared to the actual values.

Djon et al. [20] (2023), 86% f-score was obtained with XGB. Only the crime of burglary was analyzed in the Chicago Crime Dataset.

Butt et al. [21] (2024), used six deep learning and statistical methods (LSTM, SMA, EMA, LSTM-CNN, WMA and BiLSTM) to generate accurate predictions in New York (30 categories), the Chicago (28 categories) and Lahore (20 categories) crime datasets. Among the various algorithms, BiLSTM demonstrated the best performance, with minimal MAE, MAD, and MSE values. Additionally, this study

introduced a BiLSTM-based architecture, chosen for its superior accuracy in forecasting weekly and monthly crime trends.

2. METHOD

Figure 1 illustrates the architectural structure of the study. The designed framework essentially consists of obtaining the dataset, applying a series of preprocessing steps on the dataset, extracting features with DistilBERT, performing parameter optimization using GridSearchCVand classification processes with machine learning algorithms.

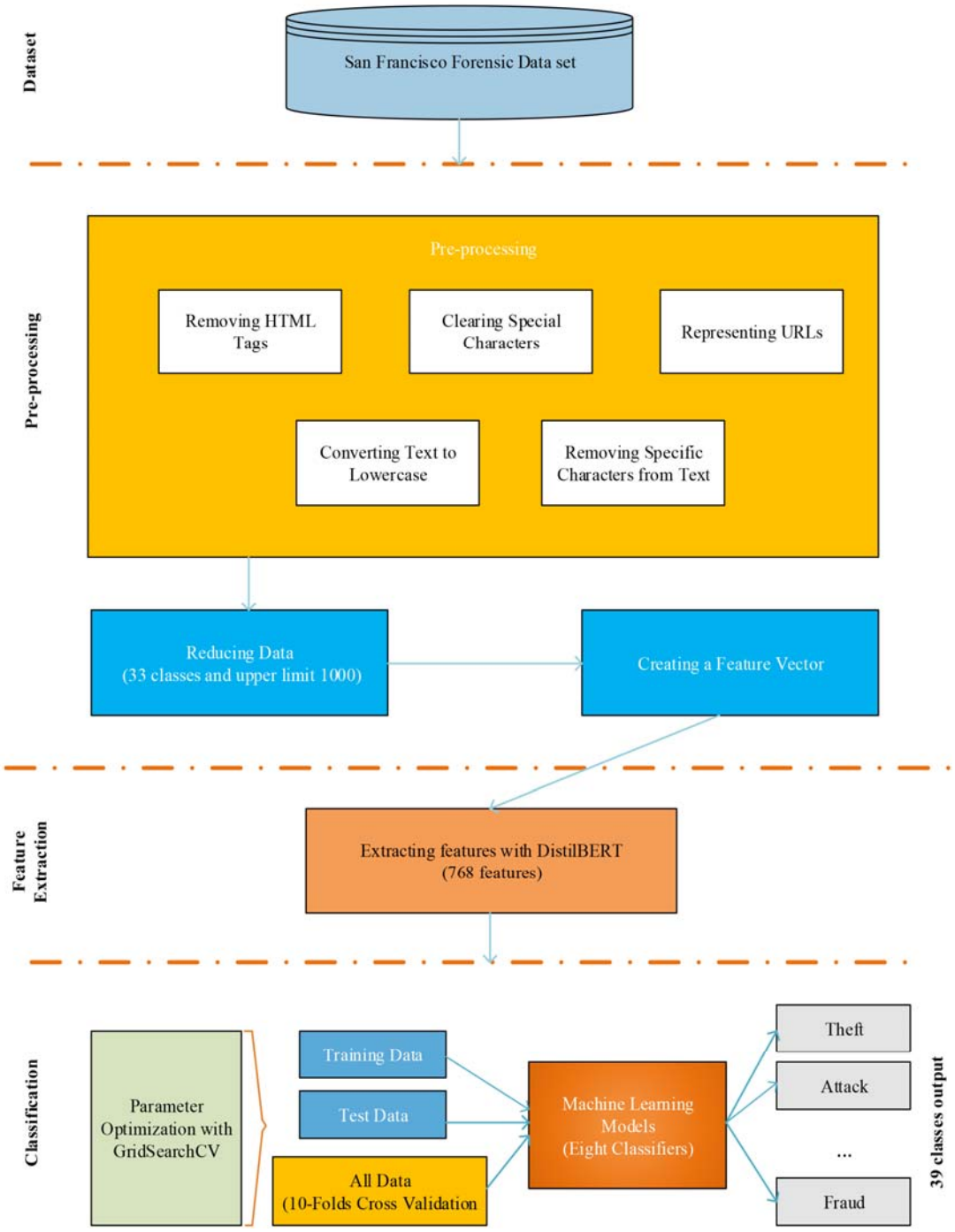

Figure 1. Architectural structure of the model

2.1. Dataset

In this research, the San Francisco crime dataset was used. San Francisco Crime Data is an open-source dataset that can be used for an online competition managed by Kaggle Inc [2]. It consists of 878049 criminal records belonging to 39 different types of crimes that took place between 2003 and 2015 [1]. The dataset includes crime data from all of San Francisco's neighborhoods.

Dataset is partitioned into two parts as training and test data. It consists of nine variables. These are the date of the crime, its day of the week, category, description, how the incident was resolved, the name of the police district, the address of the incident, latitude and longitude [22]. Although the number of samples in all categories does not show a balanced distribution. 39 crime types were added to this study in order to obtain higher recognition rates with more categories compared to similar studies in the literature.

Table 1 shows the distribution of the entire dataset by category.

Table 1. San Francisco crime dataset number of crimes by category

Type of crime	Number of crimes	Type of crime	Number of crimes
Arson	1513	Non-criminal	92304
Assault	76876	Other offenses	126182
Bad checks	406	Pornography/obscene mat	22
Bribery	289	Prostitution	7484
Burglary	36755	Recovered vehicle	3138
Disorderly conduct	4320	Robbery	23000
Driving under the influence	2268	Runaway	1946
Drug/narcotic	53971	Secondary codes	9985
Drunkenness	4280	Sex offenses forcible	4388
Embezzlement	1166	Sex offenses non forcible	148
Extortion	256	Stolen property	4540
Family offenses	491	Suicide	508
Forgery/counterfeiting	10609	Suspicious occ	31414
Fraud	16679	Trea	6
Gambling	146	Trespass	7326
Kidnapping	2341	Vandalism	44725
Larceny/theft	174900	Vehicle theft	53781
Liquor laws	1903	Warrants	42214
Loitering	1225	Weapon laws	8555
Missing person	25989	**Total**	**878049**

2.2. Pre-Processing

Before extracting features from the San Francisco crime dataset, a series of preprocessing methods were tested, and the methods that positively contributed to classification were selected. In this context, preliminary operations such as stopwords, snowballstemmer, tokenization, lemmatize, removing HTML tags, clearing special characters, representing URLs, converting text to lowercase and removing certain characters from the text were tested, and those other than stopwords, snowballstemmer, tokenization, lemmatize were used in the study.

Previously, eight variables of the dataset consisting of 9 variables, excluding the crime category, were combined in a single column and determined as Features, while the Crime Category field was used to represent the label part.

When the dataset was examined, it was seen that there were serious numerical differences in the number of samples for 39 different crime groups. When we perform model training without considering these differences, it is possible that there will be a convergence to the groups with a significantly higher sample number. For this reason, it was understood that it was necessary to make a balance between classes in the number of samples. In order to achieve this balancing, it is possible to use (1) Random oversampling or

undersampling methods. (2) It is possible to select a random sample according to a certain number. Oversampling will require a large number of synthetic samples to be produced, which reduces the reliability of the proposed model. When undersampling, there are criminal groups where there are only six samples, as in the case of TREA. For this reason, the number of samples is very low in this case. For these reasons, in order to provide a partial balance in the number of samples, random selection is used in crime groups with 1000 or more samples.

In addition, although 39 crime categories were added to the study, the maximum number of data was determined as 1000 and our dataset consists of 32272 records.

2.3. Extracting Features

After the preprocessing process was completed, the next step was the extraction of features from the crime dataset. Here, DistilBERT was used to obtain the features. 768 features were obtained for 32272 records. While obtaining these features, a partial calculation with 500 records at a time was performed.

Bert Model (Bidirectional Encoder Representations from Transformers)

BERT is an open-source natural language processing model developed by Google [23]. With this model, instead of processing the words individually, the entire sentence was considered as a whole, leading to better results [24].

The most important feature of BERT is that it is bidirectional. Other systems are unidirectional, meaning that words gain meaning by using terms on the left or right side of the text [25].Google Bert as a two-way it reveals the relationship between words and sentences on the basis of an artificial neural network [24]. In addition, while more data is needed during training in other systems, successful results can be obtained with less data because BERT is bidirectional [23] [25].

In this study, DistilBERT, which is the simplified version of BERT, was used. Although it has a transformer architecture similar to BERT, the transformer layer is 12 in BERT and 6 in DistilBERT. In addition, the token type placement and pooling layers that are included in BERT are also absent in DistilBERT. With this update, the model size has been reduced and the inference speed has increased, while the performance can be deteriorated by 2%-3% [26] [27].

2.4. Parameter Optimization

Afterwards, parameter optimization was performed with GridSearchCV (Grid Search Cross-Validation). In the process, all parameter options were given as input and the combinations with the best results were output. While conducting this analysis, a hundred records were sampled for each category in the dataset. In addition, the best parameter value was determined with 3-fold cross-validation in all classes. All the tested parameters and the list of the best results obtained are presented in Table 2.

Within the scope of the study, the results of 8 machine learning algorithms obtained with default parameters are listed in Table 3.

GridSearchCV is a method for setting parameters that systematically constructs and assesses a model for every combination of algorithm parameters within a specified grid [28]. GridSearchCV is provided by the scikit-learn framework. The instructions typically specify a glossary for storing the initial parameters to be examined, after which GridSearchCV performs all necessary model adjustments and identifies the optimal parameters [29].

2.5. Classification

In the classification process, analyzes were made with 8 different machine learning algorithms. The dataset was divided into 70% training and 30% test data, and accuracy values were obtained. Additionally, results were obtained using 10-fold cross-validation and compared.

Table 2. Tested parameters and best results obtained with GridSearchCV

Classifier	Tested parameter values	Best parameter values
Support Vector Machine (SVM)	C: [10.0, 1.0, 0.1], Coef0: [0.5, 0.0, 0.1], Gamma: [‘scale’, ‘auto’], Degree: [4, 3, 2], class_weight: [None, ‘balanced’], kernel: [‘linear’, ‘sigmoid’,’poly’, ‘rbf’]	C: 10.0 Coef0: 0.5, Gamma: scale, Degree: 4, class_weight: None, kernel: poly
K Nearest Neighbor (KNN)	n_neighbors: [7, 5, 3], algorithm: [‘ball_tree’, ‘kd_tree’, ‘brute’, ‘auto’], p: [1, 2], metric: [‘minkowski’, ‘manhattan’, ‘euclidean’] weights: [‘distance’,’uniform’] leaf_size: [30, 50]	n_neighbors: 3 algorithm: auto, p: 2, metric: minkowski, weights: distance, leaf_size: 30
Linear Discriminant Analysis (LDA)	solver: [‘svd’, ‘lsqr’, ‘eigen’], priors: [None, [0.3, 0.4, 0.3]], n_components: [None, 2, 3], tol: [1e-2, 1e-3, 1e-4], shrinkage: [0.1, None, ‘auto’]	solver: ‘svd’, priors: None, n_components: None, tol: 0.0001, shrinkage: None
Decision Tree (DT)	min_samples_leaf: [4,2,1], splitter: [‘best’, ‘random’], max_depth: [None, 30,20,10], class_weight: [None, ‘balanced’], criterion: [‘gini’, ‘entropy’], min_samples_split: [10,5,2], max_features: [‘log2’,‘auto’, ‘sqrt’]	min_samples_leaf: 1, splitter: best, max_depth: 20, class_weight: balanced, criterion: entropy, min_samples_split: 5, max_features: sqrt
Extra Tree (ET)	n_estimators: [50, 100, 200], criterion: [‘entropy’,‘gini’], min_samples_leaf: [4,2,1], bootstrap: [True, False], class_weight: [None, ‘balanced’], min_samples_split: [10,5,2], max_features: [‘auto’, ‘sqrt’, ‘log2’], max_depth: [None, 20,10],	n_estimators: 200, criterion: gini, min_samples_leaf : 1, bootstrap: false, class_weight:balanced, min_samples_split: 2, max_features: sqrt, max_depth : None
Random Forest (RF)	max_depth:[3,5,10,None], n_estimators:[10,100,200], min_samples_split:[3,2,1], min_samples_leaf:[3,2,1], max_features:[7,5,3,1]	max_depth: None, n_estimators: 200, min_samples_split: 2, min_samples_leaf: 1, max_features: 7
Naive Bayes (NB)		Priors: None
Logistic Regression (LR)	penalty: [‘l1’, ‘l2’], C: [10.0, 0.1, 1.0], solver: [‘saga’,‘liblinear’, ‘newton-cg’, ‘lbfgs’, ‘sag’], multi_class: [‘ovr’, ‘multinomial’], class_weight: [None, ‘balanced’] max_iter: [100, 200, 300]	penalty: l2, C: 10.0, solver: newton-cg, multi_class: ovr, class_weight: balanced max_iter: 100

3. RESULTS

In this research, crime category estimation was performed with machine learning (ML) algorithms after extracting the features with BERT. All results are given in Table 3. In this analysis, the dataset is grouped as 70%-30% training and testing data.

The values in Table 3 are the result of all machine-learning algorithms obtained with default parameters. As shown in Table 3, more than 99% accuracy was obtained with Linear Discriminant Analysis.

Additionally, the category classification accuracy rates of all classifiers except Decision Tree and Naive Bayes are also quite high with 97% and above.

After parameter optimization with GridSearchCV, the results are presented in Table 4. It was observed that either there was an increase in the obtained accuracy values or the same values were achieved as those obtained with the default parameters. Here, the best accuracy value was obtained with SVM and is 99.78%.

The fact that such high recognition rates have been achieved may suggest an overfit. In general, k-fold cross-validation is one of the methods used to check for overfitting. During this process, the dataset is split into K parts. During the training process, one layer at a time is used for testing and the remaining K-1 layer is used for training. This process is repeated K times. In other words, the model is tested in all subsets. Finally, all the values obtained are averaged. In this case, the model is tested on different samples.

In this study, 10-fold cross-validation was performed to test for overfitting. The results are given in Table 5. When we examine the results according to Table 5, it is seen that the results obtained in Table 4 are parallel to all classifiers. In addition, the highest recognition rates were obtained by SVM and Logistic Regression, as in the values obtained after parameter optimization. The results obtained in these two tables show that there is no overfitting situation in the model.

Table 3. Overview of all results (before parameter optimization)

Algorithms	**Accuracy value**	**Precision value**	**Re-call value**	**F1-measure value**
SVM	97.21%	97.19%	97.21%	97.16%
Logistic regression	98.71%	98.70%	98.71%	98.70%
K nearest neighbor	98.07%	98.09%	98.07%	98.06%
Decision tree	79.32%	79.31%	79.32%	79.27%
Extra tree	97.74%	97.75%	97.74%	97.71%
LDA	**99.30%**	**99.31%**	**99.30%**	**99.30%**
Naive bayes	88.23%	89.00%	88.23%	88.38%
Random forest	97.72%	97.73%	97.72%	97.69%

Table 4. Overview of all results (after parameter optimization)

Algorithms	**Accuracy value**	**Precision value**	**Re-call value**	**F1-measure value**
SVM	**99.78%**	**99.78%**	**99.78%**	**99.78%**
Logistic regression	**99.70%**	**99.70%**	**99.70%**	**99.70%**
K nearest neighbor	98.18%	98.20%	98.18%	98.18%
Decision tree	71.94%	72.29%	71.94%	72.05%
Extra tree	97.97%	97.98%	97.97%	97.94%
LDA	99.30%	99.31%	99.30%	99.30%
Naive bayes	88.23%	89.00%	88.23%	88.38%
Random forest	97.43%	97.45%	97.43%	97.39%

Table 5. 10 fold cross-validation (after parameter optimization)

Algorithm	**Accuracy**	**Algorithm**	**Accuracy**
SVM	**99.65%**	Extra Tree	97.94%
Logistic Regression	**99.65%**	LDA	99.27%
K Nearest Neighbor	98.14%	Naive Bayes	87.99%
Decision Tree	73.10%	Random Forest	97.35%

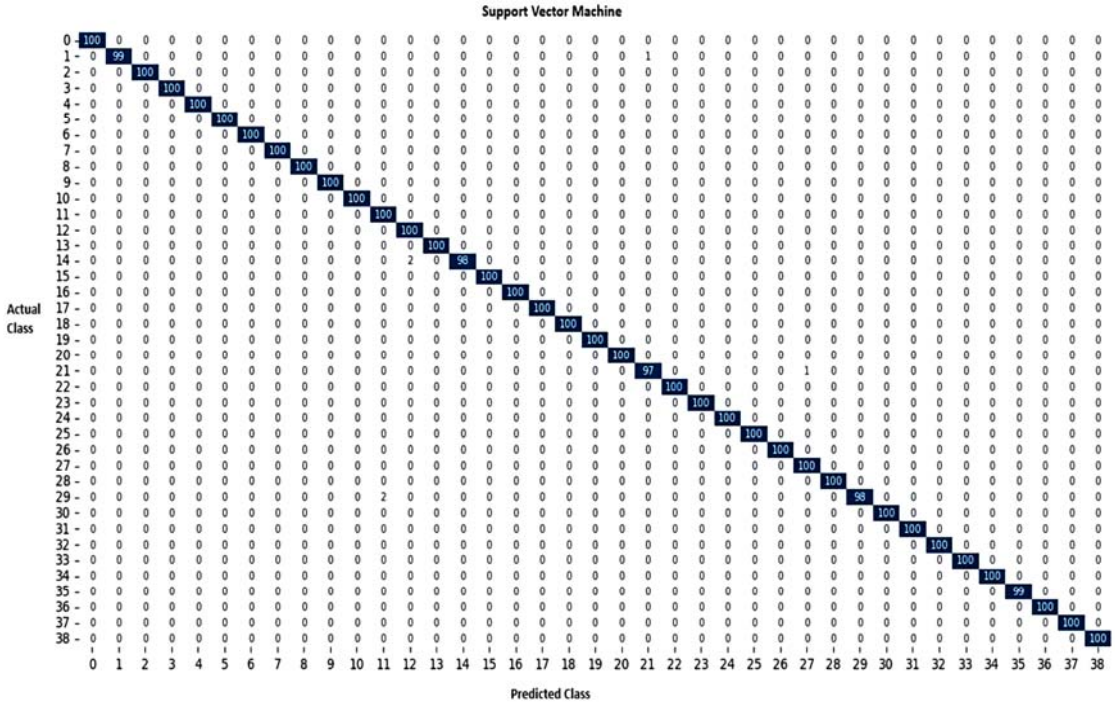

Figure 2. Confusion matrix for SVM(% values)

After parameter optimization, the confusion matrix for the SVM, which achieved the best accuracy values after both the 70%-30% train-test split and 10-fold cross-validation is shown in Figure 2.

The class labels for the confusion matrix are in the same order as those in Table 1. When examining the overall results for Logistic Regression on a class-based level, the distribution is generally balanced, with values of 97% and above. Only five categories out of the 39 showed accuracy values different from 100%. These categories are Assault, Gambling, Other Offenses, Sex Offenses Non-Forcible, and Vandalism.

Again, the ROC curve for SVM, where the best accuracy value is obtained, is shown in Figure 3. Since the data consisted of 39 classes, the ROC curve was shown as micro-average and macro-average and these two averages were compared. Both have an AUC value of 1.00 has been calculated as. In addition, the fact that both values remain at the top of the diagonal shows that a random prediction does not occur.

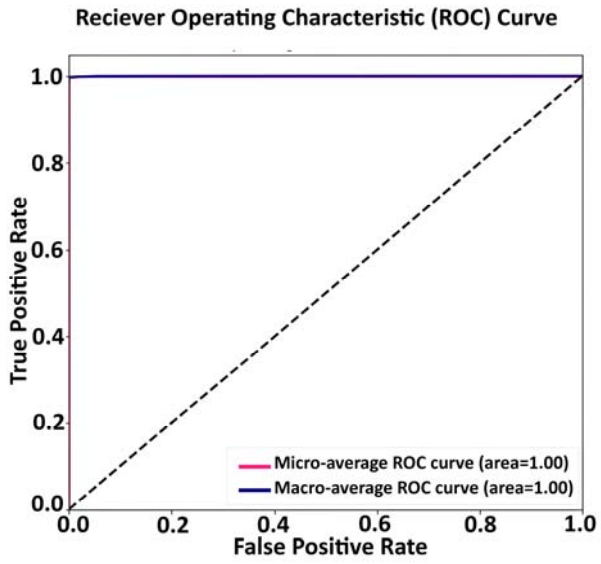

Figure 3. ROC curve for SVM

Figure 4 shows the Box Plot curve obtained by 10-fold cross-validation. According to this graph, the highest accuracy value is seen in SVM and Logistic Regression classifiers. The Decision Tree has the lowest accuracy value.

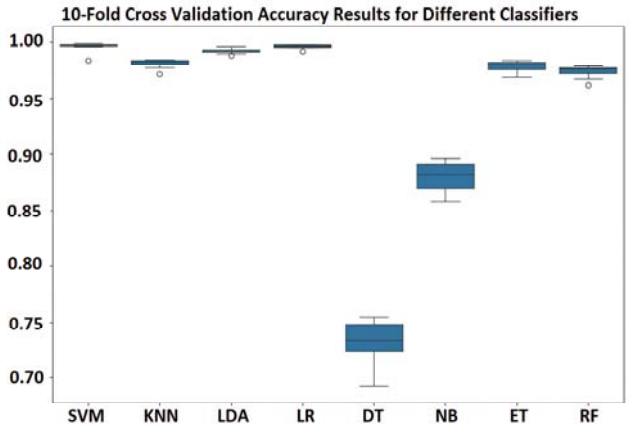

Figure 4. Box plot (for 10 fold cross-validation)

Figure 5 shows the training-test graph obtained with 10-fold cross-validation. According to this graph, the longest training time is seen in Logistic Regression. In other words, it can be called the most costly algorithm in terms of training time. It is also one of the two best algorithms in terms of accuracy value with 99.65%. The test durations are generally so short that they are not perceptible in the graph. SVM, on the other hand, is the longest algorithm in terms of test time. In SVM, training and testing times are close to each other. In fact, according to this graph, the most efficient algorithms can be interpreted as Naive Bayes and Nearest Neighbor. However, in terms of accuracy value, NB lagged behind. The same accuracy values were obtained at 10-fold cross-validation in SVM and LR. If we accept the result of the graph in Figure 5 as input, the best algorithm can be interpreted as SVM.

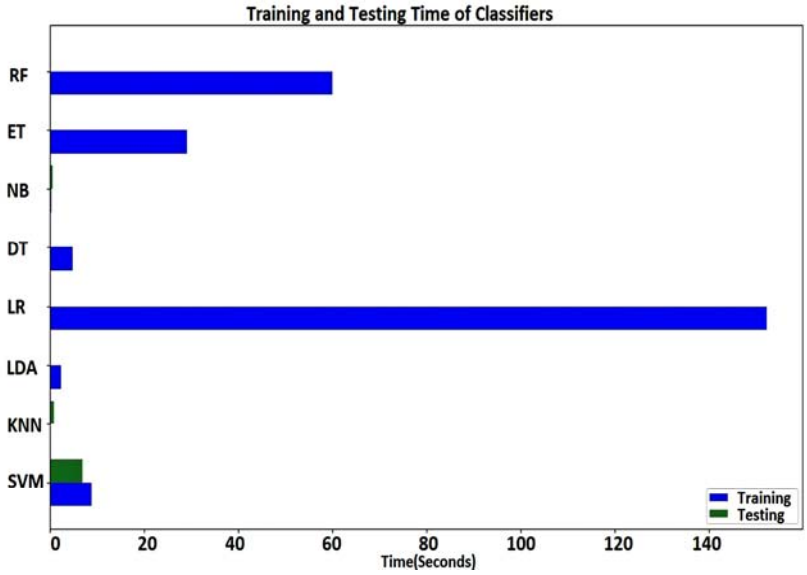

Figure 5. Training-test duration graph (for 10 fold cross-validation)

4. DISCUSSION

In Table 6, we have listed similar studies in the category of crime classification. In this comparison, we generally tried to prefer samples that used the same data set as our study. We have displayed the obtained information in terms of the dataset, the extraction of features, the classifier, and the breakdown of the best results achieved.

When we made a comparison based on the data set, it was seen that the first 10 or 15 categories in which the crime was committed the most were included in the studies in the applications using the same data set as our study. In the study of Chandrasekar et al., it was stated that 39 categories were included [13] and the results were not shared because they were not successful. In addition, in this study, an additional dataset containing demographic data was used to improve the dataset. In the other two studies with 39 classes, [10], and [12] it was seen that 2.611 and 2.39031 log losses were obtained, respectively. The difference in performance criteria in these two studies does not make it possible to compare in the evaluation process.

In the process of obtaining the features, the features from the dataset were generally preferred. Khan et al. [2] and Arslan et al. [4] obtained the features by Exploratory Data Analysis (EDA) and Doc2Vec, respectively. In our study, the features from the dataset were combined and used as input to DistilBERT, thus obtaining 768 features.

There are also studies where pre-processing processes such as data cleaning, data transformation and data reduction are applied. In our study, we tried different pre-processing methods and used the procedures in which we got positive results.

Within the scope of the review, it is seen that machine-learning algorithms are preferred as classifiers. In general, algorithms were used singularly and their results were evaluated. In our previous study [4] we developed a stacking ensemble model that includes eight ML algorithms. In our last study, we used machine-learning algorithms singularly and compared them. In addition, the highest accuracy values were generally obtained with the Random Forest and Decision Tree. In our study, the highest accuracy value was obtained with SVM.

The highest accuracy rate obtained on the same dataset (San Francisco Crime Dataset) within the scope of the review is seen in our previous study with 99.80% [4]. However, in this study, 15 criminal groups were involved in the process. In the last study, the best result was 99.78%, and 39 criminal groups were involved. In this case, it shows that the new model has better performance with more classes.

Table 6. Previous studies on crime analysis

Study	Dataset	Feature extraction	Classifier	Accuracy rate
Khan et al. [2]	San Francisco crime dataset (10 criminal groups selected)	Exploratory Data Analysis (EDA)	Gradient Boosting Decision Tree RF NB,	98.5% (2 classes)
Arslan et al. [4]	San Francisco crime dataset (15 criminal groups selected)	Doc2Vec	Stacking Ensemble (8 machine learning algorithms)	99.80%
Arslan et al. [14]	San Francisco crime dataset (15 criminal groups selected)	Available features in the dataset	Random Forest	86.5%
Arslan et al. [3]	49030 records of different crimes committed in New York State, 62 different categories of crimes, and 12 characteristics	Available features in the dataset	Decision Tree	99.9%
Bilen [16]	Elazig cybercrime dataset(6 criminal groups)	Available features in the dataset	Polynomial Regression	79%
Djon [20]	Chicago Crime Dataset(21 criminal groups)		XGBoost	86,0% (F1 score)
Pradhan [15]	San Francisco crime dataset(30 combined crime classes)	Available features in the dataset	Random Forest Naive Bayes LR KNN Decision Tree	2.3928 (log-lost)

Chandrasekar et al. [13]	San Francisco crime dataset (39 criminal groups selected) United States Census Data	Available features in the dataset	NB, RF SVC Gradient Boosted Decision Trees	96.3% (Blue/White Collar Crimes) 75.02% (Severe/Non-Violent Crime)
Abdulrahman et al. [10]	San Francisco crime dataset (39 criminal groups selected)	Available features in the dataset	KNN Naive Bayes	2.611 (log-loss)
Abouelnaga [12]	San Francisco crime dataset (39 criminal groups selected)	Available features in the dataset	KNN XGB Decision tree Bayesian Random Forest	2.39031 (log lost)
Recommended Method 2024	**San Francisco crime dataset (39 criminal groups selected)**	**DistilBERT**	**SVM LR KNN DT ET LDA NB, RF**	**99.78%**

5. CONCLUSIONS

In this study, the San Francisco dataset, managed by Kaggle Inc., was used. The dataset, which includes 39 classes, was utilized for modeling and was limited to a maximum of 1000 records. Different methods were tried in the pre-processing process and those that contributed positively to the process were preferred. Out of the nine variables in the dataset, eight were combined and used as input for DistilBERT. DistilBERT, a customized version of BERT, produced 768 features. Different machine learning algorithms were trained using these features, and their accuracy values were compared. Additionally, GridSearchCV was employed for parameter optimization. Analysis was carried out with the default parameters and the parameters obtained after optimization. Improvement was observed after parameter optimization. Among all the values compared, the SVM achieved the highest accuracy. Again, when we accept the training-test period as an input to the results obtained with 10-fold cross-validation, it is seen that the highest accuracy value is obtained with SVM.

6. REFERENCES

1. Dülgeroğlu, B., 2024. Suç kategori tespiti için istifleme topluluğu modeli kullanan sistem tasarımı. Yüksek Lisans Tezi, Kayseri Üniversitesi, Kayseri.
2. Khan, M., Azmat, A., Alharbi, Y., 2022. Predicting and preventing crime: a crime prediction model using san francisco crime data by classification techniques. Complexity, 2022(1), 4830411.
3. Horoz, A.D., Arslan, H., 2023. Crime analysis and forecasting using machine learning. Journal of Optimization and Decision Making, 2(2), 270-275.
4. Arslan, R.S., Dülgeroğlu, B., 2023. A design of crime category detection framework using stacking ensemble model. Çukurova Üniversitesi Mühendislik Fakültesi Dergisi, 38(4), 1035-1048.
5. Butt, U.M., Letchmunan, S., Hassan, F.H., Ali, M., Baqir, A., Sherazi, H.H.R., 2020. Spatio-temporal crime hotspot detection and prediction: a systematic literature review. IEEE Access, 8, 166553-166574.
6. Bharathi, S.T., Indrani, B., Prabakar, M.A., 2017. A supervised learning approach for criminal identification using similarity measures and K-Medoids clustering. In 2017 International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICICT), 646-653. IEEE.
7. Babakura, A., Sulaiman, M.N., Yusuf, M.A., 2014. Improved method of classification algorithms for crime prediction. In 2014 International Symposium on Biometrics and Security Technologies (ISBAST), 250-255. IEEE.

8. Baculo, M.J.C., Marzan, C.S., de Dios Bulos, R., Ruiz, C., 2017. Geospatial-temporal analysis and classification of criminal data in manila. In 2017 2nd IEEE International Conference on Computational Intelligence and Applications (ICCIA), 6-11. IEEE.

9. Borowik, G., Wawrzyniak, Z.M., Cichosz, P., 2018. Time series analysis for crime forecasting. In 2018 26th International Conference on Systems Engineering (ICSEng), 1-10. IEEE.

10. Abdulrahman, N., Abedalkhader, W., 2017. KNN classifier and Naive Bayse classifier for crime prediction in San Francisco context. International Journal of Database Management Systems, 9(4), 1-9.

11. Borges, J., Ziehr, D., Beigl, M., Cacho, N., Martins, A., Araujo, A., Bezerra, L., Geisler, S., 2018. Time-series features for predictive policing. In 2018 IEEE international smart cities conference (ISC2), 1-8. IEEE.

12. Yehya, A., 2016. San francisco crime classification. arXiv Preprint arXiv, 1607.03626.

13. Chandrasekar, A., Sunder, A., Kumar, P., 2015. Crime prediction and classification in San Francisco City.

14. Arslan, R.S., Dülgeroğlu, B., 2023. Crime classification using categorical feature engineering and machine learning. In 2023 International Ankara Congress on Multidisciplinary Studies-VI, 1-8.

15. Pradhan, I., 2018. Exploratory data analysis and crime prediction in San Francisco. San Jose State University, 2018.

16. Bilen, A., Özer, A.B., 2022. Regresyon yöntemlerine dayalı suç tespit analizi karşılaştırması Elazığ ili örneği. Fırat Üniversitesi Mühendislik Bilimleri Dergisi, 34(1), 115-121.

17. Sarzaeim, P., Mahmoud, Q.H., Azim, A., 2024. Experimental analysis of large language models in crime classification and prediction. In Proceedings of the Canadian Conference on Artificial Intelligence.

18. Selvakumari, S., Peter, V., 2024. Crime classification using GRU, CNN and autoencoder techniques. Educational Administration: Theory and Practice, 30(5), 2950-2964.

19. Bharath, R.R., Sulthan, H.K., Mingaz, R.M., Kumaravengatesh, S.N.A., 2024. Machine learning approach to crime analysis and forecasting for prediction and prevention. African Journal of Biological Sciences, 1300-1313.

20. Djon, D., Jhawar, J., Drumm, K., Tran, V., 2023. A comparative analysis of multiple methods for predicting a specific type of crime in the city of Chicago. arXiv Preprint arXiv, 2304.13464.

21. Butt, U.M., Letchmunan, S., Hassan, F.H., Koh, T.W., 2024. Leveraging transfer learning with deep learning for crime prediction. Plos One, 19(4), e0296486.

22. Kan, W., 2015. San Francisco crime classification. https://kaggle.com/competitions/sf-crime, Kaggle.

23. Özkan, M., Kar, G., 2022. Türkçe dilinde yazılan bilimsel metinlerin derin öğrenme tekniği uygulanarak çoklu sınıflandırılması. Mühendislik Bilimleri ve Tasarım Dergisi, 10(2), 504-519.

24. Sevli, O., Kemaloğlu, N., 2021. Olağandışı olaylar hakkındaki tweet'lerin gerçek ve gerçek dışı olarak google BERT modeli ile sınıflandırılması. Veri Bilimi, 4(1), 31-37.

25. Özkömürcü, H., 2021. Google Bert algoritması/Google Bert nedir? [Online]. Available: https://hozkomurcu.com/google-bert-algoritmasi-google-bert-nedir/, Access date: 06.2024.

26. Liu, W., Zhang, S., Zhou, L., Luo, N., Xu, M., 2024. A semi-supervised mixture model of visual language multitask for vehicle recognition. Applied Soft Computing, 159, 111619.

27. Sanh, V., Debut, L., Chaumond, J., Wolf, T., 2019. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. arXiv Preprint arXiv, 1910.01108.

28. Ranjan, G.S.K., Verma, A.K., Radhika, S., 2019. K-nearest neighbors and grid search cv based real time fault monitoring system for industries. In 2019 IEEE 5th International Conference for Convergence in Technology (I2CT), 1-5. IEEE.

29. Pirjatullah, Kartini, D., Nugrahadi, D.T., Muliadi, M., Farmadi, A., 2021. Hyperparameter tuning using GridsearchCV on the comparison of the activation function of the ELM method to the classification of pneumonia in toddlers. In 2021 4th International Conference of Computer and Informatics Engineering (IC2IE), 390-395. IEEE.