applied sciences

MDPI

Article

Predictive Fraud Analysis Applying the Fraud Triangle Theory through Data Mining Techniques

Marco Sánchez-Aguayo [1,*,†], Luis Urquiza-Aguiar [2,†] and José Estrada-Jiménez [2,†]

[1] Departamento de Informática y Ciencias de la Computación, Escuela Politécnica Nacional, Ladrón de Guevara E11-253, Quito 170517, Ecuador
[2] Departamento de Electrónica, Telecomunicaciones y Redes de Información, Escuela Politécnica Nacional, Ladrón de Guevara E11-253, Quito 170517, Ecuador; luis.urquiza@epn.edu.ec (L.U.-A.); jose.estrada@epn.edu.ec (J.E.-J.)
* Correspondence: marco.sanchez01@epn.edu.ec
† These authors contributed equally to this work.

Abstract: Fraud is increasingly common, and so are the losses caused by this phenomenon. There is, thus, an essential economic incentive to study this problem, particularly fraud prevention. One barrier complicating the research in this direction is the lack of public data sets that embed fraudulent activities. In addition, although efforts have been made to detect fraud using machine learning, such actions have not considered the component of human behavior when detecting fraud. We propose a mechanism to detect potential fraud by analyzing human behavior within a data set in this work. This approach combines a predefined topic model and a supervised classifier to generate an alert from the possible fraud-related text. Potential fraud would be detected based on a model built from such a classifier. As a result of this work, a synthetic fraud-related data set is made. Four topics associated with the vertices of the fraud triangle theory are unveiled when assessing different topic modeling techniques. After benchmarking topic modeling techniques and supervised and deep learning classifiers, we find that LDA, random forest, and CNN have the best performance in this scenario. The results of our work suggest that our approach is feasible in practice since several such models obtain an average AUC higher than 0.8. Namely, the fraud triangle theory combined with topic modeling and linear classifiers could provide a promising framework for predictive fraud analysis.

Keywords: fraud triangle; human behavior; topic modeling; data mining; text mining ; classification methods

Citation: Sánchez-Aguayo, M.; Urquiza-Aguiar, L.; Estrada-Jiménez, J. Predictive Fraud Analysis Applying the Fraud Triangle Theory through Data Mining Techniques. *Appl. Sci.* **2022**, *12*, 3382. https://doi.org/10.3390/app12073382

Academic Editor: Federico Divina

Received: 24 January 2022
Accepted: 23 March 2022
Published: 26 March 2022

1. Introduction

Fraud is a worldwide phenomenon that affects public and private organizations, including various illegal practices that involve intentional deception or misrepresentation. According to the Association of Certified Fraud Examiners (ACFE), fraud includes any intentional or deliberate act of depriving another of property or money by cunning, deception, or other unfair acts [1].

The 2020 PwC Global Economic Crime and Fraud Survey reports that 49% of respondents said their companies had been victims of fraud or economic crimes. Approximately 45% of respondents have experienced losses of less than one hundred thousand dollars; 30% have suffered losses between one hundred thousand and five million dollars; 6% have suffered losses between five million and fifty million dollars; and 3%, losses of more than fifty million dollars. This unveils a rising trend in costs caused by fraud. In organizations, 52% of cases are related to internal fraud and 41% to external. This gap is due to anyone in accounting, and financial activities are a potential risk factor for fraud [2].

The prevention of fraud could mitigate expenses related to its prosecution as well as the time and effort to detect fraud after its occurrence. When fraud is discovered, the opportunity to locate the perpetrator and recover the losses caused is scarce. Therefore,

organizations must identify factors that lead to fraudulent behaviors and understand their causes. When we look at people in a controlled environment, such as their workplace, we can more accurately identify suspicious behaviors since human behavior analysis is critical in the early identification of fraud [3].

From the psychological point of view, Donald R. Crassey proposed the fraud triangle theory (FTT) to explain the causes and committing of fraud, identifying the elements that lead the perpetrators to commit fraud. In particular, three elements are represented as the vertices of the triangle. The fraud triangle's vertices are incentives/pressures, opportunities, and attitudes/rationalization [4]. However, evidence of fraudulent activities in which communications related to this phenomenon are observed is incipient due to its critical and reserved nature, except for certain private and government entities that have access to this type of information. In this context, a valid option is to generate synthetic data sets, which, according to many experts, are the key to making machine learning within artificial intelligence faster and more precise in their predictions [5].For this investigation, a data set composed of 14,000 records balanced in two classes of fraud and non-fraud (7000×7000) was generated. Using data mining (DM) techniques, we identified patterns related to fraud and extracted relevant information. On the other hand, relying on text mining (TM) techniques, a subfield of data mining that handles textual data, provides structure to unstructured data. It analyzes it to generate new knowledge [6]. In this context, topic modeling is a widely used approach in TM that provides a comprehensive representation of a corpus by inferring latent content variables called topics. These patterns appear as categories or groups related to content in an unstructured text collection. Therefore, a topic analysis technique assigns a probability to a new text, a document belonging to a specific topic [7]. By calculating the probabilities that a document belongs to a topic, the analysis is performed using classification and deep learning methods to identify which technique is more compatible with topic modeling and efficiently identify phrases suspected of fraud.

To the best of our knowledge, research related to data mining for fraud prediction associated with the fraud triangle theory and its technological applicability is limited or incipient. Auditors detect fraud through the use of their experience, but human bias cannot be easily suppressed, and their reasoning tends to be subjective.

1.1. Contribution

The main contribution of this work is to propose a novel detector of suspicious behaviors related to the occurrence of fraud by analyzing human behavior using FTT leveraged on machine learning (ML) and deep learning (DL). Our detector combines a predefined topic model and a supervised classifier to alert a potential fraud-related text. In a nutshell, a new document is assigned to the topic of the predefined topic model. At a second step, the text within a topic is classified as a potential fraud-related document, using the topic's probability of the first stage.

We generated a balanced synthetic data set, which contains phrases related to fraud and phrases not related to fraud. More precisely, the suspicious phrases contain words that belong to a vertex of the fraud triangle (pressure, opportunity, and rationalization). On the other hand, non-fraudulent phrases have a general context that includes words unrelated to this problem. To build our novel detector, we have to do the following:

- Evaluate the performance of text mining techniques, such as latent dirichlet allocation (LDA), non-negative matrix factorization (NMF), and latent semantic analysis (LSA) in the fraud-related data set. The goal is to select the technique that provides an integral representation of the analyzed documents through clusters, i.e., topic, as separated.
- Once we select the appropriate topic analysis technique, we use the documents' probabilities on the assigned topic to determine if a text can be identified as being fraud related, using supervised machine learning models. For this purpose, we conduct experiments on seven classification methods, including logistic regression (LR), random forest (RF), gradient boosting (GB), Gaussian naive Bayes (GNB), decision tree (DT),

k-nearest neighbor (kN), and support vector machines (SVM), using the synthetically generated data set.

- Furthermore, we perform the same experiment using deep learning techniques, such as convolutional neural network (CNN), dense neural network (DNN), and long short-term memory (LSTM), to determine the performance's differences using receiver operating characteristic (ROC) curves based on the area under the curve (AUC) with the traditional ML classification methods. The goal is to show which technique is more compatible to work with topic modeling to detect suspicious behavior of fraud.

The rest of this paper is organized as follows: Section 1.2 presents a review of the literature in the area of fraud detection. Section 2 offers definitions of FTT, topic modeling, classification methods, and deep learning. Section 3 describes the data preparation and methodology used in this work. Next, Section 4 presents the experiment and the results. Finally, Section 5 presents the conclusions and future work.

1.2. Related Work

Few research papers integrate data mining techniques with analyzing human behavior via fraud triangle theory to identify possible fraud cases. The following studies were found in the literature that contributes to this topic in this context. In [8], the authors proposed a generic architectural model that considers the fraud triangle factors. In addition to traditional fraud audit, the human factor enhances the audit analysis since the transactions examined by an auditor can be differentiated and prioritized better. By distinguishing behaviors (suspicious and non-suspicious), it is possible to discover transactions that are part of a pattern that is not yet known and that would have been left undiscovered if only traditional means were used. Likewise, Carolyn Holton in [9] proposed the design of a detector of disgruntled communications mainly in email repositories through data mining techniques associated with the triangle of fraud theory to combat internal security risks. In these lines, Mieke Jans [10] focused on reducing the risk of internal fraud by combining the detection and prevention of fraud. Its analysis uses descriptive data mining techniques to identify whether an observation is fraudulent or not. In this investigation, the authors used the IFR^2 methodology [11] to reduce the risk of internal fraud, a framework that uses the fraud triangle theory to assess and minimize fraud opportunities.

Vimal Kumar [12] analyzed fraud in the banking sector, classifying the types and definitions of existing fraud mechanisms. He also listed and explained the different data mining techniques used by investigators to study fraud, taking into account the factors that cause it by using the fraud triangle model, relating the pressure, timing, and rationalization with this behavior. The author concluded that prevention is an indispensable requirement in the banking sector, and data mining techniques are essential to reduce fraud cases. Ravisankar [4] used the fraud triangle theory to identify the possible reasons for the increase in fraudulent activities in companies. The authors used the multilayer feed forward neural network (MLFF), support vector machines, group method of data handling (GMDH), genetic programming (GP), logistic regression (LR), and probabilistic neural network (PNN) to predict fraud in financial statements on a data set from 202 Chinese companies. Their results showed that PNN was the technique with the best performance, followed by GP.

Aside from the methodology proposed by Jans [11], some other frameworks for fraud detection have been proposed. Panigrahi [13] suggested the integration of an auditor knowledge base and the techniques in audit processes called "knowledge-driven internal fraud detection (KDIFD)" to help auditors in the discovery of internal financial fraud more efficiently by applying data mining techniques. Authors in [14] proposed a system based on an automated framework for fraud detection using intelligent agents, data fusion techniques, and various data mining techniques.

Works on revisions of data mining techniques and machine learning applied to fraud detection were also identified, as in the case of [15] in which the authors reviewed research works on the methods of data mining applied to financial fraud detection (FFD). In [16], the

authors classified, compared, and summarized fraud detection methods and techniques based on the mining of relevant data in published academic and industrial investigations. This work also highlighted the application of data mining in other related fields, such as epidemic detection, insider trading, intruder detection, money laundering, spam detection, and terrorist detection. In the same context, Wang et. al. [17] reviewed the literature on data structure algorithms. The authors provided a reference to optimize fraud detection models. Their objective was to contribute with public accountants to select data and data mining technologies suitable to detect fraud. Dhiya Al-Jumeily [18] compared existing systems for fraud detection and proposed the development of a new system that allows detecting potentially fraudulent applications. With this method, organizations have a good outlook on the authenticity of applicants' identities and online applications.

On the other hand, the problem of the lack of access to financial data for fraud investigation has been addressed by other works using simulation techniques; thus, the privacy concerns of accurate data are avoided. Lopez et al. presented in [19] three case studies related to financial transactions, where a method to generate synthetic data was offered, which can be used as part of the necessary input data for the research, development, and testing of fraud detection techniques. Similarly, [20] proposed a novel way to create synthetic data for fraud investigation by developing a simulation prepared with accurate data. In [21], simulation techniques aimed to recreate the behavior of fictitious clients.

All the reviewed works contribute to the detection of fraud, mainly in the banking sector, proposing reference frameworks, such as IFR^2 and even applications related to artificial intelligence. However, the fraud analysis focused on a semantic context trying to identify unusual patterns in a data set is still incipient. Moreover, the combination of text mining with the fraud triangle theory to categorize texts as being potentially fraud-related was not addressed in the previously mentioned articles. In this sense, no studies were identified with evidence of the use of data mining techniques, the application of fraud theories, and the corresponding analysis of human behavior to detect fraud, which means that there is a gap, and this is an appropriate field investigation.

2. Materials and Methods

This section briefly describes the fraud triangle theory, topic modeling strategy, classification methods, and validation methods.

2.1. Fraud Triangle Theory (FTT)

Fraud is considered a subset of internal threats, such as corruption, misappropriation of assets, and fraudulent statements, among others [22]. ACFE defines fraud as "the use of one's occupation for personal enrichment through the deliberate misuse or misapplication of the employing organization's resources or assets" [23]. There are two types of fraud: internal and external. Internal fraud covers a series of irregularities and illegal acts characterized by the scammers' intentional deception, leading to the misappropriation of a company's money and other essential resources. In the case of external fraud, this is commonly done in the financial statements, which are falsely presented in the reports [13]. The fraud triangle theory proposed by Donald R. Cressey comprehensively explains this phenomenon's occurrence. Cressey, a leading sociology expert, wrote a series of books on preventing this crime. The reasons for committing it could be summarized in the following three critical elements: perceived pressure, opportunity, and rationalization. This theory determines that all three parts must be consecutively present to suspect a desire to commit fraud. The pressure is what motivates the crime in the first place. For instance, the subject has some economic problems that he cannot solve by legitimate means, so he begins to consider carrying out an illegal act, such as stealing cash or forging financial statements, as a way to solve his problem [24]. The second element is the perceived opportunity, which defines how the person will commit the wrongful act. The person must see how he can use (abuse) their position of trust to solve their financial problems with a low perception of the risk of being discovered. Finally, the third component relates to the idea that the individual

can rationalize his dishonest actions. Most people who commit fraud do it for the first time and do not have a history of criminality. They see themselves, as normal, honest people who have come up with a series of situations. Consequently, the fraudster will justify his actions in a way that is acceptable [18]. The risk of committing fraud increases when there is a tight connection between pressure, opportunity, and rationalization.

2.2. Topic Modeling (TM)

TM is commonly applied to extract valuable knowledge when performing text mining. TM allows the identification of hidden semantic structures related to a particular "topic." TM analyzes collections of documents, where each document is represented as a mix of topics. In turn, a probability distribution over the words contained in the documents models each topic [25,26]. If a document is about a specific topic, the words related to that topic will be present more frequently than the others. For example, a chat message about the poor economic situation of a person (potentially related to the pressure component of the fraud triangle) may contain words such as "debts", "financial problems", and "late payments". Three unsupervised machine learning algorithms are commonly used to implement topic modeling: LSA, NMF, and LDA [27]. From the evaluation point of view of TM methods, the key metrics are perplexity [25] and coherence to select an adequate number of topics depending on the problem at hand. The perplexity value is a confusion metric and accounts for the level of "uncertainty" in a model's prediction result. In contrast, the coherence score indicates the level of semantic similarity between words on a topic [28]. In this sense, for this work, coherence provides a more decisive factor in parameter optimization, which is why this metric was chosen to analyze topics [29].

2.2.1. Latent Semantic Analysis (LSA)

LSA is a technique that allows us to create a vector representation of texts to create semantic content. Through this "vector" representation, LSA calculates the similarity between texts to choose the most accurately related words. LSA uses singular value decomposition (SVD) to reduce the vector space dimensions. LSA tries to capture the latent semantics in linear space [30]. The idea is to obtain vectors for each document so that we can use them to find similar words and similar documents [31]. LSA collects a large amount of text, divides it into documents, and then creates a matching matrix of terms and documents through SVD.

2.2.2. Non-Negative Matrix Factorization (NMF)

Provided a set of n documents, m unique words and k topics, NMF unveils the main hidden themes by decomposing the non-negative matrix of term-documents $D \in \mathbb{R}_{+}^{m \times n}$ in the product of two other matrices; one matrix $U \in \mathbb{R}_{+}^{m \times k}$ that represents the relationships between words and themes and matrix $V \in \mathbb{R}_{+}^{k \times n}$ encloses the topic–document information in the latent topic space (i.e., $D \approx UV$) [32]. NMF is a form of dimension reduction because the number of topics k is typically many orders of magnitude smaller than the number of words m and several documents n under consideration. Matrices U and V constitute the principal result of NMF, and the distribution of words and documents about the topics is the primary focus of interpretation [33].

2.2.3. Latent Dirichlet Allocation (LDA)

LDA is an unsupervised probabilistic generative model that allows finding the semantic structure of a corpus. LDA is based on the hierarchical Bayesian analysis of texts [34]. An LDA model considers several themes in a corpus and a document as a bag of words generated from these themes. In LDA, each document is modeled as a random mix of latent topics. In turn, each topic is characterized as a probability distribution over words; that is, each vocabulary word has a certain probability, where words with high probability are more associated with that topic than words with low probability [35].

A word is defined as a basic unit of discrete information, which will be part of a vocabulary that we can denote as $\{w_1, w_2, \ldots, w_V\}$. A document is a sequence of words represented by $\{w_1, w_2, \ldots, w_N\}$, where N denotes the number of words present in the document. A corpus $D = \{d_1, d_2, \ldots, d_M\}$ is a collection of documents that includes the texts on which the topic analysis is to be carried out, and M is the number of documents in the corpus. The K topics present in the corpus are represented by the vector β. The k topic (i.e., β_k) can be considered a distribution over the vocabulary. To the presence of the kth topic in a particular document d, we call it $\theta_{d:k}$. The assignments of a word n of a document d in a specific topic are denoted as $z_{d,n}$. Finally, the words observed in a document d are denoted as w_d, and particularly the n-th word of the document is denoted as $w_{d,n}$. More formally, Ref. [36] defined these dependencies in the generative process for LDA, which depicts the joint distribution of hidden and observable variables in the model, as can be seen in Equation (1).

$$p(\beta_{1:K}, \theta_{1:D}, w_{1:D}) = \prod_{k=1}^{K} p(\beta_k) \prod_{d=1}^{D} p(\theta_d) \left(\prod_{n=1}^{N} p(z_{d,n} \mid \theta_d) p\left(w_{d,n} \mid \beta_{1:K, z_{d,n}}\right) \right) \tag{1}$$

Figure 1 illustrates a probabilistic graphical model (PGM), where the conditional dependencies of the different variables involved in the generative process of the LDA algorithm are observed. The white nodes represent latent variables, such as the prevalence of each topic in a document, the assignment of each word in the document to a topic, and the topics themselves. The shaded node represents the unhidden and observable variable.

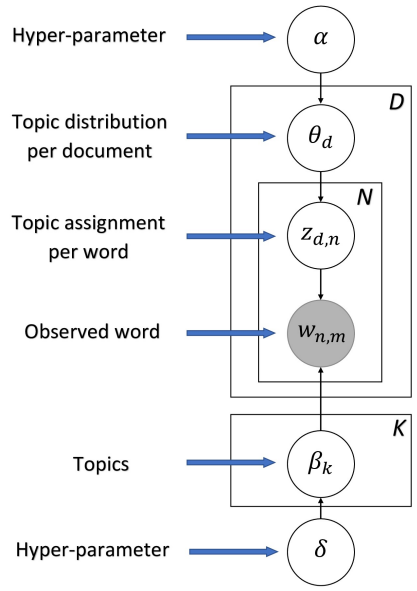

Figure 1. Representation of latent Dirichlet allocation LDA. Hidden nodes are not shaded and represent the proportions of topics, assignments, and topics.

The computational problem is to compute the conditional (posterior) distribution of the topic structure, according to Equation (2).

$$p(\beta_{1:K}, \theta_{1:D}, z_{1:D} | w_{1:D}) = \frac{p(\beta_{1:K}, \theta_{1:D}, z_{1:D}, w_{1:D})}{p(_{1:D})} \tag{2}$$

The numerator is the joint distribution of all the random variables, which can be easily computed for any hidden variables settings. The denominator is the marginal probability of the observations: the probability of seeing the observed corpus under any topic model.

2.3. Classification Methods

Supervised machine learning (ML) classifiers have several applications, including predictive data mining. These algorithms carry out the assignment of objects into labeled classes or categories of information. Classification is a supervised machine learning approach that consists of labeling data items as belonging to a particular class from a model built from a selected data set. In other words, a training data set is used to derive a model, which is then used in the new data sets to classify unseen test data [37]. In this context, the classifier observes a set of training samples and, following in them, can make predictions about the categorization of some other new samples presented.

Each classifier has an associated precision that will differ according to the type of data used. There are several evaluation metrics to compare the classification methods, and each of them could be useful depending on the kind of problem associated [38]. A receiver operating characteristic (ROC) chart is a technique for visualizing, organizing and selecting classifiers based on their performance. The area under this curve (AUC) is one of the most critical evaluation metrics that can be applied to choose the best method of classification [39]. In addition, AUC is among the most commonly used performance metric in the literature related to fraud detection. Thus, for comparability with other works, we decided to apply this metric to assess the models obtained in this work.

This work compares six well-established classification algorithms to detect fraud-related text within a topic, using the AUC criterion.

2.3.1. Logistic Regression (LR)

LR, also known as a logistic or logit method, analyzes the relationship between multiple independent variables and a categorical dependent variable, estimating the probability of occurrence of an event by fitting the data from a logistic curve.

2.3.2. k-Nearest Neighbor (kN)

The kN algorithm was developed from the need to perform discriminant analyzes with unknown parametric estimates of probability densities [40]. kN can classify unlabeled observations by assigning them to the class of the most similar labeled examples [41]. There are two essential factors related to this classifier. One is the method for calculating the distance between a sample and others belonging to the most frequent class among the closest training examples. In most cases, the kN implementation uses the Euclidean distance. The other factor is to decide how many neighbors (i.e., k) to choose in this algorithm.

2.3.3. Decision Tree (DT)

The DT algorithm solves classification and regression problems in the form of trees. DT can be updated incrementally by dividing the data set into smaller data sets (numerical and categorical), where the results are represented in the leaf nodes [42]. Decision trees are generally represented as a hierarchical structure that allows a more accessible interpretation than other methods; each internal node checks an attribute, while each branch corresponds to the value of the attribute or range of values [43].

2.3.4. Random Forest (RF)

RF is a classification method based on several decision trees, which is used to classify a new instance by majority vote. Each node in the decision tree uses a subset of attributes selected randomly from the entire original set of characteristics [44]. The correlation between trees decreases by randomly selecting the features that improve the predictability, and higher efficiency is obtained as a result [45].

2.3.5. Gaussian Naïve Bayes (GNB)

The GNB classifier applies the Bayes' theorem, assuming that all attributes are independent. Its main advantage is that it requires a small measure of training data vital for the characterization and necessary for classification [46]. The GNB classification is a case of the naive Bayes method, assuming that there is a Gaussian distribution on the attribute values, given the class label.

2.3.6. Gradient Boosting Decision Tree (GBDT)

The GBDT is based on the decision tree model. It builds the model through gradient augmentation, aiming to boost the combination of several weak and simple classifiers in a given set. This algorithm trains a new tree model that reduces the error of the whole set. To ensure that the loss function decreases continually in each iteration, the new tree model is built using the loss function's negative gradient [47]. Compared with linear regression models, GBDT can handle different types of variables (continuous, categorical, etc.) and requires little data preparation time [48].

2.3.7. Support Vector Machines (SVM)

Vapnik introduced SVM as a kernel-based machine learning model for classification and regression tasks. SVM classifier aims to find a linear hyperplane (decision boundary) that separates the data to maximize margin. For example, look at a problem of separating two classes in two dimensions [49,50]. SVM is a high-precision binary data classification technique, which has been widely used in various fields. Let $v = \{v_1, v_2, \ldots, v_m\}$, an m-dimensional input feature. We assume that each $v_i \in v$ is normalized to an interval $[0, n]$ using normalization techniques. Let $p = \{-1, +1\}$ be two different predictions, that is, negative and positive. An SVM classifier is a separating hyperplane with a maximum margin in the m-dimensional feature space, which divides the m-dimensional feature space into two subspaces, i.e., a subspace for positive prediction and the other for negative prediction [51].

2.4. Neural Networks

A neural network (NN), also known as an artificial neural network (ANN), allows non-linearity between the characteristic variables and the output signals [52]. A simple NN generally consists of an input layer, a hidden layer (s), and an output layer. The number of hidden and output layers is the neural network depth. The term deep learning refers to NN with considerable depth [53]. In this investigation, the input layer receives the information of the document probabilities. These belong to a specific topic; the output layer predicts the result, in this case, whether or not the sample is associated with cases of fraud.

2.4.1. Deep Learning (DL)

DL is a subfield of machine learning in artificial intelligence, based on algorithms that try to model high-level abstractions in data through the use of multiple layers of processing with complex structures or composed of multiple non-linear transformations [54]. TensorFlow, Keras, and PyTorch are the most used libraries for DL. For this work, we will use Keras, a high-level framework written in Python that provides a second-level abstraction, that is, instead of directly using the first-level frameworks (Theano, Torch, PyTorch, and Tensorflow). We can use a new framework over an existing one and thus further simplify the development of the deep learning model.

Dense Neural Networks

Dense neural networks (DNN) are also known as feed-forward networks because they avoid cycle formation. Determining the adequate number of neurons in hidden layers is a complicated issue (done by trial–error) since many neurons can result in overfitting problems. In contrast, a small number cannot learn from the data [55].

At the output of each layer, we have an activation function. In the next layer, the value of one of the neurons corresponds to the image of the values of the previous neurons, representing the non-linearity in a neural network. The output is composed of the selected activation functions, commonly non-linear ones, such as sigmoid, hyperbolic tangent, and ReLU, among others [56].

Convolutional Neural Networks

A convolutional neural network (CNN) is a deep learning algorithm that allows processing data with local patterns, which is very efficient for image classification [57]. It comprises an input layer, convolutional layers, and fully connected layers on top. Additionally, it uses tied weights, grouping layers, and an output layer. This architecture allows CNNs to take advantage of the 2D structure of the input data [54].

Long Short-Term Memory

Long short-term memory (LSTM) is an improvement of the recurrent neural network (RNN), which has the problem of the gradient's disappearance or explosion. LSTM blocks of memory are used instead of conventional RNN units to solve this problem. RNN sometimes fails to capture long-term dependency in a sequence. Therefore, short-term memory was invented to solve this problem by recursively applying a transition function to the input's hidden state, allowing to remember and connect the previous information to the current one [58].

LSTM retains a cell state C_t in the time interval t that allows it to learn the formed, stable sequential correlations. LSTM controls information flow through the entry gate, forgetting gate, and exit gate [59].

3. Methodology for Predicting Fraud based on the Fraud Triangle Components

Our objective is to build predictive models to enable early fraud detection. Thus, our strategy consists of identifying hidden patterns that might be related to one of the fraud triangle vertices from the fraud triangle theory. For this, we construct a model to predict whether a specific phrase belongs to one of these triangle categories. In this line, this strategy is a novel approach to fraud detection as long as it considers a new semantic view of this problem.

To detect suspicious patterns related to the vertices of the fraud triangle, we first perform topic modeling (unsupervised learning) over an unstructured text data set [60]. In particular, we select the best model obtained from LSA, NMF, and LDA.

Then, based on the coherence value, we determine the appropriate number of topics we can align with the fraud triangle theory; this involves obtaining the probabilities of documents belonging to a specific topic.

Based on such topics (or labels) and machine learning techniques, we categorize a sentence as potentially fraudulent if there is suspicion of it belonging to one of the vertices of the fraud triangle. This process is illustrated in the first flow chart of Figure 2.

Once the probability that a document belongs to a specific topic is calculated through topic modeling, a balanced data set is obtained with records labeled as fraud and non-fraud. This data set allows training learning models to predict potentially fraudulent behavior. This is illustrated in the second flow chart of Figure 2.

The performance of supervised learning methods applied over this data set is benchmarked to identify the best-performing one. Finally, the results obtained are analyzed to determine which technique is most compatible with topic analysis for fraud identification. More details on the process followed to implement this strategy are provided below.

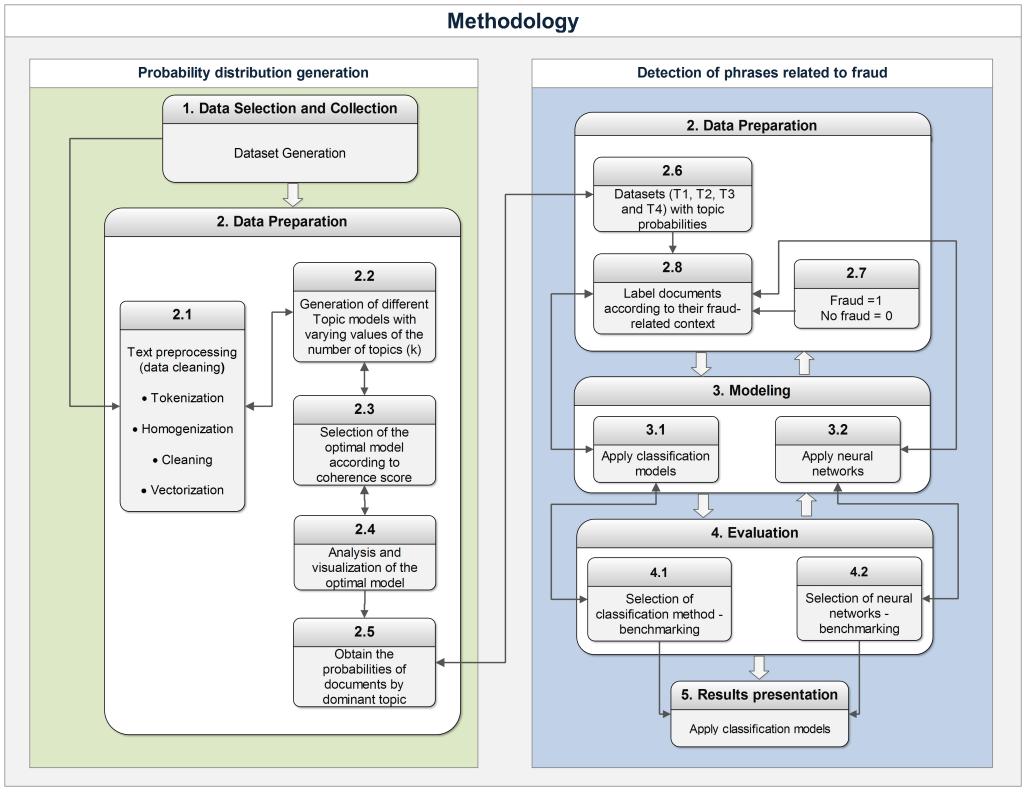

Figure 2. Methodology used to determine the existence of fraud.

3.1. Data Set Generation

Data sets involving fraud-related behavior are scarce due to several reasons, e.g., due to confidentiality policies of institutions, or due to the sensibility of the personal data included. Given the restricted access to this information, it is common to use synthetically generated data sets [61,62]. Our data set was created from a dictionary of fraud-related keywords that were purchased from the company [63]. These keywords are tagged into different categories, including pressure, rationalization, and opportunity, the three components of the fraud triangle theory. Starting from several of these keywords related to the fraud triangle theory and using different online tools to generate sentences, as in [64–66], the corresponding sentences including the selected keywords were obtained. These tools allow the generation of sentences based on a specific word with a well-defined grammatical and semantic structure. Finally, they use a web scraping tool, "Firefox Addon," which allows us to save the generated results and export them in CSV format for processing. The process followed to generate the data set is shown in Figure 3. Additionally, following the same procedure, several documents not related to fraud were generated in the same proportion as those related to fraud, with the only difference being that, for this case, keywords not related to this phenomenon were chosen, thus obtaining a balanced data set.

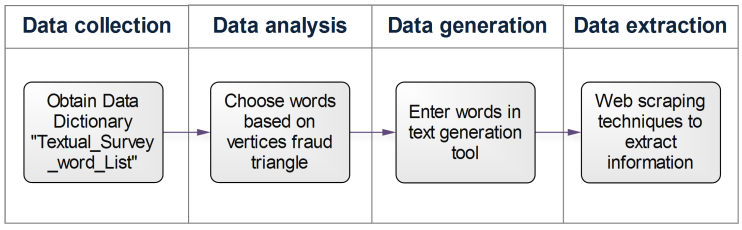

Figure 3. Flow diagram used for the generation of a synthetic dataset.

The next step is to analyze the data and identify their characteristics to classify their main parameters, which must be retained for the early detection of suspicious behavior related to fraud.

3.2. Data Preprocessing

The raw representation of the data set needs to be changed to be more suitable for topic modeling. With this aim, we use NLTK (Natural Language Toolkit), a Python library that implements some of these preprocessing phases: sequential tokenization, homogenization, cleaning, and vectorization [67]. Next, we describe them.

3.2.1. Tokenization

Tokenization is key for text processing. It consists of changing the representation of the data set so that it can be more easily processed. With this aim, tokenization involves dividing a document into its words (tokens). This was implemented using Python through the `word_tokenize` function [68] from the nltk.tokenize package.

3.2.2. Homogenization

Homogenization entails adapting the data set by eliminating certain parts of words and sentences that do not contribute to the semantic analysis of the text. Some of these activities are described below.

a . Change all tokens to lower case. This is implemented by the Python `lower` function.
b . Remove non-alphanumeric items. To identify non-alphanumeric characters, we use the Python `isalnum` function.
c . Obtain the word lexeme (lemmatization). Lemmatization turns a words into their lemma/lexeme form (for example, "runs", "running" and "ran" are all forms of the word run, and therefore "run" is the lemma of all these words). When obtaining lexemes, word sets are uniquely represented. In this way, the semantic meaning of the words is associated with the same lexeme. For this, we use the `lemmatize` [69] function of WordNetLemmatizer from NLTK.

3.2.3. Cleaning

It is essential to mention that there will be sets of words that do not add semantic value to documents. The cleaning process is based on eliminating less relevant words, that is, those that provide less information. For example, articles, prepositions, or conjunctions are words of little relevance. The `stopwords list` function [70] provided by NLTK is used to identify these words and to remove them then.

3.2.4. Vectorization

Vectorization entails obtaining a numerical representation of the words or phrases of a data set. Vectorization aims to extract more useful information when processing natural language text, e.g., through LDA.

LDA topic modeling requires vectorized documents. To implement vectorization, we use the `gensim` library. The dictionary function belonging to this library allows building a dictionary containing all the tokens that appear in the corpus and assigning them an identifier. We use both this dictionary and the function `doc2bowse` [71] that converts documents to a word bag representation. The corpus is constructed in the format necessary to carry out topic modeling through algorithms that implement LDA.

3.3. Quantitative Evaluation of Topic Modeling Algorithms

The most relevant and used topic modeling methods are LSA, NMF, and LDA. In related research, it was observed that the effectiveness of these algorithms differs in terms of the amount and type of data to be processed. In most cases and particularly for large data sets, LDA proved to be more efficient than other methods when identifying coherent topics [72,73]. In more specific cases, NMF outperformed the others [74]. In general, NMF

and LDA are similar, but LDA seems to be more consistent [75]. Although the use of LDA has become popular when handling big unstructured data, selecting the best option for topic modeling might depend on the particular data being processed. Consequently, benchmarking the efficiency of these algorithms in this context is required. First, we identify the appropriate number of topics based on the resulting coherence value of each model. This approach enables us to analyze the performance of the topic mentioned above modeling algorithms and, in particular, to identify the one that more concisely and coherently learns such topics. Having identified the k parameter (number of topics) for models obtained from LDA, NMF, and LSA, we select the algorithm offering the highest coherence value, which identifies the end of the rapid growth of coherence between topics, thus offering meaningful and interpretable topics. From this analysis, a quantitative evaluation of the topic modeling techniques is carried out, which consists of measuring the coherence of the topic C_v over a model's topic and topic–article assignment output, which will indicate an approximate measure of the quality of that result.

3.4. Selection of the Topic Modeling Algorithm

After obtaining topic models based on LDA, NMF, and LSI from the data set, we evaluate the consistency of the sets of words generated by each and determine the efficiency of classifying them into a specific topic. Each topic groups the most representative words for a given subject. We compare the sets of words obtained by the models and their distribution, prevalence, and structure. This analysis enables us to find the method that more accurately identifies the data set's topics. Once the model with the best performance is identified, it is checked whether its value $k = 9$ is the most appropriate. If this is the case and non-overlapping topics are found, that is, a coherent structure of particular topics, this would be an appropriate value of k. Otherwise, we will manually identify another value of k that meets a suitable distribution of topics based on visual inspection of the topics found.

Afterward, we obtain the modeling of the topics corresponding to this k value. Here, we analyze the distribution of words corresponding to each topic and identify the words related to fraud that are more representative or dominant; the objective is to find a relation between the context of each topic and the vertices of the fraud triangle. Then, from the LDA model, we obtain the probabilities that the documents in the data set belong to a specific topic. Each of these probabilities may represent a metric to categorize a document as being potentially related to fraud. Nevertheless, such probabilities themselves also serve as an interesting new representation of the data set. From this representation, we extract smaller data sets, each of which groups documents associated with a (dominant) topic, i.e., a topic to which the documents belong with the highest probability.

3.5. Methodology of Evaluation

Because the data set was synthetically generated, it is possible to identify, a priori, fraud-related phrases, label them accordingly, and then show how accurate a classification method is when predicting fraud activities. It is necessary to identify which model best fits the analysis of topics in the context of the data set, its size, and characteristics. When analyzing the performance of traditional machine learning and deep learning models, traditional classifiers can generally learn better than deep learning classifiers if the data set is small. On the other hand, deep learning models might obtain a performance boost when working over larger data sets. We evaluate both approaches since the intrinsic characteristics of a data set could affect their performance.

Once behavior patterns related to fraud are identified through the topic analysis and probability distributions generated, we have a data set that can be analyzed using classification methods and neural networks. We aim to evaluate how accurate the prediction of these models turns out to be. The most common classification and deep learning methods are used to identify which alternatives have better performance.

The analysis of both techniques is carried out using the ROC curve graph; this allows us to visualize, organize and select classifiers based on their performance using the AUC parameter that depicts the quality of classification methods.

4. Results and Discussion

This section presents the results obtained from testing our fraud detection mechanism in a case study. From our view, such results show its effectiveness. Details on data collection and processing are provided, followed by experiments on supervised and unsupervised model learning and the analysis of such results. Finally, the practical implications of the method and the findings are discussed.

4.1. Probability Distribution Generation

In this first scenario, we present the results from analyzing our synthetic data set to find patterns related to the fraud triangle theory, which is the proxy we use to detect potential fraud-related behaviors.

4.1.1. Optimal number of Topics

When topic modeling is used, it is essential to determine the number of topics (k) that best capture the trends in potentially fraudulent messages. We constructed several models based on LSA, NMF, and LDA with different values of k, and those with the highest coherence score were selected. Choosing several k topics associated with the maximum resulting coherence generally offers the most appropriate topics.

To obtain the coherence value of different models, `scikit-learn` and `gensim` Python libraries were used. Gensim does not have an implementation of NMF, so it was used only to implement LDA and LSA. `scikit-learn` offers a solution for NMF, allowing it to obtain the required coherence value.

The coherence validation for the different numbers of topics is shown in Figure 4. In the three models (LSA, NMF, and LDA), we can observe that the coherence value increases as the number of topics increases, demonstrating that the patterns in data are better captured with a higher number of topics. For the three models, the coherence value gradually increases to a certain k. For LSA, we obtained the highest coherence value when $k = 4$. For NMF, $k = 8$ resulted in a coherence value of 0.9143. LDA obtained the highest coherence value (0.6164) for $k = 9$. For higher values of k, for all three cases, coherence fluctuates indeterminately. This result implies that establishing a higher number of topics does not necessarily imply better performance. Instead, the time necessary for their calculation increases.

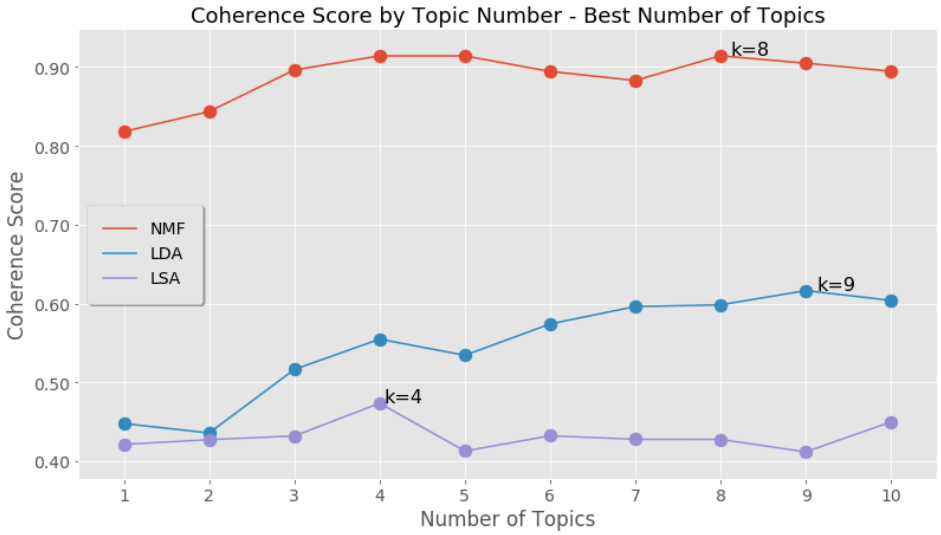

Figure 4. Comparing the techniques (LSA, NMF and LDA)—highest coherence score.

NMF showed the highest coherence score, followed by LDA and LSA, respectively. NMF classified large numbers of phrases on a specific topic. On the other hand, LDA was able to better distribute phrases along with all 9 topics, according to Table 1.

Table 1. Highest values of coherence obtained from the 3 models.

	Models		
	LSA	**NMF**	**LDA**
Coherence Values	0.4735	0.9143	0.6164

To analyze the behavior in the distribution of the topics, based on the values of k we tested, Tables 2–4 show the 10 most relevant keywords of the analyzed models associated with their related topic. Four topics were discovered using LSA, and some of the words grouped in topics overlap; this is because the dimension of latent themes depends on the range of the corresponding matrix (see Section 2), and this limit is exceeded. Additionally, LSA cannot capture the different meanings of words, offering less precision when distributing words in each topic. Table 2 depicts how words are clustered into topics (context) when using LSA and illustrates how some words are repeated for some topics, which is evidence of the problem of capturing the meaning of words. We use words in color to visually represent this phenomenon.

Table 2. Collection of topics and the top 10 keywords of the corresponding topic represented by the LSA model.

LSA			
T1	**T2**	**T3**	**T4**
problem	debt	be	job
economic	public	scare	lose
debt	problem	job	be
social	economic	lose	scare
political	country	go	ill
face	private	know	would
solve	service	get	scared
country	include	care	want
serious	reduction	think	work
issue	stock	people	earning
people	total	deserve	get

In the case of NMF, the overlapping of words along topics is also evident, as depicted in Table 3. Since more topics are involved with NMF, the repetition of words would have a less negative impact than with LSA. Note that the repetition of words may also be due to a too high value of k.

Finally, as shown in Table 4, the LDA model best groups words in topics since none of such words are repeated; this might entail a more consistent distribution of words along topics.

Since LDA behaves better on topic modeling in this particular context, we next evaluate this algorithm to detect potential fraud activities.

4.1.2. Application of LDA model

The number k of topics is an input parameter to obtain an LDA topic model. Determining the adequate value of k is critical for the model's performance. For our particular scenario (fraud detection using the fraud triangle theory), intuitively, the ideal number of topics embedded in the data set is 3, corresponding to the vertices of the fraud triangle (pressure, opportunity, and reasoning). However, from the coherence analysis described previously, 9 is the excellent value of k.

Table 3. Collection of topics and the top 10 keywords of the corresponding topic represented by the NMF model.

NMF							
T1	**T2**	**T3**	**T4**	**T5**	**T6**	**T7**	**T8**
debt	economic	tom	system	scared	review	job	easily
public	problem	mary	failure	people	period	lose	accessible
external	problem	big	error	know	currently	get	hotel
countries	social	think	file	got	keep	want	public
sustainability	political	want	data	really	kept	temporary	transport
private	issue	know	case	something	matter	steal	information
restructuring	serious	going	power	think	committee	work	car
total	countries	told	due	away	earnings	deserve	bus
reduction	people	help	event	look	countries	going	foot
management	country	thought	computer	get	board	need	city

Table 4. Collection of topics and the top 10 keywords of the corresponding topic represented by the LDA model.

LDA								
T1	**T2**	**T3**	**T4**	**T5**	**T6**	**T7**	**T8**	**T9**
steal	review	poor	want	people	big	make	economic	problem
later	think	child	deadline	know	use	care	weakness	debt
support	time	need	failure	evacuation	exploitation	job	ill	fair
say	fix	inadequate	year	deserve	right	work	life	abuse
just	help	insufficient	temporary	unethical	labor	compensation	leave	easily
tell	come	country	day	issue	family	lose	feel	accessible
woman	look	supervision	man	cause	friend	good	face	case
live	scare	really	old	situation	different	earning	thing	car
currently	like	money	ask	away	girl	way	great	information
period	world	school	change	abuse	hope	new	social	food

Such overlapping could also be analyzed through an intertopic distance map, e.g., that provided by the `pyLDAvis` Python library. `pyLDAvis` depicts an interactive, visual representation of an LDA model, through bubbles that represent the topics in a semantic topic space. Then, the closer the bubbles are to each other, the more semantic similarity they share. This map facilitates the understanding of the topic–term relationships in an adjusted LDA model and offers additional information about other perspectives on the applied model [76].

We tested the intertopic distance map in Figure 5 for different values of k, and we found an evident overlapping of topics for $k = 9$. In contrast, for $k = 4$, this representation depicted in Figure 5b showed topics adequately separated from each other. These four topics would be in line with the categories associated with the three components of the fraud triangle plus a fourth topic grouping other words.

To validate that $k = 4$ is the number of topics that best behaves according to the manual test described above, we use Algorithm 1 to adjust the hyperparameters (Dirichlet alpha and beta) of the LDA model. After testing different hyperparameters, we find that with the alpha and beta values of 0.91 and 0.31, respectively, $k = 4$ is obtained, which is the value for which the LDA model obtains the highest coherence of 0.5713.

Once the LDA model is obtained from the data set, words are manually labeled with the four resulting topics, according to the context of fraud, such as pressure, opportunity, rationalization, or others. This categorization is graphically depicted in Table 5. Labeling topics makes it possible to interpret the corpus and identify the theme implicit in this data

set. The interpretation of a topic can be achieved by examining a ranked list of the terms in each topic [77].

(**a**) (**b**)

Figure 5. Intertopic distance map for $k = 9$ and $k = 4$. (**a**) 9 topics. (**b**) 4 topics.

Algorithm 1 Algorithm to find the value of k that maximizes the coherence of an LDA model by testing different values of hyperparameters.

Require: Function ccv that compute coherence values
Input: min_topics =4, max_topics = 10, step_size = 1
Output: Csv format file containing results

Initialization α = [0.01, 0.31, 0.61, 0.91, 'symmetric', 'asymmetric']
Initialization β = [0.01, 0.31, 0.61, 0.91, 'symmetric']
TR = range(min_topics, max_topics, step_size)
for $k \in TR$ **do**
  for $a \in \alpha$ **do**
    for $b \in \beta$ **do**
      $cv = ccv(corpus, id2word, t, a, b)$ {finding the}
      $model\_results['\text{Topics}'].append(k)$
      $model\_results['\text{V.Alpha}'].append(a)$
      $model\_results['\text{V.Beta}'].append(b)$
      $model\_results['\text{Coherence}'].append(cv)$
    end for
  end for
end for

To illustrate how the words from the data set are distributed along with the four topics. We organize such words by topic and prevalence in Table 5. In addition, since each word related to fraud in our data set is originally labeled with its corresponding vertex from the fraud triangle, we color each word in Table 5 according to such a vertex. Words unrelated to the vertices are not colored. We can see that topics obtained from the LDA model might not reflect the vertices of the fraud triangle since the words within each topic are distributed through different components of the triangle.

Although the LDA model may not cluster words in topics following the components of the fraud triangle, the probabilities that phrases belong to such topics, provided by the model, are helpful to feed a classification algorithm to detect a fraud-related phrase.

Table 5. Most prevalent words from each topic related to the fraud triangle in our data set. Words are colored orange, blue, and green, representing the vertices pressure, rationalization, and opportunity, respectively.

Topics			
T1	**T2**	**T3**	**T4**
review	debt	problem	want
care	think	economic	know
poor	later	make	job
steal	fix	big	work
temporary	just	people	lose
say	tell	abuse	support
new	inadequate	fair	deadline
man	look	compensation	help
really	failure	child	come
insufficient	weakness	good	time
state	ill	earning	exploitation
money	unethical	easily	deserve
issue	life	accessible	scare
evacuation	world	country	right
leave	try	need	like
woman	let	way	day
year	talk	pay	use
long	old	school	scared
change	feel	home	ask
period	place	thing	car

4.2. Detection of Phrases Related to Fraud

To detect fraud, we represent our original data set with the probabilities that the documents belong to each topic (obtained from the LDA model). We also labeled each record with 1 or 0 to indicate whether it is related or unrelated to fraud, respectively. This new representation of the data set was used as input for different classification algorithms whose models could be used to detect fraud-related documents.

We specifically selected the documents grouped in each topic and its fraud-related/fraud-unrelated flag to build corresponding data sets (T1, T2, T3, and T4) that served as input for several classification algorithms.

Next, we discuss the process of building such classification models and the results of assessing them.

4.2.1. Classification Algorithms

From the previously described data sets, we built classification models to unveil the trends that would enable us to say whether a new document is related or unrelated to fraud. We tested several classification algorithms to reveal which of them performs better with this particular set of data and, in general, if our approach to detect fraud would be feasible in practice.

The use and selection of an adequate classification method are directly related to the information's characteristics. Within the spectrum and analysis of classifiers, the distinction between linear and non-linear models was made, taking into account the characteristics of each of these and the nature and quantity of the data. Specific differences between these two concepts can be mentioned. The linear ones are simple and easy to handle, and the fact that they have low computational consumption makes them ideal for use in topics such as automatic text classification. On the other hand, the non-linear ones directly related to neural networks assign data in higher-dimensional spaces [78].

4.2.2. Comparison of Classification Models

Depending on the information involved, learning algorithms may behave differently. Thus, we next comment on how these algorithms perform for the specific scenario proposed in this work. The process we followed for such evaluation is described in the following tasks:

- We preprocessed the information by dominant topic, importing the LDA data, and labeling the documents, to later be transformed into CSV format.
- Training was carried out after selecting a portion of data for testing (20%) and another for training (80%). The data set was divided into four subsets, where the first was used to train the algorithm with the corresponding attributes; the second was used to test the attributes. The third is made up of the labels related to the training set, and the fourth contains the labels corresponding to the test set.
- Finally, we evaluated and compared different classifiers (linear and no-linear algorithms vs. neural networks).

4.2.3. Classifier Performance

To benchmark these different classifiers, choosing a corresponding metric is critical. For this work, we selected AUC since it is very popular and adequate when we care about ranking predictions and not necessarily about obtaining well-calibrated probabilities [79]. Particularly, if classes are balanced, and there is no certainty that the classifier chose the best decision threshold, it is best to select AUC, which is equivalent to the probability that the classifier will assign the highest score to the relevant classes compared to the irrelevant ones [80]. As described in Section 2, ROC is a curve that represents the true positive rate vs. the false positive rate, where the area determines the performance of the model under such a curve. The closer the AUC score is to 1, the better the model distinguishes between classes. On the other hand, if it is closer to 0.5, the model performs just as well as a coin toss.

For our work, we use the ROC curve to depict the performance of different machine learning models when classifying documents as being related or unrelated to fraud. The results can be seen in Figure 6, but are also presented in Table 6.

Table 6. Performance, measured with AUC, of different machine learning models when classifying a document as related or unrelated to fraud. T1, T2, T3, and T4, correspond to each data set, where a topic learned from LDA is dominant.

Classification Method's	**Predictive Accuracy**				**Mean**
	T1	**T2**	**T3**	**T4**	
Logistic Regression: AUC	0.83	0.64	0.68	0.65	0.70
Random Forest: AUC	0.88	0.77	0.80	0.79	**0.81**
GNB: AUC	0.86	0.70	0.74	0.73	0.76
Gradient Boosting: AUC	0.89	0.77	0.79	0.79	**0.81**
k-NN: AUC	0.86	0.72	0.76	0.74	0.77
Decision Tree: AUC	0.80	0.71	0.73	0.75	0.74
SVM: AUC	0.86	0.70	0.75	0.74	0.76

These results show that random forest and gradient boosting obtain the best performance with a mean AUC of 0.81. Interestingly, *k*-nearest neighbors, GNB, and SVM also perform well with a mean AUC higher than 0.75. These results suggest that our approach to detecting activity related to fraud, based on identifying topics with LDA, might be feasible in practice when building machine learning models.

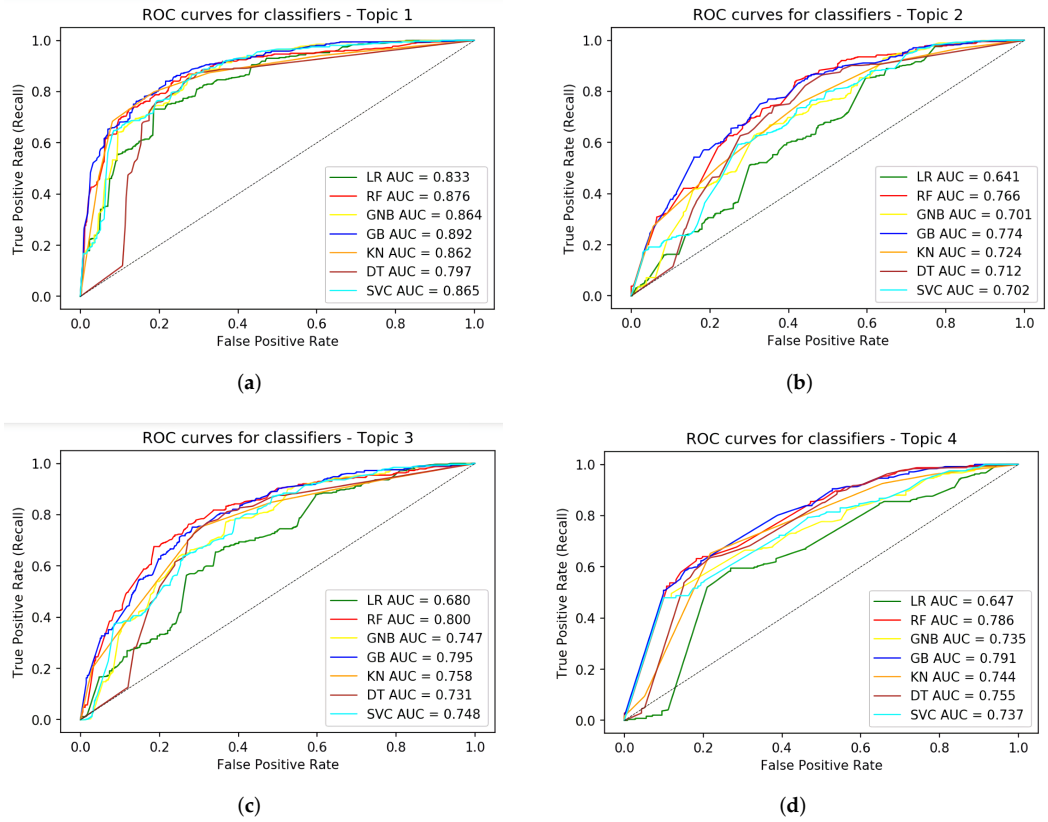

Figure 6. ROC curves of different classifiers for the data sets related to the dominant topics. SVC is the function in Scikit-learn, to implement SVM. (**a**) Topic 1. (**b**) Topic 2. (**c**) Topic 3. (**d**) Topic 4.

4.2.4. Deep Learning

Given their popularity and power, we also assessed deep learning models when classifying documents as related or unrelated to fraud. We tested the dense neural network (DNN), convolution neural network (CNN), and long short-term memory (LSTM). As with classical machine learning classifiers, we used the ROC curve as the performance metric. The results of measuring such performance when using neural networks are illustrated in Figure 7.

The best performing DNN has three layers and achieves an average accuracy of approximately 68% for the four topics analyzed. A sequential model was used because the network consists of a linear stack of layers. We represent the input layer that implements the activation function and the number of input dimensions that the network will have; there are 10 predictors in our case. This process is then repeated for hidden layers but omits the input parameter. The activation function used is a rectified linear unit or ReLU, which is the most used activation function because it is not linear and cannot activate all neurons simultaneously. We created the output layer with two nodes because two output classes, 0 and 1, correspond to being related to fraud and unrelated to fraud.

A one-dimensional CNN was also configured, including filters and a convolution operator to reduce the parameters. It did not offer adequate performance for classification, reaching an average precision of about 69% for the topics analyzed. The recurring network did not achieve the same level of precision as simple dense networks.

Finally, the best performing LSTM network was a two-layer network with 64 hidden drives. Its accuracy was about 67%. LSTMs exceed this average when information must be stored for an extended period.

In any case, the performance reached by deep learning models is lower than that of machine learning classifiers. This might not be the case if more data are involved in our

scenario since deep learning is known to perform much better when models are built from big data.

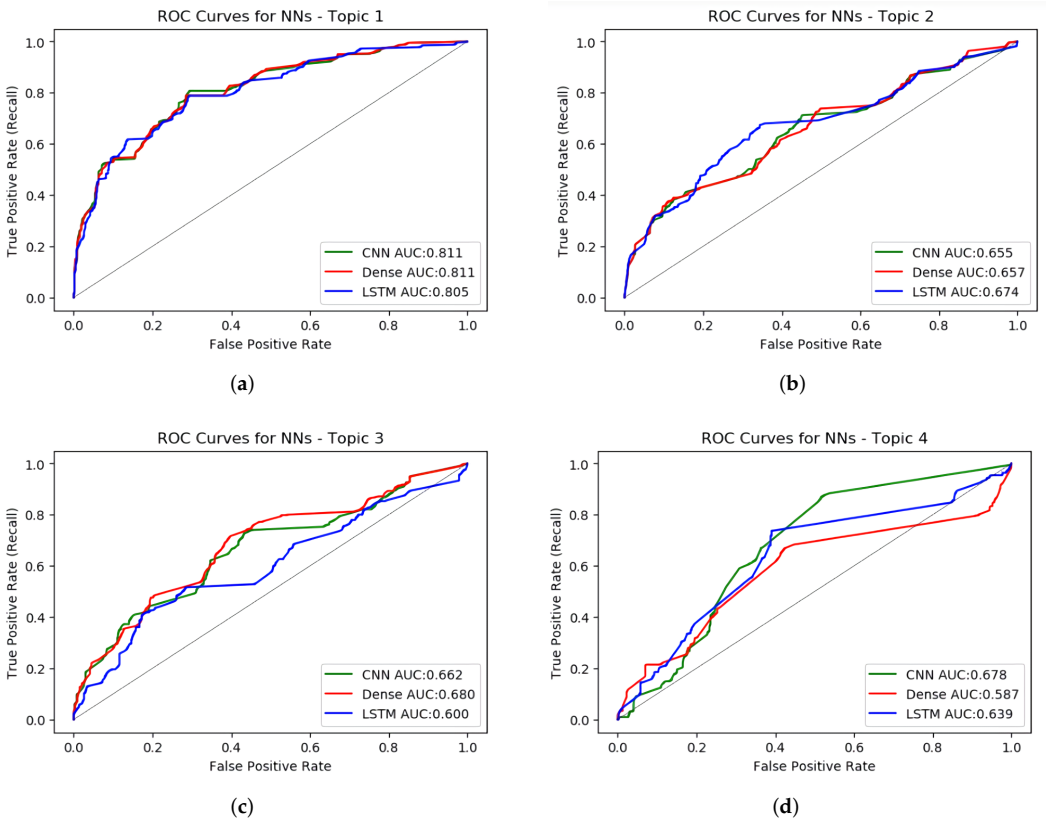

Figure 7. ROC curves of different neural networks algorithm's for the data sets related to the dominant topics. (**a**)Topic 1. (**b**) Topic 2. (**c**) Topic 3. (**d**) Topic 4.

The AUC values obtained from the different ROC curves corresponding to the deep learning algorithms analyzed, when classifying a document as being related or not to fraud within each data set T1, T2, T3, and T4, identify that there is not much difference between the models evaluated with similar average performance percentages between them (DNN = 0.68; LSTM = 0.679), with a slight superiority of DNN with 0.69.

4.2.5. Comparative Analysis

First, in this subsection, we compare the performance of linear classifiers and neural networks when applied over this scenario. The most efficient classification methods were RF and GB, averaging an AUC of 81%, as shown in Table 6. On the other hand, in evaluating the models related to neural networks, it was determined that they have similar performance; DNN slightly exceeded the others with a 1% difference, obtaining an AUC of 69%. Based on these results for the present case study, it is shown that the classification methods' performance is better when making predictions, outperforming deep learning models.

Regarding the performance of our approach compared with that of other works, there are serious issues that complicate the reproduction of their experiments when using other techniques. The most critical issue is the restricted availability of the data sets used in such works, commonly due to privacy concerns. Thus, assessing our approach directly against those of other works is an intricate task, all the more considering that ours is a novel method for detecting fraud-related behavior. Given this pitfall, we performed an additional experiment that enables comparison with the results of our work. This experiment incorporates a baseline method of topic modeling and further compares its results with those obtained with our method. This baseline method was originally oriented

to detecting spam, but the classification logic is similar to detecting fraud. Thus, we applied this strategy to our data set and compared the AUC obtained with our approach. The baseline method obtained an AUC of 0.68, whereas our fraud triangle-based approach obtained an AUC of 0.81, suggesting that our proposal is valid.

5. Conclusions

Fraud and all its variants as a social phenomenon is a latent security risk in any environment, so its analysis and study are necessary, especially investigating measures for its early detection and providing alternatives for its mitigation. This research made it possible to determine suspicious behavior by using topic modeling and the fraud triangle theory to identify patterns related to fraud within a data set.

This evidence is related to the vertices of the fraud triangle theory (pressure, opportunity, and rationalization), supporting the presence of this type of behavior for later analysis. The lack of access to information that evidences the existence of fraudulent behaviors was a critical factor in the development of this work since it forced us to generate a synthetic data set. Furthermore, an analysis of the three most popular algorithms for topic modeling (LDA, LSA, and NMF) was performed. LDA was the most effective in identifying latent themes in the study corpus and provided more "consistent" topics.

A graphical analysis of the inter-topic distance revealed that allocating documents in four topics resulted in a more coherent data set interpretation. In addition, a new representation of the data set, in terms of the probabilities of the documents belonging to each topic, was used to feed several classification algorithms to detect documents related or unrelated to fraud.

After assessing linear machine learning and deep learning algorithms, we found that some of the former were the best performers and obtained interesting results of AUC. This suggests that our approach based on the fraud triangle theory to detecting fraud-related activity is feasible under the proposed scenario. In addition, the effectiveness of deep learning models could be improved if more data are used as input.

As can be noted, the novelty of this work lies in the combination of a machine learning mechanism with a sociological model to detect fraud-related behavior. As far as we know, such a model, the fraud triangle theory, is not used as a reference frame in any other work. Thus, our approach might pave the way for addressing this problem from different perspectives, but especially for incorporating other multidisciplinary approaches.

Future Work

Due to the lack of public fraud-related data, which are key to studying fraud, an avenue of future work involves collecting more such data to feed machine learning algorithms. In addition, if this were real data, the findings here described could be confirmed in practice.

Undoubtedly, the fraud triangle theory is not the only one that tries to explain the source of fraud. Future work could be inspired by other sociological theories looking to improve the results described in this work.

Regarding text mining, we planned to apply other topic modeling techniques to improve the precision when clustering words in topics, thus contributing to the efficiency of the algorithms applied for detecting fraud-related behavior. In addition, since AUC evaluates all possible cut points, even unsuitable in practical fraud-detection applications, we will focus on other metrics, such as partial AUC.

Author Contributions: Conceptualization, M.S.-A. and L.U.-A.; methodology, M.S.-A. and J.E.-J.; validation, M.S.-A., L.U.-A. and J.E.-J.; investigation, M.S.-A.; writing—original draft preparation, M.S.-A.; writing—review and editing, L.U.-A. and J.E.-J.; supervision, L.U.-A. All authors have read and agreed to the published version of the manuscript.

Funding: The publication of this article was founded by the Vicerrectorado de Investigación, Innovación y Vinculación of the Escuela Politécnica Nacional.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author. The data are not publicly available due to privacy limitations concerning the use of personal information.

Acknowledgments: This work was partially supported by Escuela Politécnica Nacional under the research project PII-DETRI-2021-02 "Detección de fraude mediante análisis de tópicos y métodos de clasificación". Marco Sánchez is a recipient of a teaching assistant fellowship from Escuela Politécnica Nacional for doctoral studies in computer science.

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

References

1. Sanchez, M.; Torres, J.; Zambrano, P.; Flores, P. FraudFind: Financial fraud detection by analyzing human behavior. In Proceedings of the 2018 IEEE 8th Annual Computing And Communication Workshop And Conference (CCWC), Las Vegas, NV, USA, 8–10 January 2018. https://doi.org/10.1109/CCWC.2018.8301739.
2. PwC. (This Link Contains Information about FRAUD). Available online: https://www.pwc.com (accessed on 8 September 2021)
3. Abdullahi, R.; Mansor, N. Fraud Triangle Theory and Fraud Diamond Theory. Understanding the Convergent and Divergent for Future Research. *Int. J. Acad. Res. Account. Financ. Manag. Sci.* **2015**, *5*, 10. https://doi.org/10.6007/IJARAFMS/v5-i4/1823.
4. Ravisankar, P.; Ravi, V.; Rao, G.; Bose, I. Detection of financial statement fraud and feature selection using data mining techniques. *Decis. Support Syst.* **2011**, *50*, 491–500. https://doi.org/10.1016/j.dss.2010.11.006.
5. Guan, J.; Li, R.; Yu, S.; Zhang, X. A Method for Generating Synthetic Electronic Medical Record Text. *IEEE/ACM Trans. Comput. Biol. Bioinform.* **2021**, *18*, 173–182. https://doi.org/10.1109/TCBB.2019.2948985.
6. Talib, R.; Kashif, M.; Ayesha, S.; Fatima, F. Text Mining: Techniques, Applications and Issues. *Int. J. Adv. Comput. Sci. Appl.* **2016**, *7*, 414–418. https://doi.org/10.14569/IJACSA.2016.071153. Available online: https://thesai.org (accessed on 8 September 2021).
7. Kozbagarov, O., Mussabayev, R. & Mladenovic, N. A New Sentence-Based Interpretative Topic Modeling and Automatic Topic Labeling. *Symmetry* **2021**, *13*, 837. https://10.3390/sym13050837.
8. Hoyer, S.; Zakhariya, H.; Sandner, T.; Breitner, M. Fraud Prediction and the Human Factor: An Approach to Include Human Behavior in an Automated Fraud Audit. In Proceedings of the 2012 45th Hawaii International Conference On System Sciences, Maui, HI, USA, 4–7 January 2012. https://doi.org/10.1109/HICSS.2012.289.
9. Holton, C. Identifying disgruntled employee systems fraud risk through text mining: A simple solution for a multi-billion dollar problem. *Decis. Support Syst.* **2009**, *46*, 853–864. https://doi.org/10.1016/j.dss.2008.11.013.
10. Jans, M.; Lybaert, N.; Vanhoof, K. Internal fraud risk reduction: Results of a data mining case study. *Int. J. Account. Inf. Syst.* **2010**, *11*, 17–41. https://doi.org/10.1016/j.accinf.2009.12.004.
11. Jans, M.; Lybaert, N.; Vanhoof, K. A framework for internal fraud risk reduction at it integrating business processes. *Int. J. Digit. Account. Res.* **2009**, *9*, 1–29. https://doi.org/10.4192/1577-8517-v9_1.
12. Kumar, V.; Sriganga, B. A review on data mining techniques to detect insider fraud in banks. *Int. J. Adv. Res. Comput. Sci. Softw. Eng.* **2014**, *4*, 370–380.
13. Panigrahi, P. A Framework for Discovering Internal Financial Fraud Using Analytics. In Proceedings of the 2011 International Conference On Communication Systems And Network Technologies, Katra, India, 3–5 June 2011. https://doi.org/10.1109/CSNT.2011.74.
14. Jayabrabu, R.; Saravanan, V.; Tamilselvi, J. A framework for fraud detection system in automated data mining using intelligent agent for better decision making process. In Proceedings of the 2014 International Conference On Green Computing Communication And Electrical Engineering (ICGCCEE), Coimbatore, India, 6–8 March 2014. https://doi.org/10.1109/ICGCCEE.2014.6922411.
15. Yue, D.; Wu, X.; Wang, Y.; Li, Y.; Chu, C. A Review of Data Mining-Based Financial Fraud Detection Research. In Proceedings of the 2007 International Conference On Wireless Communications, Networking And Mobile Computing, Shanghai, China, 21–25 September 2007. https://doi.org/10.1109/WICOM.2007.1352.
16. Phua, C.; Lee, V.; Smith, K.; Gayler, R. A comprehensive survey of data mining-based fraud detection research. *arXiv* **2010**, arXiv:1009.6119. https://doi.org/10.1016/j.chb.2012.01.002.
17. Wang, S. A Comprehensive Survey of Data Mining-Based Accounting-Fraud Detection Research. In Proceedings of the 2010 International Conference on Intelligent Computation Technology and Automation, Changsha, China, 11–12 May 2010. https://doi.org/10.1109/ICICTA.2010.831.
18. Al-Jumeily, D.; Hussain, A.; MacDermott, A.; Tawfik, H.; Seeckts, G.; Lunn, J. The Development of Fraud Detection Systems for Detection of Potentially Fraudulent Applications. In Proceedings of the International Conference on Developments of E-Systems Engineering (DeSE), Dubai, United Arab Emirates, 13–14 December 2015. https://doi.org/10.1109/DeSE.2015.59.
19. Lopez-Rojas, E.; Axelsson, S. Social Simulation of Commercial and Financial Behaviour for Fraud Detection Research. In Proceedings of the 10th Social Simulation Conference, Barcelona, Spain, 1–5 September 2014. https://doi.org/10.13140/2.1.3512.9601.
20. Lopez-Rojas, E.; Gorton, D.; Axelsson, S. Using the RetSim Simulator for Fraud Detection Research. *Int. J. Simul. Process Model.* **2015**, *10*, 144–155. https://doi.org/10.1504/IJSPM.2015.070465.

21. Lopez-Rojas, E.; Axelsson, S. A review of computer simulation for fraud detection research in financial datasets. In Proceedings of the 2016 Future Technologies Conference (FTC), San Francisco, CA, USA, 6–7 December 2016. https://doi.org/10.1109/FTC.2016.7821715.
22. Cappelli, D.; Moore, A.; Trzeciak, R.; Shimeall, T. *Common Sense Guide to Prevention and Detection of Insider Threats*; CERT, Software Engineering Institute, Carnegie Mellon University: Pittsburgh, PA, USA, 2009. Available online: https://resources.sei.cmu.edu/library/asset-view.cfm?assetid=50275 (accessed on 23 March 2022).
23. ACFE. (ACFE—Association of Certified Fraud Examiners). Available online: https://www.acfe.com/rttn-introduction.aspx (accessed on 8 September 2021).
24. Mui, G.; Mailley, J. A tale of two triangles: comparing the Fraud Triangle with criminology's Crime Triangle. *Account. Res. J.* **2015** *28*, 45–58. https://doi.org/10.1108/ARJ-10-2014-0092..
25. Vu, H.; Li, G.; Law, R. Discovering implicit activity preferences in travel itineraries by topic modeling. *Tour. Manag.* **2019** *75*, 435-446. https://doi.org/10.1016/j.tourman.2019.06.011.
26. Daume, S.; Albert, M.; Gadow, K. Assessing Citizen Science Opportunities in Forest Monitoring Using Probabilistic Topic Modelling. *For. Ecosyst.* **2014**, *1*, 11. https://doi.org/10.1186/PREACCEPT-7308562713104429. Available online: https://forestecosyst.springeropen.com (accessed on 8 September 2021).
27. Tunazzina Islam Yoga-Veganism: Correlation Mining of Twitter Health Data. *arXiv* **2019**, arXiv:1906.07668.
28. Tresnasari, N.; Adji, T.; Permanasari, A. Social-Child-Case Document Clustering based on Topic Modeling using Latent Dirichlet Allocation. *IJCCS Indonesian J. Comput. Cybern. Syst.* **2020**, *14*, 179. https://doi.org/10.22146/ijccs.54507.
29. Schneider, P. App Ecosystem Out of Balance: An Empirical Analysis of Update Interdependence between Operating System and Application Software. Master's Thesis, Technical University of Munich, Garching, Germany, 2020.
30. Wu, Y.; Ding, Y.; Wang, X.; Xu, J. A comparative study of topic models for topic clustering of Chinese web news. In Proceedings of the 2010 3rd International Conference On Computer Science And Information Technology, Chengdu, China, 9–11 July 2010. https://doi.org/10.1109/ICCSIT.2010.5564723.
31. Alghamdi, R.; Alfalqi, K. A Survey of Topic Modeling in Text Mining. *Int. J. Adv. Comput. Sci. Appl.* **2015**, *6*. https://doi.org/10.14569/IJACSA.2015.060121. Available online: https://thesai.org (accessed on 8 September 2021).
32. O'Callaghan, D.; Greene, D.; Carthy, J.; Cunningham, P. An analysis of the coherence of descriptors in topic modeling. *Expert Syst. Appl.* **2015**, *42*, 5645–5657. https://doi.org/10.1016/j.eswa.2015.02.055.
33. Kuang, D.; Brantingham, P.; Bertozzi, A. Crime Topic Modeling. *Crime Sci.* **2017**, *6*, 12. https://doi.org/10.1186/s40163-017-0074-0. Available online: https://crimesciencejournal.biomedcentral.com (accessed on 8 September 2021).
34. Hidayatullah, A.; Aditya, S.; Karimah, Gardini, S. Topic modeling of weather and climate condition on twitter using latent dirichlet allocation (LDA). *IOP Conf. Ser. Mater. Sci. Eng.* **2019**, *482*, 012033. https://doi.org/10.1088/1757-899X/482/1/012033.
35. Jacobi, C.; Atteveldt, W.; Welbers, K. Quantitative analysis of large amounts of journalistic texts using topic modelling. *Digit. J.* **2015**, *4*, 89–106. https://doi.org/10.1080/21670811.2015.1093271.
36. Blei, D. Probabilistic topic models. *Commun. ACM* **2012**, *55*, 77. https://doi.org/10.1145/2133806.2133826.
37. Cosovic, M.; Amelio, A.; Junuz, E. Classification Methods in Cultural Heritage. In Proceedings of the Visual Pattern Extraction and Recognition for Cultural Heritage Understanding (VIPERC2019), Pisa, Italy, 30 January 2019. Available online: http://ceur-ws.org (accessed on 8 September 2021).
38. EntezariMaleki, R.; Rezaei, A.; MinaeiBidgoli, B. Comparison of Classification Methods Based on the Type of Attributes and Sample Size. *J. Converg. Inf. Technol.* **2009**, *4*, 94–102. https://doi.org/10.4156/jcit.vol4.issue3.14.
39. Fawcett, T. Introduction to ROC analysis. *Pattern Recognit. Lett.* **2006**, *27* 861–874. https://doi.org/10.1016/j.patrec.2005.10.010.
40. Novakovic, J.; Veljovic, A.; Ilic, S.; Papic, M. Experimental study of using the k-nearest neighbour classifier with filter methods. In Proceedings of the Computer Science and Technology, Varna, Bulgaria, 30 April 2016.
41. Zhang, Z. Introduction to machine learning: K-nearest neighbors. *Ann. Transl. Med.* **2016**, *4*, 218. https://doi.org/10.21037/atm.2016.03.37.
42. Basha, S.; Rajput, D. Chapter 9—Survey on Evaluating the Performance of Machine Learning Algorithms: Past Contributions and Future Roadmap. *Deep. Learn. Parallel Comput. Environ. Bioeng. Syst.* **2019**, 153–164. https://doi.org/10.1016/B978-0-12-816718-2.00016-6.
43. Mashat, A.; Fouad, M.; Yu, P.; Gharib, T. A Decision Tree Classification Model for University Admission System. *J. Adv. Comput. Sci. Appl.* **2012**, *3*. https://doi.org/10.14569/IJACSA.2012.031003. Available online: https://thesai.org (accessed on 8 September 2021).
44. Oshiro, T.; Perez, P.; Baranauskas, J. How Many Trees in a Random Forest? In *International Workshop on Machine Learning and Data Mining in Pattern Recognition*; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2012; Volume 7376. https://doi.org/10.1007/978-3-642-31537-4_13. Available online: https://www.researchgate.net (accessed on 8 September 2021).
45. Ali, J.; Khan, R.; Ahmad, N.; Maqsood, I. Random Forests and Decision Trees. *Int. J. Comput. Sci. Issues* **2012**, *9*, 272. Available online: http://ijcsi.org/papers/IJCSI-9-5-3-272-278.pdf (accessed on 23 March 2022).
46. Kamel, H.; Abdulah, D.; Al-Tuwaijari, J. Cancer Classification Using Gaussian Naive Bayes Algorithm. *Int. J. Intell. Eng. Syst.* **2019**, *14*, 134–146. https://doi.org/10.1109/IEC47844.2019.8950650.
47. Yang, T.; Chen, W.; Cao, G. Automated classification of neonatal amplitude-integrated EEG based on gradient boosting method. *Biomed. Signal Process. Control.* **2016**, *28*, 50–57. https://doi.org/10.1016/j.bspc.2016.04.004.

48. Ding, C.; Cao, X.; Næss, P. Applying gradient boosting decision trees to examine non-linear effects of the built environment on driving distance in Oslo. *Transp. Res. Part Policy Pract.* **2018**, *110*, 107–117. https://doi.org/10.1016/j.tra.2018.02.009.
49. Cervantes, J.; García-Lamont, F.; Rodríguez, L.; Lopez-Chau, A. A comprehensive survey on support vector machine classification: Applications, challenges and trends. *Neurocomputing* **2020**, *408*, 189–215. https://doi.org/10.1016/j.neucom.2019.10.118.
50. Amatriain, X.; Pujol, J. Data Mining Methods for Recommender Systems. In *Recommender Systems Handbook*; Springer: Boston, MA, USA, 2015; pp. 227–262, https://doi.org/10.1007/978-0-387-85820-3_2
51. Liang, J.; Xue, L.; Lin, X.; Shen, X. Verifiable and Secure SVM Classification for Cloud-Based Health Monitoring Services. *IEEE Internet Things J.* **2021**, *8*, 17029–17042. https://doi.org/10.1109/JIOT.2021.3075540.
52. Zhang, Z. A gentle introduction to artificial neural networks. *Ann. Transl. Med.* **2016**, *4*, 370. https://doi.org/10.21037/atm.2016.06.20.
53. Nhu, V.; Hoang, N.; Nguyen, H.; Thao, N.; Bui, T.; Hoa, P.; Samui, P.; Bui, D. Effectiveness assessment of Keras based deep learning with different robust optimization algorithms for shallow landslide susceptibility mapping at tropical area. *Catena* **2020**, *188*. https://doi.org/10.1016/j.catena.2020.104458.
54. Benuwa, B.; Zhan, Y.; Ghansah, B.; Wornyo, D.; Banaseka, F. A Review of Deep Machine Learning. *Int. J. Eng. Res. Afr.* **2016** , *24*, 124–136. https://doi.org/10.4028/www.scientific.net/JERA.24.124.
55. Volz, B.; Behrendt, K.; Mielenz, H.; Gilitschenski, I.; Siegwart, R.; Nieto, J. A data-driven approach for pedestrian intention estimation. In Proceedings of the 2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC), Rio de Janeiro, Brazil, 1–4 November 2016. https://doi.org/10.1109/ITSC.2016.7795975.
56. Nazari, F.; Yan, W. Convolutional versus Dense Neural Networks: Comparing the Two Neural Networks Performance in Predicting Building Operational Energy Use Based on the Building Shape. *arXiv* **2021**, arxiv:2108.12929.
57. Yamashita, R.; Nishio, M.; Do, R.; Togashi, K. Convolutional Neural Networks: An Overview and Application in Radiology. *Insights Into Imaging* **2018**, *9*, 611–629. https://doi.org/10.1007/s13244-018-0639-9. Available online: https://insightsimaging.springeropen.com (accessed on 8 September 2021).
58. Islam, M.; Islam, M.; Asraf, A. A Combined Deep CNN-LSTM Network for the Detection of Novel Coronavirus (COVID-19) Using X-ray Images. *Informatics Med. Unlocked* **2020**, *20*, 100412. https://doi.org/10.1016/j.imu.2020.100412.
59. Li, W.; Tao, W.; Qiu, J.; Liu, X.; Zhou, X.; Pan, Z. Densely Connected Convolutional Networks With Attention LSTM for Crowd Flows Prediction. *IEEE Access* **2019**, *7*, 140488–140498. https://doi.org/10.1109/ACCESS.2019.2943890.
60. Ozyirmidokuz, E. Mining Unstructured Turkish Economy News Articles. *Procedia Econ. Financ.* **2014**, *16*, 320–328. https://doi.org/10.1016/S2212-5671(14)00809-0.
61. Santos, Brito, Y.; Santos, C.; Paula, Mendonca, S.; Araujo, T.; Freitas, A.; Meiguins, B. A Prototype Application to Generate Synthetic Datasets for Information Visualization Evaluations. In Proceedings of the 2018 22nd International Conference Information Visualisation (IV), Fisciano, Italy, 10–13 July 2018. https://doi.org/110.1109/iV.2018.00036.
62. Redpath, R.; Srinivasan, B. Criteria for a Comparative Study of Visualization Techniques in Data Mining. In *Intelligent Systems Design and Applications*; Springer: Berlin/Heidelberg, Germany, 2003; Volume 23, pp. 609–620. https://doi.org/10.1007/978-3-540-44999-7_58.
63. Audinet. (Using Key Word Analysis of an Organization's Big Data For Error and Fraud Detection). Available online: https://www.auditnet.org/key-word-analytics (accessed on 8 September 2021).
64. Randomwordgenerator. (Random Word Generator). Available online: https://www.randomwordgenerator.org (accessed on 8 September 2021).
65. Reverso. (Reverso Context). Available online: https:/https://context.reverso.net/traduccion/ingles-espanol (accessed on 8 September 2021).
66. Sentencedict. (Sentence Dict). Available online: https://sentencedict.com/ (accessed on 8 September 2021).
67. Kastrati, Z.; Kurti, A.; Imran, A. WET: Word Embedding-Topic Distribution Vectors for MOOC Video Lectures Dataset. *Data Brief.* **2020**, *28*, 105090. Available online: http://www.sciencedirect.com (accessed on 23 March 2022).
68. Maldonado, M.; Alulema, D.; Morocho, D.; Proano, M. System for monitoring natural disasters using natural language processing in the social network Twitter. In Proceedings of the 2016 IEEE International Carnahan Conference On Security Technology (ICCST), Orlando, FL, USA, 24–27 October 2016. https://doi.org/10.1109/CCST.2016.7815686.
69. Maier, D.; Waldherr, A.; Miltner, P.; Wiedemann, G.; Niekler, A.; Keinert, A.; Pfetsch, B.; Heyer, G.; Reber, U.; Häussler, T.; et al. Applying LDA Topic Modeling in Communication Research: Toward a Valid and Reliable Methodology. *Commun. Methods Meas.* **2018**, *12*, 93–118. https://doi.org/10.1080/19312458.2018.1430754.
70. Schofield, A.; Magnusson, M.; Mimno, D. Pulling Out the Stops: Rethinking Stopword Removal for Topic Models. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*; Association for Computational Linguistics: Valencia, Spain, 2017. https://doi.org/10.18653/v1/E17-2069.
71. Rehurek, R.; Sojka, P. Software Framework for Topic Modelling with Large Corpora. In Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks, Valletta, Malta, 22 May 2010. https://doi.org/10.13140/2.1.2393.1847.
72. Kherwa, P.; Bansal, P. Topic Modeling: A Comprehensive Review. *EAI Endorsed Trans. Scalable Inf. Syst.* **2020**, *7*, e2. https://doi.org/10.4108/eai.13-7-2018.159623. Available online: https://eudl.eu (accessed on 8 September 2021).
73. Albalawi, R.; Yeap, T.; Benyoucef, M. Using topic modeling methods for short-text data: A comparative analysis. *Front. Artif. Intell.* **2020**, *3*, 42. https://doi.org/10.3389/frai.2020.00042.

74. George, S. Comparison of LDA and NMF Topic Modeling Techniques for Restaurant Reviews. *Indian J. Nat. Sci.* **2020**, *10*. Available online: https://www.researchgate.net (accessed on 8 September 2021).
75. Mifrah, S.; Benlahmar, E. Topic modeling coherence: A comparative study between LDA and NMF models using COVID-19 corpus. *Int. J. Adv. Trends Comput. Sci. Eng.* **2020**, *9*, 5756–5761. https://doi.org/10.30534/ijatcse/2020/231942020.
76. Merino, S.; Atzmueller, M. Multimodal Behavioral Mobility Pattern Mining and Analysis Using Topic Modeling on GPS Data. *Behav. Anal. Soc. Ubiquitous Environ.* **2019**, *11406*, 68–88. https://doi.org/10.1007/978-3-030-34407-8_4.
77. Zhao, Y.; Zhang, J.; Wu, M. Finding Users' Voice on Social Media: An Investigation of Online Support Groups for Autism-Affected Users on Facebook. *Int. J. Environ. Res. Public Health* **2019**, *16*, 4804. https://doi.org/10.3390/ijerph16234804.
78. Jain, N. Data mining techniques: A survey paper. *Int. J. Res. Eng. Technol.* **2013**, *2*, 116–119. https://doi.org/10.15623/ijret.2013.0211019.
79. AUC. Available online: https://neptune.ai/blog/f1-score-accuracy-roc-auc-pr-au (accessed on 15 July 2021).
80. Straube, S.; Krell, M. How to Evaluate an Agent's Behavior to Infrequent Events?—Reliable Performance Estimation Insensitive to Class Distribution. *Front. Comput. Neurosci.* **2014**, *8*, 43. Available online: https://www.frontiersin.org/article/10.3389/fncom.2014.00043 (accessed on 23 March 2022).