Machine Learning for Text Anomaly Detection: A Systematic Review

Karima Boutalbi, Faiza Loukil, David Telisson, Kavé Salamatian, Hervé Verjus

Machine Learning for Text Anomaly Detection: A Systematic Review

Karima Boutalbi*§, Faiza Loukil*, Hervé Verjus*, David Telisson*, Kavé Salamatian*
* LISTIC, Savoie Mont Blanc University, Annecy-le-Vieux, France;
{firstName.lastName}@univ-smb.fr
§ Cegedim SRH, Lyon, France;
karima.boutalbi@cegedim-srh.com

***Abstract*—Anomaly detection is a common task in various domains, which has attracted significant research efforts in recent years. Existing reviews mainly focus on structured data, such as numerical or categorical data. Several studies treated review of anomaly detection in general on heterogeneous data or concerning a specific domain. However, anomaly detection on unstructured textual data is less treated. In this work, we target textual anomaly detection. Thus, we propose a systematic review of anomaly detection solutions in the text. To do so, we analyze the included papers in our survey in terms of anomaly detection types, feature extraction methods, and machine learning methods. We also introduce a web scrapping to collect papers from digital libraries and propose a clustering method to classify selected papers automatically. Finally, we compare the proposed automatic clustering approach with manual classification, and we show the interest of our contribution.**

***Index Terms*—Anomaly detection, Document clustering, Machine Learning, Feature Extraction.**

I. Introduction

Anomaly detection (AD) is a classical problem in computer science, with several methods developed for different applications. According to [4], anomalies *"are patterns in data that do not conform to a well-defined notion of normal behaviour"*. When normal behaviour is well-defined, outliers are points that are not following the normal "behaviour". Anomaly detection (or outlier detection) is the ability to separate normal observations from abnormal ones in a data stream.

In this review, we concentrate on textual data. In this context, grammatical errors, inappropriate words, like swearword, or intentionally false statements (fake news), can be considered as anomalies, and detecting them falls in the area of text anomaly detection. Moreover, the textual analysis might be used to detect anomalies in systems generating the text, *e.g.*, in security, the analysis of textual logs can be the mean for intrusion detection.

AD algorithms generally use measures or metrics that evaluate how far an observed sequence is from a normal behaviour model. Several approaches have been developed with different metrics and various normal behaviour model structures. In this study, we will focus on methods leveraging Machine Learning (ML) for evaluating the metrics or for building the normal behaviour models.

The section II describes the steps of the adopted systematic review methodology. Section III presents the study results and gives answers to our research questions. Section IV proposes a method for an automatic classification of papers based on their abstracts. Section V discusses existing reviews that study text anomaly detection and section VI concludes the paper.

II. Research methodology

We perform here a systematic review of the literature on text AD that follows the guidelines proposed by Petersen et al. [18]. The systematic review is divided into five stages, namely defining research questions, conducting a search strategy, screening of papers, using the abstract to extract keywords, and finally extracting data and presenting results. These steps are detailed below. Through this literature review, we aim into providing helpful references to researchers who are willing to use text AD in their studies.

A. Research questions

This step consists in defining research questions. Our goal is to classify the contributions of the research in AD, according to the areas of application, the used mechanisms of feature extraction and text vectorization, and the application of machine learning methods. The research questions are described below:

1) RQ1: What are the main categories of AD types proposed recently for text-based applications?
2) RQ2: What are the different feature extraction methods used to represent textual data in AD?
3) RQ3: What are the categories of machine learning methods applied to textual AD?

B. Conducting a search strategy

The second step of the adopted methodology entails collecting papers from digital libraries by selecting carefully keywords related to text AD and the application of machine learning to it.

The search was done over four scientific paper search engines, namely ACM, IEEE Xplore, Springer, and Science Direct for papers published between 2012 and 2022. We use boolean operators (ORs and ANDs), as well as more specific terms, like intrusion, or outliers, to limit the number of search

results. We show in Table I the queries we used for this search. As Springer Library does not allow searching in the papers' abstracts, we limit the search to the title with keywords "anomaly detection" and "text". ScienceDirect makes possible to search in the title, the abstract, or the keywords specified in the papers, resulting into a better precision. For IEEE and ACM, we search keywords in the abstract and use an extended list of words, with synonyms, like 'outlier', 'intrusion', etc.

TABLE I: Used research query per digital library.

Digital library	Used Research Query
Springer	Title: "anomaly detection" AND ("text" OR "learning")
IEEE	Abstract: ("Anomaly detection" OR "Outlier detection" OR "intrusion") AND ("Text" OR "textual")
ACM	Abstract: ("Anomaly detection" OR "Outlier detection" OR "intrusion") AND ("Text" OR "textual")
Science Direct	Title, abstract, keywords: "anomaly detection" AND "text"

C. Screening of papers

The third step consists of choosing papers that are relevant to answer the research questions listed before. For this purpose, a set of inclusion and exclusion criteria have been defined; we exclude papers that do not include AD keywords even if they use ML methods, papers that are not published in English, and papers that consist of literature review papers and surveys. We only keep papers that apply ML methods to text, and that are published during the considered period.

This resulted into collecting 2020 papers that are reduced to 108 papers after applying the inclusion and exclusion criteria. The remaining papers mostly concerned text anomaly detection using machine learning methods. Table II shows the number of selected and excluded papers for each digital library.

TABLE II: Selected papers results.

Library	Nbr of papers	Nbr of selected papers	Nbr of excluded papers
Springer	50	13	37
IEEE	240	62	178
ACM	1700	21	1679
Science Direct	30	12	18

We developed a web scraping software [15] in Python to fetch the papers. We gathered a dataset of 2021 samples. Web scraping is a technique for extracting content from websites via a script or a program. However, not all digital libraries' websites allow web scrapping; for example, Springer and IEEE authorize it, but ACM and Science Direct do not.

For each search result page related to the ACM library, we download the Bibtex text file, containing 10 papers per file; then we concatenate them and apply text processing to structure the data in the form of a data frame, allowing the transformation from the Bibtex format to text structured in a table. For Science Direct, data was collected manually, and fortunately, the search results were quite small, around 30 papers. Once the data were well structured and merged, the next step was to refine our research. We begin to look for keywords in the paper abstract because the paper title does not always contain important information. Our method helped us to save time by reducing the number of papers to process.

III. Results

The results of each research question are addressed in detail in the following sub-sections. In order to synthesize the information gathered from the selected papers, we used various processes to answer the research questions.

A. Publication Trends

To examine the trend of the text AD field in terms of the publication date, we carry out a statistical analysis to understand the distribution of selected papers regarding the year and the journal. We notice that the number of papers has a growing trend between 2012 and 2021. With peaks in 2016 and 2020 with 13 and 19 papers, respectively. Moreover, the number of papers found in IEEE (62 papers) is greater than the number of papers in other digital libraries.

B. Types of text-based anomaly detection

In this section, we address RQ1, which aims to identify types of text-based AD studies. Our study shows that several types of text AD are adopted, in different application domains, such as healthcare, financial, and manufacturing. We identify four main types, namely system intrusion detection, spam detection, anomalous topic document discovery, and event detection. These text-based anomaly detection types are the most studied in recent years between 2012 and 2022. These identified types refer to 25 different application domains, like health surveillance, social media cyberbullying, fraud detection, etc.

Figure 1 illustrates the distribution of text AD types per year during the considered period. In 2012, we find the lowest number of papers; note that the publication only concerns spam detection. We can also observe that intrusion detection was the most studied in the last years, between 2019 and 2022.

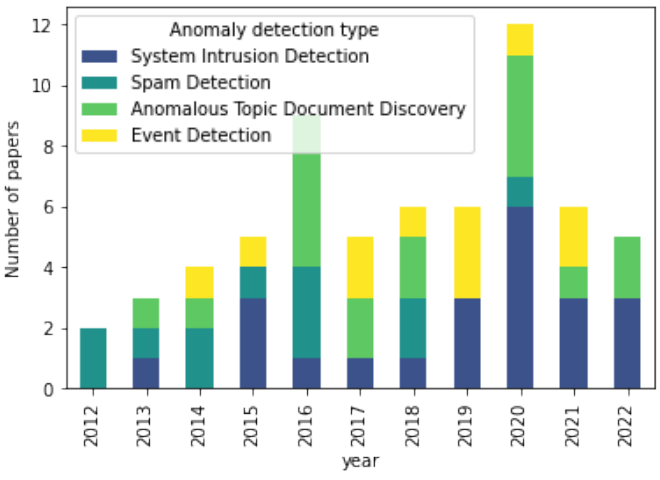

Fig. 1: Main types of text-based anomaly detection by year.

We detail below each identified type of text-based AD.

1) System intrusion detection: An intrusion in a system can be considered as an anomaly, it is a computer security breach, an abnormal access to a computer system. It aims to identify intrusions with a low false alarm rate and a high detection rate [25]. Monitoring security can be done through log analysis. Log files are text files that contain information about records generated by a computer for different types of tasks. The exploitation of such textual information can be used for the detection of malicious attacks. The spreading of malicious social media messages is considered as an anomaly.

2) Spam detection: Spam is an unwanted message usually received by mail or via social networks or web pages. It can have a commercial purpose or be an attempt at intrusion. The most relevant studies include e-mail spam detection, web spam detection, and opinion spam detection.

3) Anomalous topic document discovery: Anomalous topic document discovery focuses on detecting atypical patterns (topics) in text documents. It identifies outlier documents whose content is different or dissimilar to inlier sub-topics within a larger document corpus. This type of AD is used on the web and blogs. A subset of unusual web pages on a website or blog pages on a blog deviating from the common themes in the website/blog. Besides, detecting uncommon new posts or comments on social media may help to flag them as emerging, hate comments, or fake news. According to [1] in judicial-based text-based documents, a court case is to be considered as an anomaly if the judge's decision differs significantly from existing decisions in similar cases.

4) Event Detection: Detecting an abnormal event is detecting anomalies in streaming data or online data. The goal of automatic event detection is to deal with large volumes of data and to satisfy real-time processing constraints to recognize what is happening in real time. For instance, messages carrying critical information are detected for the damage assessment task, for disaster management. According to [12], fake news can be regarded as an anomaly event on social networks. The spread of fake news brings great negative effects on people's daily life and even causes social panic.

C. Methods of feature extraction for text representation

Feature extraction for text representation is the process of converting text data into numeric data. It is also called text vectorization or text embedding. In the context of text AD, text vectorization is used to convert text documents into numerical representations that capture the features of the text data, such as words and their frequency. This enables the use of statistical and machine learning methods to identify patterns or anomalies in the data. This step is very important to improve the precision of the proposed models.

In this section, we address RQ2, which aims to present different feature extraction methods for text representation. We identify 14 different feature extraction methods in the selected papers. In Natural Language Processing (NLP), feature extraction is one of the most important steps. After the initial text cleaning, it is transformed into its features to be used for modeling. Document data are not computable; so they must be transformed into numerical data, namely a vector space model. This transformation task is generally called feature extraction of document data. In our study, we distinguish two types of feature extraction methods, namely frequency-based methods and prediction-based methods. Due to the lack of space, we illustrate this taxonomy, which can be accessed via this link[1]. The first one is based on the frequency of words in the text. The second one is based on a predicted model that uses a learning-based model to predict the vector's document.

1) Frequency-based methods: are as below:

Bag-of-words: It is one of the most used text vectorization techniques. A bag-of-words (BOW) is a representation of text that describes the occurrence of words within a document. This method is specially used in text classification tasks because it is simple and intuitive. The size of each document's vector is the same after BOW. But, it can create sparsity because of the large vocabulary, and this method does not consider the order of sentences [19].

Bag of n-grams: A bag-of-n-grams model represents a text document as an unordered collection of its n-grams. It is a probability model of word sequences. It is simple and easy to implement. Moreover, it is able to capture the semantic meaning of the sentence. It can slow down the algorithm when we move from unigram (using one word of the document) to n-gram (using n number of words of the document) because the dimension of vector formation increases [3].

Term Frequency and Inverse Document Frequency: It is a statistical measure that evaluates how relevant a word is to a document in a collection of documents. Term Frequency (TF) is the number of times a word appears in each document. It is divided by the total number of words in that document, and the IDF value is normalized using a log because, without it, the IDF value could be high. This technique is widely used but deals with a problem of sparsity in a large dataset, because the large vocabulary leads to dimensionality increases, slowing down the algorithm. Besides, the semantic meaning using TF-IDF is not considered [20].

Graph representation: A graph is represented with a square adjacency matrix; with the same documents in rows and columns. Documents represent the nodes of the graph, and a distance, like cosines distance, is computed between each couple of documents in the matrix [22].

2) Prediction-based methods: are as below:

Bidirectional Encoder Representations from Transformers (BERT): It belongs to the family of transfer learning models, it is a pre-trained model that NLP users can download and use for free. These models can be used to extract high-quality language features from text data and fine-tune these models on a specific task with your own data to produce state-of-the-art predictions. BERT captures the contextual meaning of a sentence, even if there's no keyword or phrase overlap. That means if a given same word appears in two different sentences, BERT will not produce the same word embedding for each same word [6].

[1] https://drive.google.com/file/d/1L7j6e_2S_2H_1KaKy7IoHVICGi4qBEVR/view?usp=share_link

Latent Semantic Analysis (LSA): Its basic concept consists in mapping texts represented in a high-dimensional vector space model to lower-dimensional latent semantic space. This mapping is achieved through SVD (singular value decomposition) of items or document matrix [9].

Glove: It is an unsupervised learning algorithm for obtaining vector representations of words. Training is performed on aggregated global word-word co-occurrence statistics from a corpus, and the resulting representations showcase interesting linear substructures of the word vector space [17].

Word2vec: It is a deep learning-based technique that converts a given word into a vector as a collection of numbers. Word2vec captures the semantic meaning of a word in a sentence. Words with similar contexts occupy close spatial positions. Word2vec creates a low-dimension dense vector (non-zeros) vector where each word is represented by a vector with a size that we can specify [5].

Doc2vec: It is also a deep learning-based technique that converts a given sentence into a vector as a collection of numbers, like Word2vec.

AutoEncoder: An AutoEncoder for text vectorization takes inputs from a sequence of text, passes it through a layer with fewer nodes than the input layer, and outputs it to a decoder which tries to reconstruct the exact same input. This approach allows learning vector representations to any unstructured text of any length. After training the AutoEncoder, the encoder model can be used to generate embeddings to any input.

Convolutional neural network (CNN): It is a neural network classification model used for text feature extraction [10]. Filters with different lengths are used to convolve the text matrix. The max pooling is employed to operate extractive vectors of every filter. Finally, each filter corresponds to a digit and connects these filters to obtain a vector representing this sentence, on which the final prediction is based.

D. Categories of machine learning methods

In this section, we address the RQ3, which aims to present a classification of used ML methods in included research papers. In our study, we identify three categories of ML methods used in text AD, namely supervised, unsupervised, and semi-supervised. We observe that supervised methods are the most used, followed by unsupervised ones. On the opposite, semi-supervised methods and studies that combine supervised and unsupervised methods are not widely used.

We detail below each identified category:

1) Supervised learning for anomaly detection: Supervised learning is a category in which labeled data are fed as input to ML algorithms. The input and output are already known. For instance, Bobur et al. [1] present a model for searching anomalies in judicial practice. For searching anomalies, they mix two models, including classification models, such as Logistic regression as well as similarity algorithms, such as Latent Dirichlet Allocation (LDA).

For their part, the authors of [24] analyze access logs for detecting anomalous activities, such as intrusion. The proposed approach identifies the suspicious activities and serious anomalies that may be one of the ways for hackers to hack the system. The supervised neural network approach using Naive Bayes Multinomial Text Algorithm gives better anomaly prediction than the unsupervised neural network approach for static logs and achieves maximum prediction accuracy compared to other classifier algorithms since the input attributes (logs) are strings.

2) Unsupervised learning for anomaly detection: Unsupervised learning methods are a category of learning methods in which only input data are fed to the model but no corresponding output data. In such cases, ML algorithms find the similarities among different input data and group them based on the similarity. The number of groups (clusters) must be specified in advance. The unsupervised problem in AD is a problem in which a new instance of test data differs in some aspect from the data available during training. These methods do not need large amounts of labeled data to train the model. For instance, detecting fake news as anomalies requires an unbalanced dataset in which the number of abnormal observations is lower than the number of normal observations.

Therefore, some researchers apply unsupervised methods to detect fake news on social networks. In [13], the authors proposed a method based on text content for unsupervised fake news detection. They make use of auto-encoder as the basic unsupervised learning method and suggested an improved method based on auto-encoder, namely UFNDA, for unsupervised fake news detection.

In [8], Eshraqi et al. consider spam as an anomaly problem and identify comprehensive features of spammers and spam tweets. They use a clustering algorithm in short textual messages, namely tweets. They use the DenStream algorithm for clustering, which is an algorithm that is developed based on the DBSCAN density-based clustering algorithm.

3) Semi-supervised learning for anomaly detection: Semi-supervised learning is a category of ML methods in which input data are known and only some input data are labeled. The data are partially annotated. Semi-supervised outlier detection methods identify anomalous/rare behavior through data when the normal behavior is only known and defined. Semi-supervised outlier detection methods are usually used when only examples of normal classes can be acquired, unlike anomalous/rare classes [7].

For instance, Steyn and de Waal [23] construct a Multinomial Naïve Bayes classifier and enhance it with an augmented Expectation Maximization (EM) algorithm. Thus, they use large amounts of unlabelled data and show how the EM algorithm can increase the accuracy of the Naïve Bayes classifier. The process is applied to a binary classification environment to detect anomalies in text.

IV. Automatic clustering of selected papers

The objective of this section is to provide an automatic unsupervised classification (or clustering) of selected papers and compare the obtained classification with the manual classification presented in sub-section III-C.

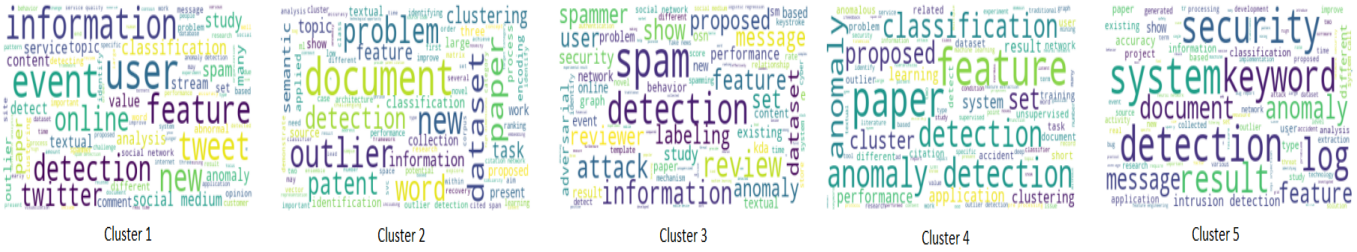

Fig. 2: Word cloud of the five obtained automatic clusters.

A. Text clustering

Text clustering aims to automatically group documents into a fixed number of categories using unsupervised methods. It uses explicit or implicit similarity/dissimilarity measures to determine how similar or dissimilar texts are. It is used in several application domains to discover text clusters based on extracted features (or text representation). A well-liked approach for text representation is the Bag-Of-Words (BOW), where the word occurrences describe the text. The BOW representation is then used to apply various specialized clustering methods, such as K-means or Spherical K-means [2], to categorize the text. The BOW representation may produce good clustering outcomes, mainly when the topics of the clusters are very different. However, the text's rich contextual information and sequence information are not recorded by BOW representation. Recent state-of-the-art outcomes on various NLP (Natural Language Processing) tasks, such as question answering and text production, have been attained by complex language models like the famous Bidirectional Encoder Representations from Transformers (BERT) [6]. The word representations offered by transformers rely on the context of the given word, in contrast to earlier word embedding techniques that provided one distinct vector for each lexicon word. As a result, they are also known as contextual word embeddings, which are really effective for the text classification task.

B. Proposed approach

In this section, we describe our approach that we developed for the automatic clustering of selected papers. As discussed in the previous section III, the transformers-based approach provides an effective text representation that depends on a context allowing important improvements for the text clustering task. Thus, we use SentenceBert [21], which is a transformer model to encode sentences. Unlike classical transformers that encode words, SentenceBert provides a vector representation for text fragments. This representation is better than a frequency-based representation or static word embedding in our study since all topics covered by selected papers deal with the area of AD. SentenceBert is able to capture the context of the paper's abstract, which helps us to obtain a better representation. On the other hand, we use the Spherical K-means [2] as a clustering algorithm and apply it to the abstract representation of papers using SentenceBert. Noting that in Spherical K-means, we need to specify the number of clusters. Hence, we choose the same number of the most frequent types of text-based AD that we identified to respond to RQ1, namely system intrusion detection, spam detection, anomalous topic document discovery, and event detection. Then, we add an additional cluster to group the rest of the papers. Thus, the number of input clusters is equal to five.

C. Manuel clustering vs. automatic clustering

Figure 2 represents the word cloud of the five obtained clusters, for that we extract the most frequent 100 words in each text cluster to represent the main topic. We can see that:

- Cluster 1 represents the topic of event detection in textual data and mainly in Twitter. We find frequent words, such as event, detection, online, Twitter, etc.
- Cluster 2 represents the outlier document discovering with words, such as document, outlier, new, paper, etc.
- Cluster 3 represents the topic of spam and attack detection. We find words, such as spam, attack, anomaly, etc.
- Cluster 4 represents an open cluster with paper dealing with different kinds of anomaly without any specificity.
- Cluster 5 represents the topic of SID through several words, such as security, system, log, and message.

Table III presents the number of papers classified manually, in each AD type, the number of papers in each cluster obtained with Spherical K-means, and the percentage of well-classified papers. We can notice that the percentage of common papers between the manual and the automatic clustering for some clusters is really close, such as clusters 2 and 3 representing anomalous topic document discovery and spam detection, with percentages of common papers equal to 76% and 83%, respectively. For clusters 1 and 5, the percentage of common papers between the manual and the automatic clustering is lower than the previous clusters, with percentages around 54% and 36%, respectively, which can be explained by the fact that clusters 2 and 3 are about more specific topics than cluster 1 and 5 that are more general topics, such as event detection and SID. Finally, cluster 4 encompasses all the other topics that are not included in the previous clusters. These other topics are those with the lowest percentage, which is explained by the fact that it is the biggest cluster with diverse topics.

V. Related literature reviews

Anomaly detection has received more attention in the past few years. For instance, Nassif et al. [16] conducted a systematic literature review dealing with machine learning methods

TABLE III: Manual clustering vs. automatic clustering results.

Clusters	Manual classified papers	Well-classified papers	% of commons papers	Cluster size
Event Detection / Cluster 1	11	4	36	16
Anomalous Topic Document Discovery / Cluster 2	17	13	76	28
Spam Detection / Cluster 3	12	10	83	21
Other / Cluster 4	45	11	24	16
System Intrusion Detection (SID) / Cluster 5	22	12	54	24

for AD. The study provided an overview of ML methods, vectorization, and evaluation metrics but does not address text data. Several literature reviews have addressed AD in text data but each addressed a specific domain or application. Kokatnoor et al. [11] provided a comparative study of text mining algorithms for AD in online social networks, but is restricted to a comparative analysis of the performance of four classification algorithms for a Twitter dataset. Finally, Mangathayaru et al. [14] explored a text mining-based approach for intrusion detection. They discussed the feature reduction methods they adopted to achieve dimensionality reduction in high-dimensional text documents. Moreover, they compared different methods of prototypes implemented in the detection of abusive content by analyzing both images and text.

To sum up, the existing systematic reviews concerning AD in the text consider only one anomaly detection type, one feature extraction method, or propose a review of ML methods in general. Our work extends the existing reviews by trying to explore all these aspects.

VI. Conclusion

Text AD is a common task in various domains. Several works treated AD considering different types of data. In this paper, we presented a systematic review of machine-learning approaches for text AD. Thus, we included 108 papers to address the defined research questions. Thus, we first carried out a statistical analysis of the selected papers. Then, we proposed a classification of the selected papers based on AD types and feature extraction methods for text representation. Moreover, we investigated the machine learning methods used for AD in the text, and we found that the majority of the included papers used supervised ML approaches. Finally, we proposed an approach for the automatic clustering of selected papers using the Spherical K-means algorithm and SentenceBert to represent the paper's abstracts. We discover 5 clusters that have been compared to the manual classification. We found that the fifth cluster is an open cluster that encompasses all the other topics that are not included in the previous clusters. A potential future work might analyze the open cluster and extract additional underrepresented classes for AD in text.

References

[1] Bobur, M., Aibek, K., Abay, B., Hajiyev, F.: Anomaly detection between judicial text-based documents. In: 2020 IEEE 14th International Conference on Application of Information and Communication Technologies (AICT). pp. 1–5 (2020)

[2] Buchta, C., Kober, M., Feinerer, I., Hornik, K.: Spherical k-means clustering. Journal of statistical software **50**(10), 1–22 (2012)

[3] Cavnar, W.B., Trenkle, J.M., et al.: N-gram-based text categorization. In: Proceedings of SDAIR-94, 3rd annual symposium on document analysis and information retrieval. vol. 161175. Las Vegas, NV (1994)

[4] Chandola, V., Banerjee, A., Kumar, V.: Anomaly detection: A survey. ACM computing surveys (CSUR) **41**(3), 1–58 (2009)

[5] Church, K.W.: Word2vec. Natural Language Engineering **23**(1), 155–162 (2017)

[6] Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: Pre-training of deep bidirectional transformers for language understanding. In: ACL. pp. 4171–4186 (2019)

[7] El-Kilany, A., Tazi, N.E., Ezzat, E.: Semi-supervised outlier detection via bipartite graph clustering. In: 2016 IEEE/ACS 13th International Conference of Computer Systems and Applications (AICCSA). pp. 1–6 (2016)

[8] Eshraqi, N., Jalali, M., Moattar, M.H.: Detecting spam tweets in twitter using a data stream clustering algorithm. In: 2015 International Congress on Technology, Communication and Knowledge (ICTCK). pp. 347–351 (2015)

[9] Evangelopoulos, N.E.: Latent semantic analysis. Wiley Interdisciplinary Reviews: Cognitive Science **4**(6), 683–692 (2013)

[10] Kim, Y.: Convolutional neural networks for sentence classification. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). pp. 1746–1751. Association for Computational Linguistics, Doha, Qatar (Oct 2014)

[11] Kokatnoor, S.A., Krishnan, B.: A comparative study of text mining algorithms for anomaly detection in online social networks. In: Jat, D.S., Shukla, S., Unal, A., Mishra, D.K. (eds.) Data Science and Security. pp. 29–37. Springer Singapore, Singapore (2021)

[12] Li, D., Guo, H., Wang, Z., Zheng, Z.: Unsupervised fake news detection based on autoencoder. IEEE Access **9**, 29356–29365 (2021)

[13] Li, D., Guo, H., Wang, Z., Zheng, Z.: Unsupervised fake news detection based on autoencoder. IEEE Access **9**, 29356–29365 (2021)

[14] Mangathayaru, N., Kumar, G.R., Narsimha, G.: Text mining based approach for intrusion detection. In: 2016 International Conference on Engineering & MIS (ICEMIS). pp. 1–5 (2016)

[15] Mitchell, R.: Web scraping with Python: Collecting more data from the modern web. " O'Reilly Media, Inc." (2018)

[16] Nassif, A.B., Talib, M.A., Nasir, Q., Dakalbab, F.M.: Machine learning for anomaly detection: A systematic review. IEEE Access **9**, 78658–78700 (2021)

[17] Pennington, J., Socher, R., Manning, C.: Glove: Global vectors for word representation. vol. 14, pp. 1532–1543 (01 2014)

[18] Petersen, K., Feldt, R., Mujtaba, S., Mattsson, M.: Systematic mapping studies in software engineering , (ease) 12 (pp. 1–10). In 12th International Conference on Evaluation and Assessment in Software Engineering (2008)

[19] Qader, W., M. Ameen, M., Ahmed, B.: An overview of bag of words;importance, implementation, applications, and challenges. pp. 200–204 (06 2019)

[20] Ramos, J.: Using tf-idf to determine word relevance in document queries (01 2003)

[21] Reimers, N., Gurevych, I.: Sentence-bert: Sentence embeddings using siamese bert-networks. arXiv preprint arXiv:1908.10084 (2019)

[22] Rodriguez, A., Tosyali, A., Kim, B., Choi, J., Lee, J.M., Coh, B.Y., Jeong, M.K.: Patent clustering and outlier ranking methodologies for attributed patent citation networks for technology opportunity discovery. IEEE Transactions on Engineering Management **63**(4), 426–437 (2016)

[23] Steyn, C., de Waal, A.: Semi-supervised machine learning for textual anomaly detection. In: 2016 Pattern Recognition Association of South Africa and Robotics and Mechatronics International Conference (PRASA-RobMech). pp. 1–5 (2016)

[24] Tharshini, M., Ragavinodini, M., Senthilkumar, R.: Access log anomaly detection. In: 2017 Ninth International Conference on Advanced Computing (ICoAC). pp. 375–381 (2017)

[25] Veeramreddy, J., Prasad, K.: Anomaly-Based Intrusion Detection System (06 2019)