IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Effectiveness Of Natural Language Processing Based Security Tools In Strengthening The Security Over Fin-Tech Platforms

[1]Ramakrishna Ramadugu, [2]Laxman Doddipatla

[1] Expert Business consultant, Finastra technologies
[2]Technology Engineer, PNC Bank

Abstract

The rapid evolution of financial technology (Fin-Tech) platforms has revolutionized the financial landscape but has simultaneously introduced significant cybersecurity challenges. Natural Language Processing (NLP)-based security tools have emerged as a promising solution to enhance security measures in these platforms. This study explores the effectiveness of NLP in identifying, mitigating, and preventing cyber threats, such as Fraudulent, Anomaly, and Unauthorized Access. This study explores the efficiency of NLP-based security tools in enhancing the security of fin-tech platforms by leveraging machine learning techniques such as Support Vector Machines (SVM), Random Forest (RF), and K-Nearest Neighbors (KNN). By analyzing patterns in textual data, these tools aim to detect vulnerabilities, predict threats, and mitigate risks in real-time. In this study, researchers used the financial data and textual data. The research evaluates the performance of these ML models, focusing on their accuracy ($A_{accuracy}$), precision ($P_{precision}$), recall (R_{recall}), and F1-score ($F1_{score}$). The experimental result shows that the proposed RF model shows the highest $A_{accuracy}$ (98.83%), $P_{precision}$ (97.70%), R_{recall} (98.64%), and $F1_{score}$ (98.17%).

Keywords: NLP, Financial technology (Fin-Tech), Machine Learning (ML), Cyber security

1. Introduction

Fin-Tech has had unlimited and acknowledged expansion throughout the previous decade. This is a completely new phrase that defines the financial technology industry [1]. Its primary goal is to regulate businesses and other organizations that focus on enhancing service quality through the use of Information Technology (IT). Some of the other technologies that contributed to the development of Fin-Tech are mobile embedded systems

and related networks [2-6], trust management [7,8], big data [9-11], data analytics methods [12,13], image processing, and cloud computing [14,15].

In recent years, fintech has digitised internet financial services. Independent financial services are provided to consumers by fin-tech enterprises [16]. As seen in Figure 1, the Fin-Tech sector offers a wide variety of digital financial services.

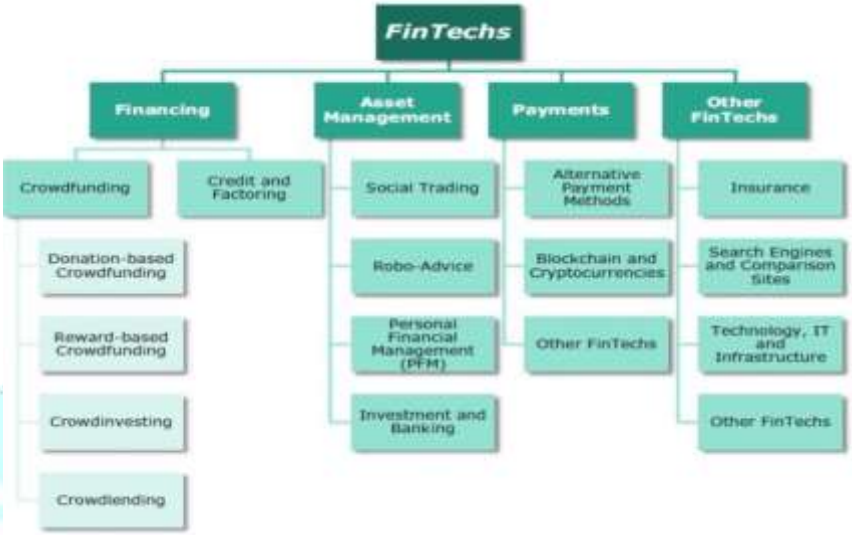


Figure 1: Fin-Tech Services [17].

It should be mentioned that Fin-Tech offers a variety of services to its consumers, as shown in Figure 1. However, the security and privacy concerns mostly arise from the IT applications that are used to deliver these services [18,19]. Furthermore, preserving confidentiality, integrity, authorisation, and availability are additional problems that Fin-Tech must address in order to perform its functions reliably [20]. In the absence of solutions to these problems, the reliability and efficiency of Fin-Tech integration of IT applications would remain in doubt [21]. Financial technology (FinTech) security and privacy problem mapping with technological solutions is shown in Figure 2.

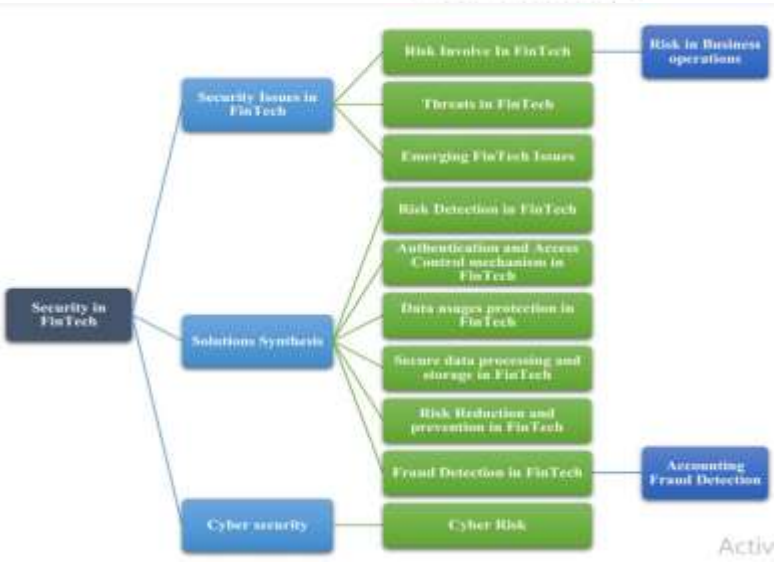


Figure 2: Mapping Fin-Tech security and privacy challenges and technology solutions [22].

The integration of Natural Language Processing (NLP) in security tools has significantly bolstered the security of financial technology (FinTech) platforms. FinTech systems, which often handle sensitive financial data and facilitate transactions, are prime targets for cyberattacks. NLP-based tools enhance these systems by detecting and mitigating potential threats in real time. They excel in analyzing vast amounts of unstructured data, such as customer communications, transaction records, and social media feeds, to identify anomalies or suspicious patterns indicative of fraud or cyber threats. Additionally, these tools enable robust user authentication through voice recognition and sentiment analysis, providing an additional layer of security. NLP algorithms also assist in identifying phishing attempts and malicious activities by interpreting the semantics of emails or messages. By automating the detection and response to threats, NLP-based security tools improve the resilience of Fin-Tech platforms, ensuring both user trust and compliance with regulatory standards. This technology is becoming an essential asset in the evolving landscape of digital financial services.

The aim of the paper based on the effectiveness of NLP based security tools in strengthening the security over fin-tech platform. The research objectives based on the study follow as:

- To assess how NLP techniques enhance the effectiveness of security tools in detecting and preventing security threats (such as fraud, unauthorized access, and anomalies) on fin-tech platforms.
- To evaluate and compare the performance of the SVM, RF, and KNN models in terms of key metrics, including $A_{accuracy}$, $P_{precision}$, R_{recall}, and $F1_{score}$, for identifying security threats on fin-tech platforms.
- To compare the effectiveness of different ML algorithms such as SVM, RF, and KNN in combination with NLP for security threat detection on fin-tech platforms.

2. Literature Review

In this section, the authors provide the previous work based on the effectiveness of NLP based security tools in strengthening the security over fin-tech platform using ML and DL models with some evaluation parameters such as Accuracy ($A_{accuracy}$), Precision ($P_{precision}$), Recall (R_{recall}), and F1-score ($F1_{score}$).

Sonawane et al. (2023) [23] evaluated the intersection between Industry 5.0 and blockchain-powered FinTech by assessing the benefits and challenges of using cryptocurrency in Indian FinTech, such as compliance with regulators and scalability issues. The rise of new business models, independent monetary ecosystems, and enhanced consumer experiences are among the noteworthy possible effects that are disclosed. To sum up, this study promotes further research into how blockchain could influence Indian businesses in the future, stimulate creative thinking, and provide insightful information on emerging trends in using blockchain for Indian FinTech outside the framework of Industry 5.0.

Jayasingh et al. (2023) [24] used ML to generate a Transaction Anomaly Detection (TAD) model for online transactions made by consumers using credit cards. In this study, a data collection from kaagle.com including 284807 records of online credit card transactions with a class designation is examined. The XGB classifier is one of the ML algorithms used in the suggested TAD model. We found that the best algorithm for this dataset is

the XGB Classifier, which identifies the fraudulent transactions with an $A_{accuracy}$ of 99.96% and a R_{recall} of 83%.

Lokanan et al. (2023) [25] used ML classifiers to forecast transactions identified as fraudulent in mobile money transfers. LR provides flexibility, Decision Trees (DT) give interpretability, and gradient descent ensures efficiency; together, these algorithms present strong and varied methodologies for addressing prediction challenges in the intricate domain of financial crimes. The RF classifier demonstrated exceptional performance because to its ability to learn from high-dimensional data and its capacity for optimisation, achieving an $A_{accuracy}$ of 0.96, R_{recall} of 0.80, and an $F1_{score}$ of 0.87.

Alwadain et al. (2023) [26] presented an innovative method for forecasting financial fraud via ML. Additionally, around 5000 additional data samples were produced using a Conditional Generative Adversarial Network for Tabular Data (CTGAN). Among the 27 classifiers, XG-Boost exhibited superior performance with an $A_{accuracy}$ score of 0.999. However, when subjected to rigorous repeated 10-fold cross-validation, the average $A_{accuracy}$ score for XG-Boost was 0.998.

Umer et al. (2023) [27] suggested an ensemble learning methodology for detecting fraudulent bitcoin transactions by combining two deep learning techniques: Convolutional Neural Network (CNN) and Long Short-Term Memory (LSTM). The performance of off-the-shelf CNN and LSTM, ensemble CNN, and ensemble LSTM using bagging and boosting techniques is evaluated based on $A_{accuracy}$ and losses from training and test datasets. The assessment findings demonstrate that the bagged LSTM ensemble method achieves a notable $A_{accuracy}$ of 96.4% and surpasses other techniques.

Xiong et al. (2022) [28] examined potential risks associated with Internet finance and the determinants influencing user behaviour in the context of big data. An Internet financial fraud detection model is developed based on the BP neural network (BPNN), accompanied with the establishment of relevant touch rules. The predictive performance is quantitatively compared with that of the SVM and RF algorithms. The financial fraud identification model using BPNN has the best $P_{precision}$ rate (88.14%), $A_{accuracy}$ rate (96.37%), R_{recall} rate (70.96%), and F-Score value (16.36) compared to the other two methods, while also demonstrating the lowest mistake detection rate (7.19%).

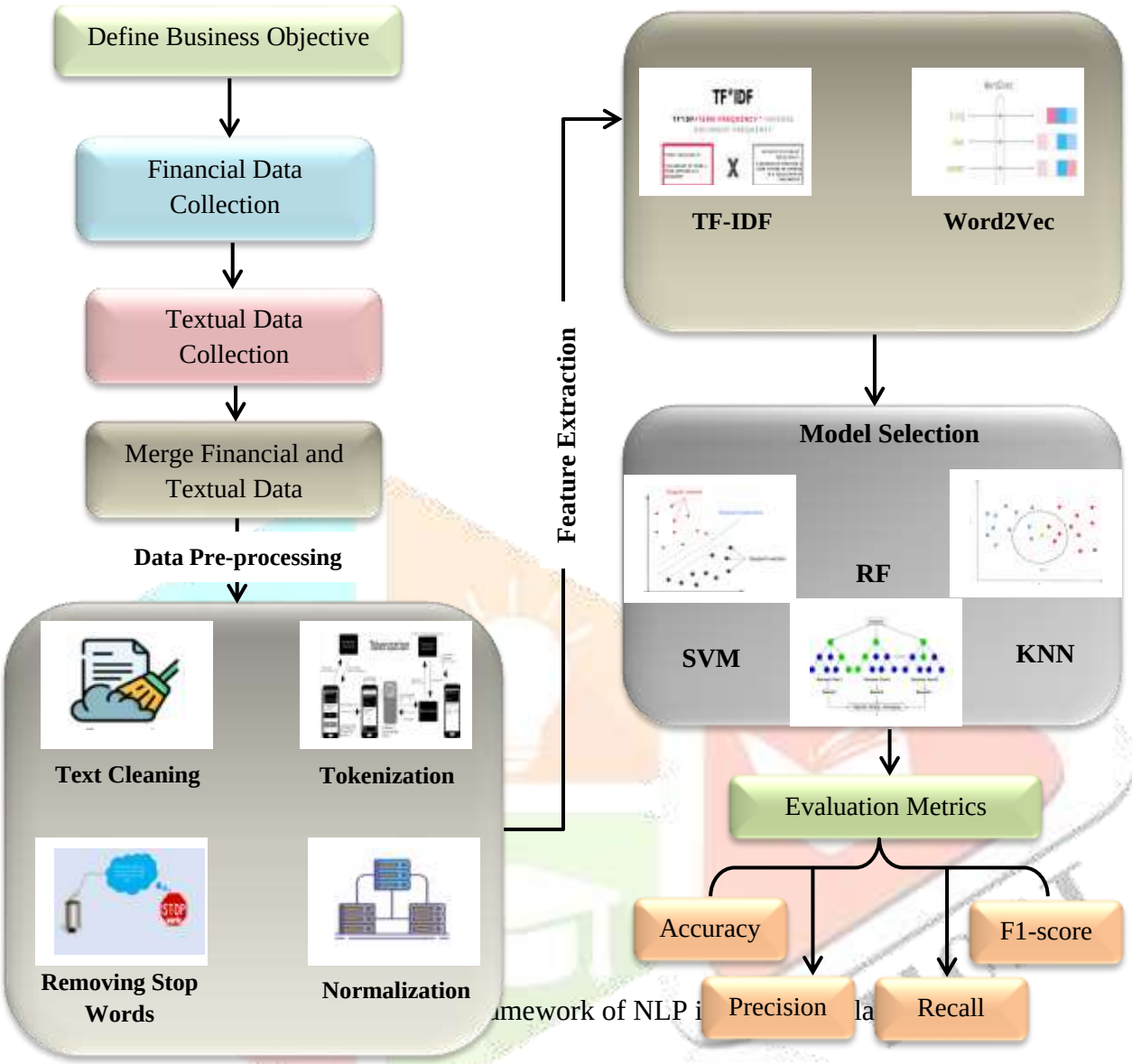
3. Problem Formulation

The rapid adoption of fin-tech platforms has revolutionized the financial industry, enabling efficient transactions, investment management, and access to financial services. However, this transformation has also introduced significant security challenges, such as data breaches, fraud, and unauthorized access to sensitive financial information. Traditional security measures are often insufficient to address the sophisticated and evolving nature of cyber threats targeting fin-tech platforms.

NLP-based security tools have emerged as a potential solution to enhance the security infrastructure of these platforms. These tools leverage NLP techniques to detect anomalies, analyze threats, and provide actionable insights by processing large volumes of unstructured data, such as user communications, system logs, and transaction details. Despite their potential, the effectiveness of NLP-based tools in strengthening the security of fin-tech platforms remains underexplored. This research seeks to evaluate the effectiveness of NLP-based security tools in identifying and mitigating cyber threats on fin-tech platforms.

4. Research methodology

The figure illustrates a framework that demonstrates the integration of NLP techniques to strengthen security in fin-tech platforms by leveraging financial and textual data. The process begins by defining the business objective, which serves as the foundation for collecting relevant financial and textual data. These datasets are then combined to ensure a comprehensive analysis. Next, a critical phase, data preprocessing, is conducted, which includes various sub-steps such as text cleaning, tokenization, removing stop words, and normalization. These steps refine and standardize the data to improve the accuracy and efficiency of subsequent NLP tasks. Feature extraction follows, employing methodologies like Term Frequency-Inverse Document Frequency (TF-IDF) and Word2Vec to transform the processed textual and financial data into numerical representations suitable for ML models. The transformed features are then passed to the model selection stage, where different ML algorithms, such as SVM, RF, and KNN, are evaluated for their effectiveness in identifying security threats. The performance of the selected models is assessed using evaluation metrics such as $A_{accuracy}$, $P_{precision}$, R_{recall}, and $F1_{score}$. These metrics ensure that the implemented security tools are reliable and efficient in safeguarding fin-tech platforms from vulnerabilities. This structured approach highlights the potential of NLP-based security tools in enhancing the robustness of financial technologies.



mework of NLP i la

4.1 Financial data Collection (Phishing Email Dataset)

Phishing email datasets are essential resources for analyzing and mitigating cyber security threats, particularly within the context of fin-tech platforms. These datasets typically consist of collections of phishing emails, which are fraudulent messages crafted to deceive recipients into revealing sensitive information such as login credentials, financial details, or personal identification [29]. A typical phishing dataset might include around 50,000 to 100,000 emails, with approximately 20% classified as phishing attempts. These records encompass various features, including email subject lines, sender information, body text, and metadata like timestamps and IP addresses. Table 1 show the dataset summary.

4.2 Textual Data Collection

The data includes a wide range of security-related text sources, such as financial transaction logs, user authentication protocols, phishing emails, malicious chat logs, and vulnerability reports. The collection process ensures the inclusion of both structured and unstructured data, enabling the study of how NLP techniques like

sentiment analysis, anomaly detection, and text classification can identify and mitigate threats. Textual data plays a critical role in strengthening the security of Fin-Tech platforms by providing insights that go beyond numerical analysis. Identifies unusual patterns, scams, and fraudulent activities in unstructured data like user complaints, transaction descriptions, or chat logs. Tracks and interprets textual user input to detect anomalies in behavior. The datasets are preprocessed through text cleaning, tokenization, and normalization to enhance their quality and usability for NLP model training and evaluation.

Table 1: Dataset Summary

Dataset Type	Number of Records	Unique Feature
Financial Data	50,000	50
Textual Data	8,000	5,000

4.3 Data Pre-processing

This stage ensures that the raw data is refined and structured, allowing for more accurate and meaningful outcomes when applying NLP-based security tools. The key steps involved in data preprocessing are:

- **Text Cleaning**: This step involves removing unnecessary or irrelevant components from the text, such as special characters, punctuation, numerical digits, and HTML tags. It ensures that the textual data is free of noise, making it suitable for analysis.
- **Tokenization**: Tokenization breaks down the cleaned text into smaller units called tokens, such as words or phrases. This step is essential for processing text at a granular level, enabling the NLP models to analyze individual words or sequences effectively.
- **Removing Stop Words**: Commonly used words that do not contribute to the contextual meaning of a sentence, such as "and," "the," or "is," are removed in this step. Eliminating these stop words reduces the dimensionality of the data and focuses the analysis on meaningful and significant terms.
- **Normalization**: This process standardizes the text to a uniform format. Techniques like converting all text to lowercase, lemmatization (reducing words to their base form), and stemming (removing affixes from words) are used to ensure consistency across the dataset. Normalization minimizes variations and helps the models recognize similar patterns more effectively.

4.4 Feature Extraction

In this section, the authors provide two feature extraction methods such as TF-IDF, and Word2vec.

4.4.1 Term Frequency - Inverse Document Frequency (TF-IDF)

TF-IDF is a prominent weighting approach used in feature extraction and selection, often employed in information retrieval due to its efficacy [30]. This technique evaluated the significance of a pre-existing term. A word's contribution value increases with its frequency in a text. However, if the term only appeared in a few publications, then the resultant contribution's value was reduced. TF-IDF integrates two distinct methodologies: TF, which quantifies the frequency of a word inside a text, and IDF, which assesses the prevalence of documents containing certain terms [31]. The TF-IDF formula is shown in Equation 1.

$$TF - IDF(t, d) = tf_{t,d} \times log\left(\frac{N}{df_t}\right) \quad (1)$$

Then, df_t is the number of documents that include the word t, N is the number of documents in the dataset, and $tf_{t,d}$ is the frequency of the word t occurring in dataset d. The TF-IDF method gives more weight to words that occur more often in a text and gives less weight to terms that appear less frequently [32].

4.4.2 Word2Vec

Word2vec was created, patented, and published in 2013 by a team of researchers led by Tomas Mikolov at Google [33]. Word2vec is a method for NLP. The word2vec approach uses a NN model to learn word connections from a massive text corpus. A model with this level of learning could suggest new words to add to a phrase and find synonyms automatically. Each word is assigned a unique collection of integers called a vector by word2vec, as the name implies. Given sufficient data, use, and contexts, Word2vec is able to generate very accurate definition predictions for words based on their past occurrences [34]. One possible application for these educated assumptions is to group and categorise documents according to their subject matter, or to find out the relationship between words (for example "man" is to "boy" and "woman" is to "girl").

4.5 Model Classifiers

In this section, the authors provide the classifiers for fin-tech platform including Support Vector Machine (SVM), Random Forest (RF), and K-Nearest Neighbors (KNN).

4.5.1 Support Vector machine

One supervised ML approach that has seen extensive application is SVM, which primarily deals with classification issues but can also handle regression. It has the capacity to identify intricate concealed patterns. SVMs are an optimised option for classification issues because to their stability and their ability to interact with kernel-based learning [35]. Support vector machines (SVMs) attempt to partition two-dimensional data into two classes using a hyperplane that optimally divides the classes. This optimises the margin between the two classes

by increasing the distance between their support vectors, which are the points closest to the hyperplane [36]. The SVM demonstration is shown in Figure 4.

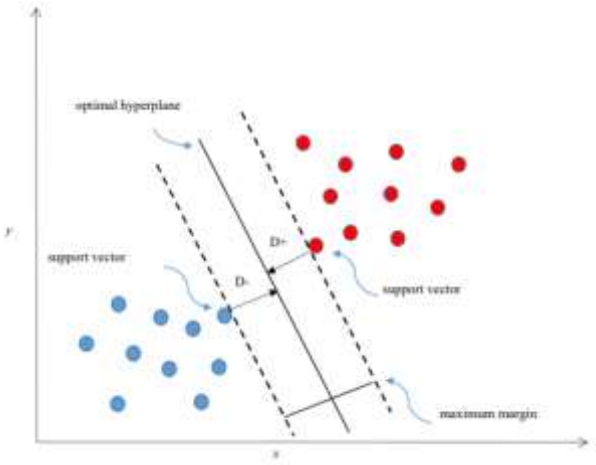


Figure 4: An illustration of SVM [37].

4.5.2 Random Forest

One kind of supervised learning method is RF, which uses a mix of tree predictors to make predictions. Classification and regression are two of its main uses [38]. Figure 5 show the workflow of RF. Among general-purpose learning approaches, it is thought to be among the most accurate. With its simple implementation and ability to handle many input variables without over-fitting, the RF is a great choice [39]. RF adds unpredictability to the model as the trees develop. It finds the best feature from a randomly selected selection of characteristics rather than the most important feature when splitting a node.

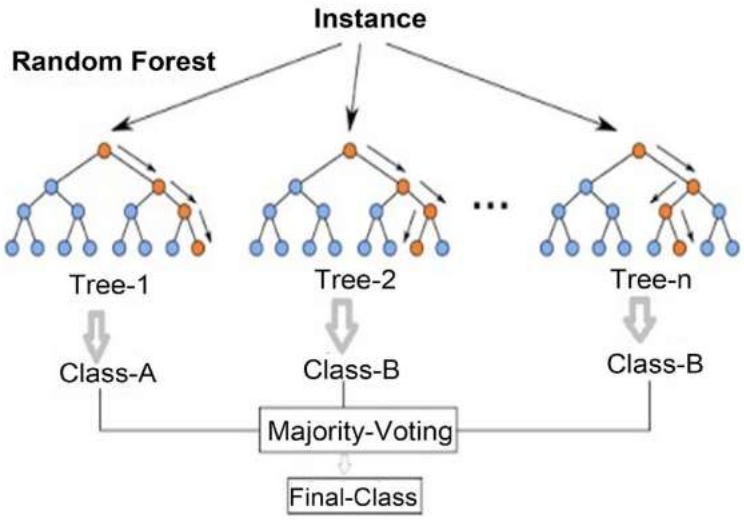


Figure 5: An illustration of RF [40].

4.5.3 K-Nearest Neighbors (KNN)

KNN is a method for supervised ML that can handle regression and classification tasks. Among the simplest and most basic categorization algorithms, it requires little in the way of processing power and time to run [41]. The principle behind the procedure is that items that are similar tend to cluster together; in other words, the closer two samples are, the more likely they are to belong to the same group. To begin, they need to know how many neighbors a particular point has, which is represented by the *k* parameter (Figure 6). Subsequently, it determines the distance of the newly added data to the sample data set using distance functions [42].

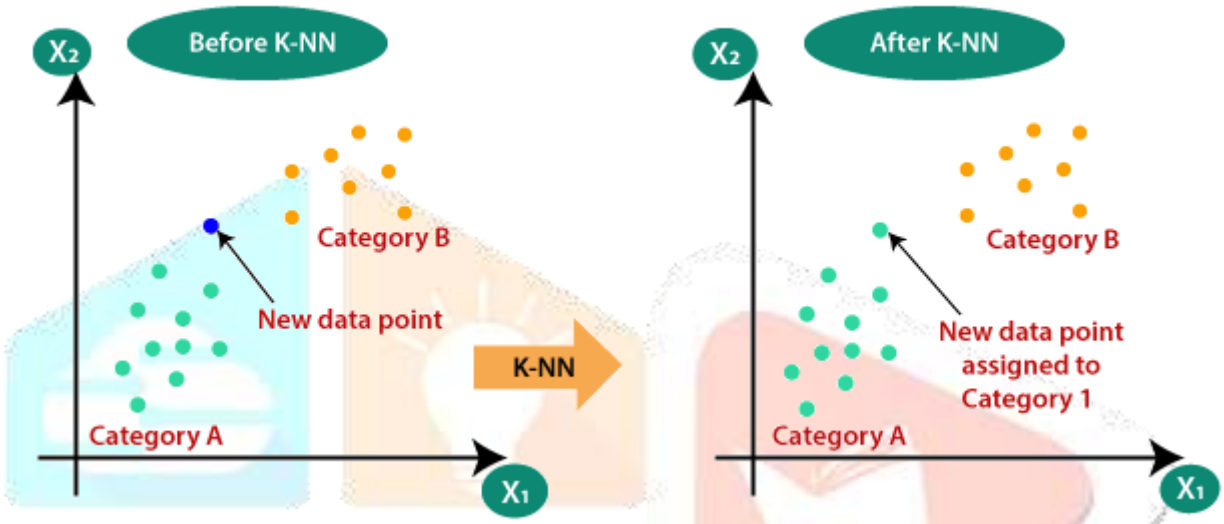


Figure 6: Illustration of the KNN algorithm [43].

4.6 Evaluation Metrics of the Model

Table 2 shows the confusion matrices. The following metrics are used to assess the proposed methodology:

Table 2: Confusion Matrices

	Actual positive	Actual negative
Predicted positive	True Positive (TP)	False Positive (FP)
Predicted negative	False Negative (FN)	True Negative (TN)

$$\boldsymbol{Accuracy} = \frac{\boldsymbol{TP+TN}}{\boldsymbol{TP+TN+FP+FN}} \quad (2)$$

$$\boldsymbol{Recall = Sensitivity} = \frac{\boldsymbol{TP}}{\boldsymbol{TP+FN}} \quad (3)$$

$$\boldsymbol{Precision} = \frac{\boldsymbol{TP}}{\boldsymbol{TP+FP}} \quad (4)$$

$$F1 - score = \frac{2*Precision*Recall}{Precision+Recall} \quad (5)$$

5. Result and Analysis

In this section, the authors provide the results based on the parameter $A_{accuracy}$, $P_{precision}$, R_{recall}, and F1score.

5.1 Feature Analysis

The table 3 above compares two popular text representation methods, TF-IDF and Word2Vec, in their ability to capture key features related to security threats, namely Fraudulent, Anomaly, and Unauthorized Access, within a fintech context. For the Fraudulent feature, Word2Vec outperforms TF-IDF, with a similarity score of 0.92 compared to 0.85 for TF-IDF. This suggests that Word2Vec's ability to understand contextual relationships between words in a vector space gives it an edge in recognizing fraudulent activities, as it can capture nuances and semantic relationships better than TF-IDF, which relies heavily on term frequencies in specific documents.

Similarly, for the Anomaly feature, Word2Vec again demonstrates a stronger performance with a score of 0.88, compared to TF-IDF's 0.75. This indicates that Word2Vec is more effective in detecting unusual patterns or behaviors within the data, as it can interpret anomalies based on a broader understanding of word context, whereas TF-IDF is more limited in this regard, focusing only on the frequency of terms without deep semantic comprehension.

Lastly, for Unauthorized Access, both models show relatively high scores, with Word2Vec at 0.91 and TF-IDF at 0.80. While both methods perform well in identifying unauthorized access attempts, Word2Vec's higher score again reflects its superior capacity to identify access-related patterns that cannot be immediately evident through simple term frequency analysis, further showcasing Word2Vec's advantage in capturing contextual meaning.

In summary, the feature analysis highlights that Word2Vec generally performs better than TF-IDF across all three security-related features, demonstrating its greater capacity to capture the semantic and contextual nuances necessary for effective security threat detection in fintech platforms. Figure 7 show the graph of Feature Analysis.

Table 3: Feature Analysis

Feature	TF-IDF	Word2Vec
Fraudulent	0.85	0.92
Anomaly	0.75	0.88
Unauthorized Access	0.80	0.91

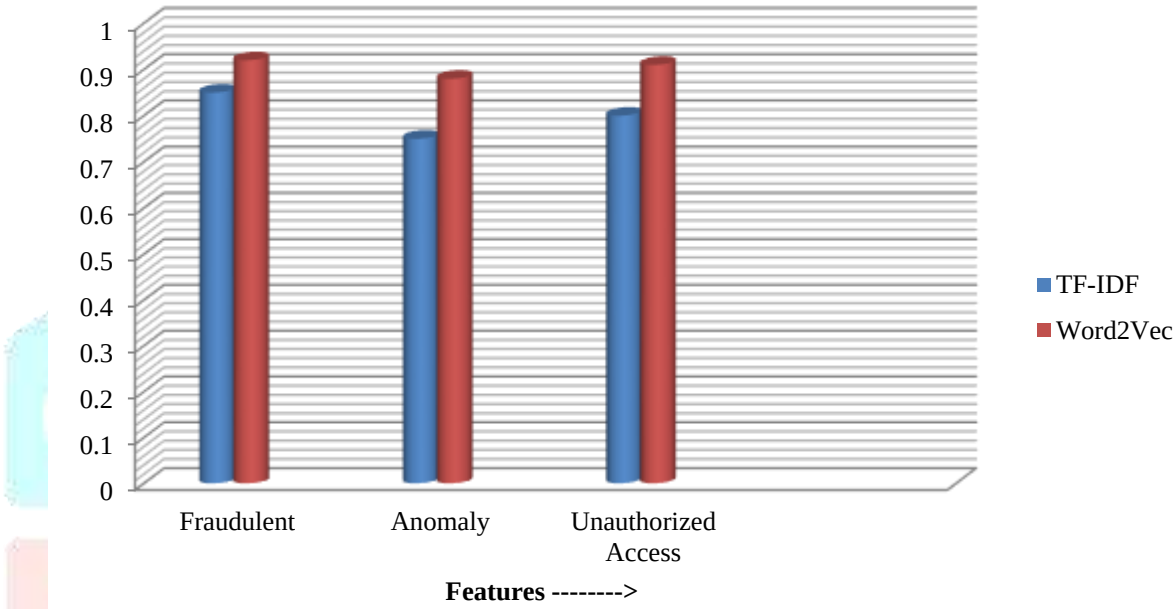


Figure 7: Feature Analysis

5.2 Support Vector Machine (SVM)

Table 4 above presents the performance metrics of a SVM model used for security threat detection, likely within the context of a fin-tech platform. The results indicate that the model achieves high effectiveness in identifying and classifying security threats.

With an $A_{accuracy}$ of 96.79%, the SVM model demonstrates strong general performance, correctly identifying security threats and non-threats with a high degree of reliability. A $P_{precision}$ of 94.27% indicates that, when the model predicts a security threat, it is highly likely to be correct, with relatively few false positives. The R_{recall} score of 95.74% shows that the model is very effective in identifying most security threats, missing only a small fraction of actual threats. Finally, the $F1_{score}$, which is the harmonic mean of $P_{precision}$ and R_{recall}, is 95%. Overall, these performance metrics illustrate that the SVM model is highly proficient in detecting security threats on fin-tech platforms, with excellent $A_{accuracy}$, $P_{precision}$, R_{recall}, and $F1_{score}$, indicating robust and reliable performance in real-world applications. Figure 8 show the confusion metrics of SVM model.

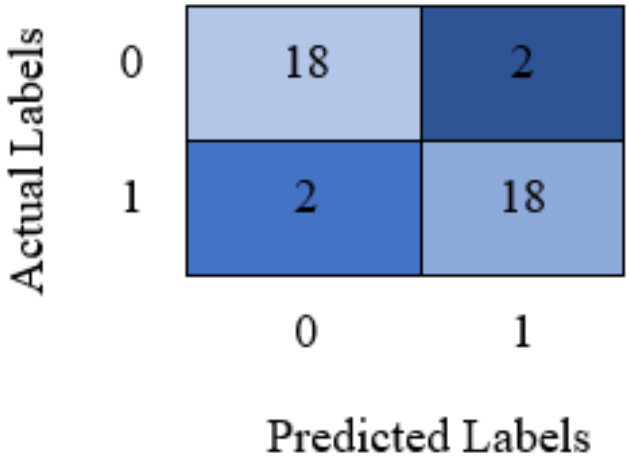


Figure 8: Confusion Metrics of SVM model

Table 4: The result of the SVM model

S.No.	Performance Metrics	Result
1	$A_{accuracy}$	96.79%
2	$P_{precision}$	94.27%
3	R_{recall}	95.74%
4	$F1_{score}$	95%

5.3 Random Forest (RF)

Table 5 presents the performance metrics of a RF model, which has been evaluated for its effectiveness in detecting security threats, likely in a fintech platform context. $A_{accuracy}$, at 98.83%, shows that the RF model achieves a very high rate of correct classifications, correctly identifying both positive (threat) and negative (non-threat) instances with great reliability. $P_{precision}$, with a value of 97.70%, demonstrates that when the model predicts a security threat, it is correct 97.7% of the time. R_{recall}, at 98.64%, reveals that the model is able to correctly identify 98.64% of all actual security threats in the data. This metric shows that the RF model misses very few true threats, performing exceptionally well in identifying most of the relevant instances. A high $F1_{score}$ of 98.17% indicates that the RF model achieves a good trade-off between $P_{precision}$ and R_{recall}, effectively minimizing both false positives and false negatives. Figure 9 show the confusion metrics of the RF model.

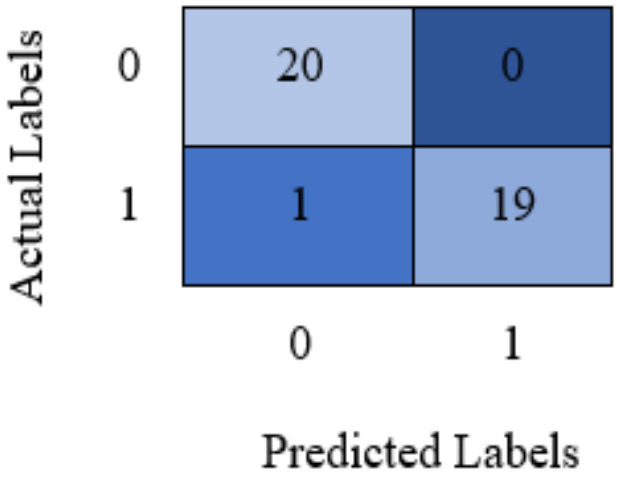


Figure 9: Confusion matrices of the RF model

Table 5: The result of the RF Model

S.No.	Performance Metrics	Result
1	$A_{accuracy}$	98.83%
2	$P_{precision}$	97.70%
3	R_{recall}	98.64%
4	$F1_{score}$	98.17%

5.4 K-Nearest Neighbors (KNN)

Table 6 presents the performance metrics of a KNN model, evaluating its effectiveness in detecting security threats, likely within a fintech platform environment. $A_{accuracy}$, at 97.29%, indicates that the KNN model is highly accurate, correctly classifying both security threats and non-threats the majority of the time. $P_{precision}$, with a score of 94.36%, shows that when the KNN model predicts a security threat, it is correct 94.36% of the time. R_{recall}, also at 97.29%, reveals that the KNN model is capable of identifying 97.29% of all actual threats. Finally, the $F1_{score}$ of 95.80% provides a balanced view of the model's performance, combining both $P_{precision}$ and R_{recall}. This metric demonstrates that the KNN model effectively balances its ability to correctly identify threats while minimizing false positives and negatives, making it a reliable tool for threat detection. Figure 10 show the confusion metrics of the KNN model.

In summary, the KNN model exhibits strong performance with high $A_{accuracy}$, $P_{precision}$, R_{recall}, and $F1_{score}$, making it a competitive option for detecting security threats on fin-tech platforms. Its performance, though slightly lower than some other models (like RF), is still impressive, demonstrating its ability to reliably identify both threats and normal behavior.

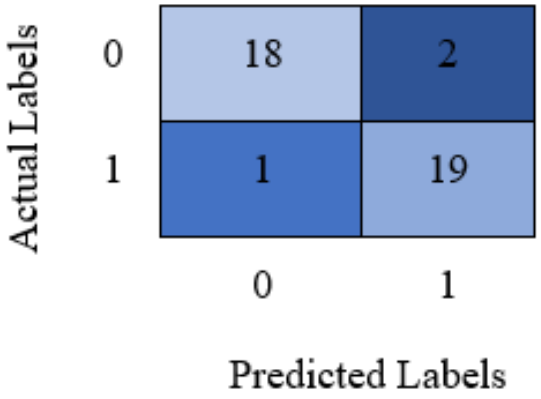


Figure 10: Confusion Metrics of KNN model

Table 6: The result of the KNN model

S.No.	Performance Metrics	Result
1	$A_{accuracy}$	97.29%
2	$P_{precision}$	94.36%
3	R_{recall}	97.29%
4	$F1_{score}$	95.80%

5.5 Comparison Analysis

They do a comparative analysis in this section, which is divided into two parts. The first part is a comparison of the mL methods with each other. In the second part, a comparative analysis of the best model with the approaches used in the previous study.

5.5.1 Comparison of the ML methods

Table 7 presents the performance metrics of three popular ML models—SVM, RF, and KNN—used for security threat detection, likely in the context of fin-tech platforms. $A_{accuracy}$ is a key indicator of overall model performance, and here, RF achieves the highest $A_{accuracy}$ at 98.83%, closely followed by KNN at 97.29% and SVM at 96.79%. This shows that RF performs the best at correctly classifying both threats and non-threats, while SVM is slightly less accurate, with KNN performing in between.

In summary, RF outperforms the other two models (SVM and KNN) across all metrics, making it the most effective model for security threat detection in this case. However, KNN and SVM still show strong performance, with KNN offering competitive R_{recall} and SVM providing high $P_{precision}$. The select of model would depend on the precise requirements of the task, such as whether minimizing false positives or capturing more true threats is prioritized. Figure 11 show the bar graph of comparison ML methods.

Table 7: Comparison of the ML methods

Method Used	$A_{accuracy}$ **(%)**	$P_{precision}$ **(%)**	R_{recall} **(%)**	$F1_{score}$ **(%)**
SVM	96.79	94.27	95.74	95
RF	98.83	97.70	98.64	98.17
KNN	97.29	94.36	97.29	95.80

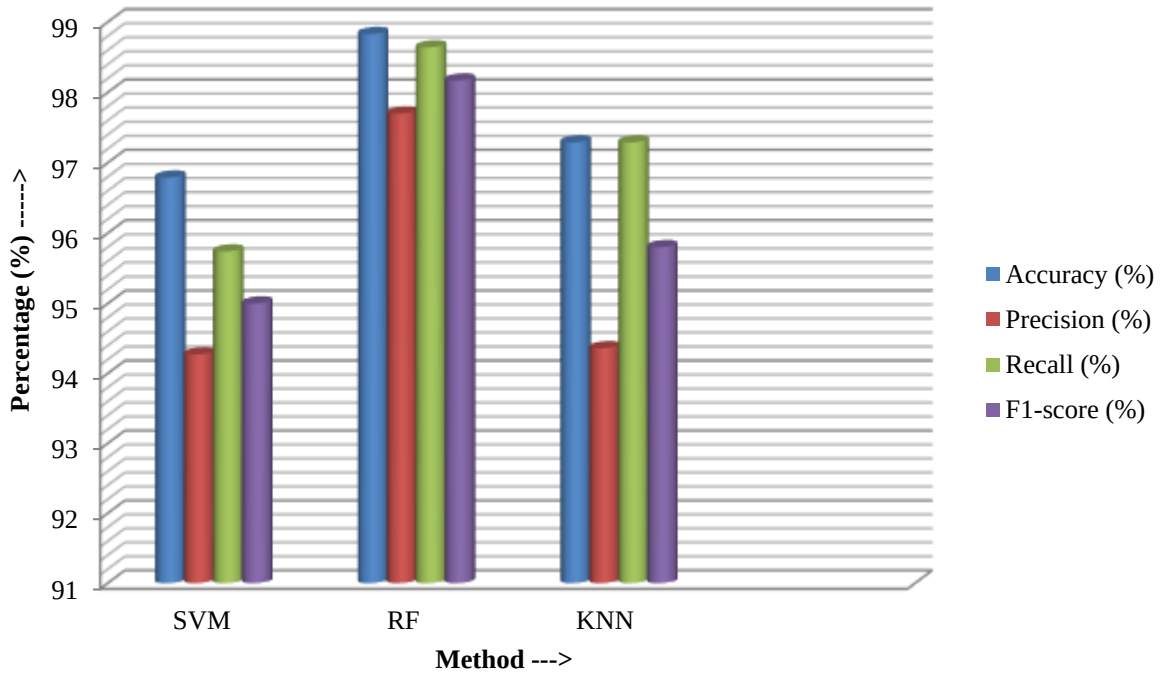


Figure 11: Comparison of the ML methods

5.5.2 Comparison of the best model with previous work

Table 8 compares the performance metrics of various models for security threat detection across different studies. Abdirahman et al. (2023) used LR with lower performance ($A_{accuracy}$: 77%, $P_{precision}$: 54%, R_{recall}: 57%, $F1_{score}$: 56%). Lokanan et al. (2023) applied RF, achieving high $A_{accuracy}$ (97%) and $P_{precision}$ (96%) but lower R_{recall} (80%) and $F1_{score}$ (87%). Xiong et al. (2022) used a BPNN with decent $A_{accuracy}$ (96.37%) but very low $F1_{score}$ (16.36%). In comparison, the proposed model (RF) outperforms all, achieving the highest metrics ($A_{accuracy}$: 98.83%, $P_{precision}$: 97.70%, R_{recall}: 98.64%, $F1_{score}$: 98.17%). Figure 12 show the bar graph of comparison best model with previous work.

Table 8: Comparison of best model with previous work

Reference	**Methodology Used**	$A_{accuracy}$	$P_{precision}$	R_{recall}	$F1_{score}$
Abdirahman et al., (2023) [23]	**LR**	77	54	57	56
Lokanan et al., (2023) [25]	**RF**	97	96	80	87
Xiong et al., (2022) [28]	**BPNN**	96.37	88.14	70.96	16.36
Work	**Proposed Model**	**98.83**	**97.70**	**98.64**	**98.17**

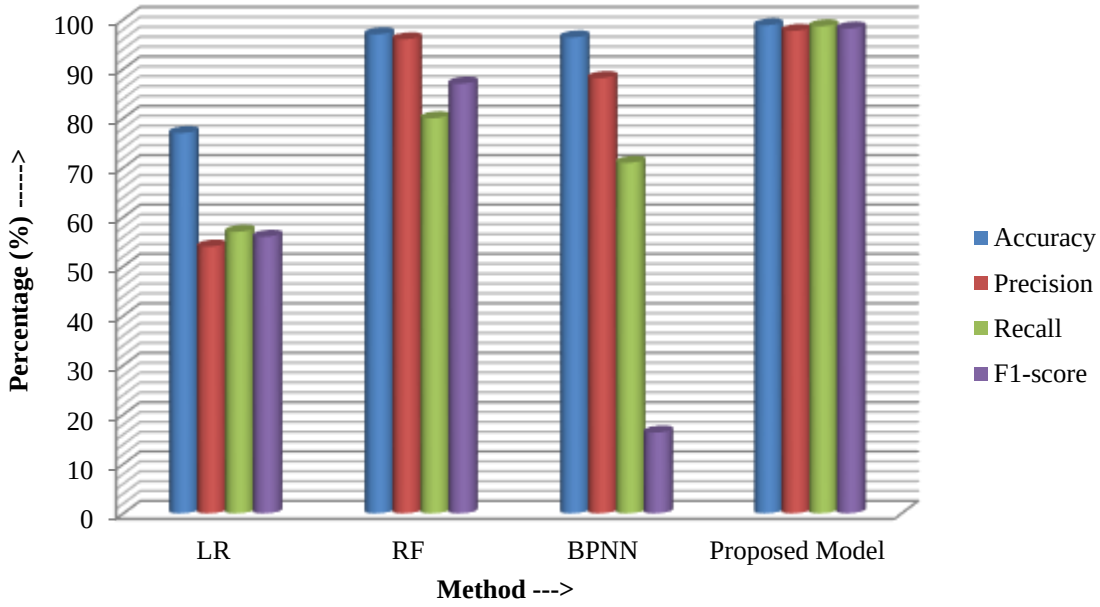


Figure 12: Comparison of best model with previous work

6. Conclusion

This study explores the effectiveness of NLP-based security tools in enhancing the security of fin-tech platforms by leveraging ML techniques such as SVM, RF, and KNN. The study demonstrates that NLP techniques, when combined with these ML models, offer powerful solutions for identifying and mitigating potential security risks in fin-tech platforms. Among the models tested, RF exhibited the best overall performance, achieving the highest $A_{accuracy}$ (98.83%), $P_{precision}$ (97.70%), R_{recall} (98.64%), and $F1_{score}$ (98.17%). This shows that RF can effectively balance the need to identify most security threats while minimizing false positives. SVM and KNN also performed well, with strong metrics, though they were slightly less effective compared to RF, especially in $P_{precision}$ and R_{recall}. The results suggest that NLP-based tools can significantly improve security by analyzing large volumes of unstructured text data (e.g., transaction logs, emails) and identifying anomalies indicative of fraudulent or malicious activities. These findings underscore the importance of integrating advanced ML techniques with NLP in developing more robust, efficient, and adaptive security systems for fin-tech platforms.

Reference

1. Meng, Weizhi, Liqiu Zhu, Wenjuan Li, Jinguang Han, and Yan Li. "Enhancing the security of FinTech applications with map-based graphical password authentication." Future Generation Computer Systems 101 (2019): 1018-1027.
2. Gai, Keke, Longfei Qiu, Min Chen, Hui Zhao, and Meikang Qiu. "SA-EAST: security-aware efficient data transmission for ITS in mobile heterogeneous cloud computing." ACM Transactions on Embedded Computing Systems (TECS) 16, no. 2 (2017): 1-22.
3. Zhang, Yan, Rong Yu, Shengli Xie, Wenqing Yao, Yang Xiao, and Mohsen Guizani. "Home M2M networks: Architectures, standards, and QoS improvement." IEEE Communications Magazine 49, no. 4 (2011): 44-52.
4. Wen, Sheng, Wei Zhou, Jun Zhang, Yang Xiang, Wanlei Zhou, and Weijia Jia. "Modeling propagation dynamics of social network worms." IEEE Transactions on parallel and distributed systems 24, no. 8 (2012): 1633-1643.
5. Gai, Keke, Meikang Qiu, Lixin Tao, and Yongxin Zhu. "Intrusion detection techniques for mobile cloud computing in heterogeneous 5G." Security and communication networks 9, no. 16 (2016): 3049-3058.
6. Zhang, Jun, Chao Chen, Yang Xiang, Wanlei Zhou, and Yong Xiang. "Internet traffic classification by aggregating correlated naive bayes predictions." IEEE transactions on information forensics and security 8, no. 1 (2012): 5-15.
7. Zhang, Qingchen, Laurence T. Yang, and Zhikui Chen. "Privacy preserving deep computation model on cloud for big data feature learning." IEEE Transactions on Computers 65, no. 5 (2015): 1351-1362.
8. Abawajy, Jemal, Guojun Wang, Laurence T. Yang, and Bahman Javadi. "Trust, security and privacy in emerging distributed systems." Future generation computer systems 55 (2016): 224-226.
9. Yin, Hua, and Keke Gai. "An empirical study on preprocessing high-dimensional class-imbalanced data for classification." In 2015 IEEE 17th international conference on high performance computing and communications, 2015 IEEE 7th international symposium on cyberspace safety and security, and 2015 IEEE 12th international conference on embedded software and systems, pp. 1314-1319. IEEE, 2015.
10. Plageras, Andreas P., Kostas E. Psannis, Christos Stergiou, Haoxiang Wang, and Brij B. Gupta. "Efficient IoT-based sensor BIG Data collection–processing and analysis in smart buildings." Future Generation Computer Systems 82 (2018): 349-357.
11. Psannis, Kostas E., Christos Stergiou, and Brij Bhooshan Gupta. "Advanced media-based smart big data on intelligent cloud systems." IEEE Transactions on Sustainable Computing 4, no. 1 (2018): 77-87.
12. Qiu, Meikang, Diqiu Cao, Hai Su, and Keke Gai. "Data transfer minimization for financial derivative pricing using Monte Carlo simulation with GPU in 5G." International Journal of Communication Systems 29, no. 16 (2016): 2364-2374.
13. Lee, Tae-heon, and Hee-Woong Kim. "An exploratory study on fintech industry in Korea: Crowdfunding case." In 2nd Int'l conf. on Innovative Engineering Technologies, pp. 58-64. 2015.
14. Stergiou, Christos, Konstantinos Psannis, Andreas P. Plageras, Yutaka Ishibashi, and Byung-Gyu Kim. "Algorithms for efficient digital media transmission over IoT and cloud networking." Journal of multimedia information system (2018).
15. Stergiou, Christos, Kostas E. Psannis, Byung-Gyu Kim, and Brij Gupta. "Secure integration of IoT and cloud computing." Future Generation Computer Systems 78 (2018): 964-975.
16. Okamura, Toshihiko, and Isamu Teranishi. "Enhancing FinTech security with secure multi-party computation technology." NEC Technical Journal 11, no. 2 (2017): 46-50.
17. Mehrotra, Anupam, and Satish Menon. "Second round of FinTech-Trends and challenges." In 2021 2nd International Conference on Computation, Automation and Knowledge Management (ICCAKM), pp. 243-248. IEEE, 2021.

18. Li, Guozhong, Jian Sheng Dai, Eun-Mi Park, and Seong-Taek Park. "A study on the service and trend of Fintech security based on text-mining: Focused on the data of Korean online news." Journal of Computer Virology and Hacking Techniques 13 (2017): 249-255.

19. Lim, Se Hun, Dan J. Kim, Yeon Hur, and Kunsu Park. "An empirical study of the impacts of perceived security and knowledge on continuous intention to use mobile fintech payment services." International Journal of Human–Computer Interaction 35, no. 10 (2019): 886-898.

20. Stewart, Harrison, and Jan Jürjens. "Data security and consumer trust in FinTech innovation in Germany." Information & Computer Security 26, no. 1 (2018): 109-128.

21. Hornuf, Lars, and Gregor Dorfleitner. "FinTech and Data Privacy in Germany." Machine Lawyering (2020).

22. Mehrban, Sobia, Muhammad Waqas Nadeem, Muzammil Hussain, Mohammad Masroor Ahmed, Owais Hakeem, Shazia Saqib, ML Mat Kiah, Fakhar Abbas, Mujtaba Hassan, and Muhammad Adnan Khan. "Towards secure FinTech: A survey, taxonomy, and open research challenges." Ieee Access 8 (2020): 23391-23406.

23. Sonawane, Swapnil S., and Dilip Motwani. "Blockchain-Powered FinTech: Shaping the Future of Indian Industries." In 2023 IEEE International Conference on Blockchain and Distributed Systems Security (ICBDS), pp. 1-7. IEEE, 2023.Jayasingh, Bipin Bihari, and G. Bhagya Sri. "Online transaction anomaly detection model for credit card usage using machine learning classifiers." In 2023 International Conference on Emerging Smart Computing and Informatics (ESCI), pp. 1-5. IEEE, 2023.

24. Lokanan, Mark E. "Predicting mobile money transaction fraud using machine learning algorithms." Applied AI Letters 4, no. 2 (2023): e85.

25. Alwadain, Ayed, Rao Faizan Ali, and Amgad Muneer. "Estimating financial fraud through transaction-level features and machine learning." Mathematics 11, no. 5 (2023): 1184.

26. Umer, Qasim, Jian-Wei Li, Muhammad Rehan Ashraf, Rab Nawaz Bashir, and Hamid Ghous. "Ensemble deep learning based prediction of fraudulent Cryptocurrency transactions." IEEE Access (2023).

27. Xiong, Tianlang, Zhishuo Ma, Zhuangzhuang Li, and Jiangqianyi Dai. "The analysis of influence mechanism for internet financial fraud identification and user behavior based on machine learning approaches." International Journal of System Assurance Engineering and Management 13, no. Suppl 3 (2022): 996-1007.

28. https://www.kaggle.com/datasets/naserabdullahalam/phishing-email-dataset

29. Ye, Jingyi, Xiaojun Jing, and Jia Li. "Sentiment analysis using modified LDA." In Signal and Information Processing, Networking and Computers: Proceedings of the 3rd International Conference on Signal and Information Processing, Networking and Computers (ICSINC) 3, pp. 205-212. Springer Singapore, 2018.

30. Yulietha, Irene, Said Faraby, and Adiwijaya Adiwijaya. "Klasifikasi Sentimen Review Film Menggunakan Algoritma Support Vector Machine." eProceedings of Engineering 4, no. 3 (2017).

31. Azzahra, Nadhia, Danang Murdiansyah, and Kemas Lhaksmana. "Toxic comment classification on social media using support vector machine and chi square feature selection." International Journal on Information and Communication Technology (IJoICT) 7, no. 1 (2021): 64-76.

32. Mikolov, Tomas. "Efficient estimation of word representations in vector space." arXiv preprint arXiv:1301.3781 3781 (2013).

33. Faisal, Mohammad Reza, Radityo Adi Nugroho, Rahmat Ramadhani, Friska Abadi, Rudy Herteno, and Triando Hamonangan Saragih. "Natural disaster on twitter: Role of feature extraction method of word2vec and lexicon based for determining direct eyewitness." Trends in Sciences 18, no. 23 (2021): 680-680.

34. Wah, Yap Bee, Hezlin Aryani Abd Rahman, Haibo He, and Awang Bulgiba. "Handling imbalanced dataset using SVM and k-NN approach." In AIP Conference Proceedings, vol. 1750, no. 1. AIP Publishing, 2016.

35. Botchey, Francis Effirim, Zhen Qin, and Kwesi Hughes-Lartey. "Mobile money fraud prediction—a cross-case analysis on the efficiency of support vector machines, gradient boosted decision trees, and naïve bayes algorithms." Information 11, no. 8 (2020): 383.

36. Mpanya, Dineo, Turgay Celik, Eric Klug, and Hopewell Ntsinjana. "Machine learning and statistical methods for predicting mortality in heart failure." Heart failure reviews 26, no. 3 (2021): 545-552.

37. Breiman, Leo. "Random forests." Machine learning 45 (2001): 5-32

38. Biau, Gérard. "Analysis of a random forests model." The Journal of Machine Learning Research 13, no. 1 (2012): 1063-1095.

39. Alghamdi, Bandar, and Fahad Alharby. "An intelligent model for online recruitment fraud detection." Journal of Information Security 10, no. 3 (2019): 155-176.

40. Taunk, Kashvi, Sanjukta De, Srishti Verma, and Aleena Swetapadma. "A brief review of nearest neighbor algorithm for learning and classification." In 2019 international conference on intelligent computing and control systems (ICCS), pp. 1255-1260. IEEE, 2019.

41. Li, Shuangjie, Kaixiang Zhang, Qianru Chen, Shuqin Wang, and Shaoqiang Zhang. "Feature selection for high dimensional data using weighted k-nearest neighbors and genetic algorithm." IEEE Access 8 (2020): 139512-139528.

42. K-Nearest Neighbor(KNN) algorithm for machine learning - JavatPoint. (n.d.). www.javatpoint.com.https://www.javatpoint.com/k-nearest-neighbor-algorithm-for-machine-learning