Hindawi
Security and Communication Networks
Volume 2021, Article ID 8869681, 12 pages
https://doi.org/10.1155/2021/8869681

Research Article

Design and Analysis of a Novel Authorship Verification Framework for Hijacked Social Media Accounts Compromised by a Human

Suleyman Alterkavı[1] and Hasan Erbay[2]

[1]*Computer Engineering Department, Engineering Faculty, University of Kırıkkale, Yahsihan, Kırıkkale 71450, Turkey*
[2]*Computer Engineering Department, Engineering Faculty, University of Turkish Aeronautical Association, Etimesgut, Ankara 06790, Turkey*

Correspondence should be addressed to Suleyman Alterkavı; sleman-terkawi@hotmail.com

Received 29 April 2020; Revised 15 December 2020; Accepted 12 January 2021; Published 23 January 2021

Academic Editor: Mamoun Alazab

Compromising the online social network account of a genuine user, by imitating the user's writing trait for malicious purposes, is a standard method. Then, when it happens, the fast and accurate detection of intruders is an essential step to control the damage. In other words, an efficient authorship verification model is a binary classification for the investigation of the text, whether it is written by a genuine user or not. Herein, a novel authorship verification framework for hijacked social media accounts, compromised by a human, is proposed. Significant textual features are derived from a Twitter-based dataset. They are composed of 16124 tweets with 280 characters crawled and manually annotated with the authorship information. XGBoost algorithm is then used to highlight the significance of each textual feature in the dataset. Furthermore, the ELECTRE approach is utilized for feature selection, and the rank exponent weight method is applied for feature weighting. The reduced dataset is evaluated with many classifiers, and the achieved result of the F-score is 94.4%.

1. Introduction

Online social networks (OSNs) are considered as essential sources of information and platforms that bring people together. The wide usage of the OSNs along with the crowds' self-confidence makes OSNs unprotected against hijackers, which might be accompanied by hijacking attacks that represent significant challenges in OSNs [1]. Originating mainly from the science of stylometry, which studies the writing style of the author, and the authorship verification counting features in the text to indicate the personality of the author [2], the goal of authorship verification is to determine whether two separate documents were written by the same author [3]; Rocha et al. in [4] defined authorship verification as two given tweets; a prediction is made as to whether or not they are from the same author. The OSNs that have accompanied hijacking attacks could be the following.

1.1. Fake Accounts. These accounts are counterfeit accounts injected in the OSNs to spread fake news, compromise systems, or make Sybil attacks. Thus, with the increasing popularity of the OSNs, the fake accounts are increasing, reaching up to 3 to 4% of Facebook accounts [5]. The Sybil attacks are discussed in [5–8].

1.2. Compromised Accounts by a Bot. Hackers control the accounts of genuine users through stealing the user's credentials [9] for phishing and spamming activities [10]. On 15 July 2020, hackers took control of the significant accounts on Twitter to spread a message request donation in Bitcoin and post crypto scam messages to the followers [11]. Compromising such verified accounts is attractive and motivating to the intruder because the user network, without a doubt, will positively interact with the hacker's requests, especially detecting the accounts that took five days. Moreover, 60% of the case studies were uncovered in the last days [12]. This kind of attack has been discussed in [8, 9, 13–17].

1.3. Compromised Accounts by Human Beings. Hackers gain access to the OSN's account through credentials information or misusing the account security features by the user. The intruder is disguised as a genuine user to post fake news or gain private information from the user's network, which affects the reputation and leads to economic loss [15]. Trang et al. [1] have mentioned many examples describing this kind of attack. This kind of attack has been discussed by [1, 12, 18]. The apparent anomalous behaviors of the fake accounts and compromised accounts by a bot are the key insights for the researchers to uncover [11, 17]. Twitter has shut down 70 million of these accounts in 2018. Unlike the compromised accounts by human beings, it is still a challenge [11], as there is no way to authenticate users' writing styles in the OSNs yet.

Motivated by the challenge of finding a method for profiling users through the most significant textual features that have existed in the short messages on OSNs to authenticate their writings with high accuracy, we propose a model throughout the study.

As a summary of the main steps of the study, a Twitter-based dataset has been crawled and manually annotated with the authorship information to evaluate the proposed model, which gathers some stylometry features from the tweets. Then, the XGBoost algorithm is implemented for feature extraction of the dataset. To further improve the classification results, one of the Multicriteria Decision-Making (MCDM) approaches, called ELECTRE, is employed for feature reduction. In addition, ELECTRE chooses the most important features and eliminates the features that have a negative impact on the performance of the classifiers. The selected features are assigned a weight according to their rank using the rank exponent weight method. Finally, they are given to different classifiers, but the logistic regression algorithm has achieved the highest performance. Later, the proposed model is compared with the traditional state of the models using the performance measures recall, precision, F-score, and accuracy.

The rest of this paper is divided into four sections. Section 2 presents the review of literature. A detailed explanation of the proposed approach is discussed in Section 3. Then, in Section 4, we shed light on the experimental results. Section 5 incorporates the conclusion of the paper and limitations of the study.

2. Literature Review

Egele et al. [13], by following a profile-based paradigm, have developed a system called COMPA to detect the inconsistent behavior in the compromised accounts through clustering the behavioral features that describe the genuine user idiolect, using Sequential Minimal Optimization (SMO) classifier, the model validated on Twitter, and Facebook datasets. COMPA shows low precision in the small-scale hijacking attacks [1].

Trång et al. [1], following an instance-based paradigm, have improved COMPA to detect single hijacking attacks through replacing rather than clustering the accounts according to similar behavior by classifying the account whether it is compromised or not. The improved version of COMPA shows lower-than-expected result in the single hijackings. The authors have suggested COMPA to include moving anomaly scores and stylometric features in future works.

Barbon et al. [19] have proposed a model to detect the compromised accounts through a bot by combining the textual features that describe the user's profile. In their model, the k-NN algorithm is used for classification purposes. Their model has been evaluated using the Twitter dataset that involved 1000 users. They have achieved classification accuracy rate over 93%, whereas the accuracy decreased in the Twitter accounts that are not used regularly. Therefore, the authors have suggested adding nonlinguistic features to improve the accuracy.

Lagerholm [20] has proposed to measure the similarity among benign and malicious user tweets. In the model proposed by Lagerholm, the basic feature set includes n-gram, term frequency-inverse document frequency (tf-idf), and Bag of Words; Long Short-Term Memory (LSTM) Neural Network, then, is used as classifier. The experimental evaluation of Lagerholm's scheme involved the tweets of eight different users with cross-topics, and the approach attained accuracy of 93.32%. Barbon et al. [19] have suggested using a dataset with related topics to make the model more applicable in real life. Another research in [21] has developed a system for continuous authentication, a combination of deep belief networks and Gaussian units that have been introduced for classification purposes. The proposed approach has been evaluated using short messages that consist of blocks of texts of 140, 280, and 500 characters based on Enron e-mails and Twitter feeds, yielding an EER ranging from 8.21% to 16.73% of different configurations.

Kaur et al. have introduced a model to quantify the dissimilarity in a text by known and unknown users. K-means algorithm is used for classification purposes. Their feature set has included Bag of Words (BOW), n-grams, folksonomy, and stylometric features. Their model has been evaluated using the public Twitter dataset that has scored 89.24% as classification accuracy rate [22]. A similar model has been developed by Seyler et al., using feature set that included statistical measures extracted from public Twitter dataset, which is classified using logistic regression classifier. Thus, the achieved accuracy for synthetic data is 85% [16].

Recently, Savyan and Bhanu [15] have proposed an unsupervised system for authorship verification named UbCadet, which analyzes the anomalous behavior of OSN's user through quantifying the similarity between tweet text, hashtag, time, and geolocation. UbCadet has been evaluated using Yelp and Twitter datasets. UbCadet system has produced an overall accuracy of 83.1% when analyzing the feature set of five users.

After reviewing the studies for authorship verification in the literature, there seems to be, reasonably, little research on authorship verification of compromised accounts by human beings, especially Twitter-based datasets, which has given the scholars the motivation to work on the current study. The present study is taking features reduction into account to select the most influential features and is noticing the high prediction accuracy of the machine learning (ML) model depending on extracting the most relevant features and applying the appropriate dimension reduction method [23]. The current paper improves the authorship verification accuracy of hijacked social media accounts compromised by a human by proposing a three-layered dimension reduction approach followed by ML algorithms to classify the messages. The main contributions of the manuscript are listed as follows:

(i) Creating and manually annotating the same genre and same topic on Twitter-based dataset with the authorship information for evaluating the proposed system.

(ii) Verifying the authorship of a tweet and combining lexical, syntactic, and semantic features that can be used effectively on any short text messaging service.

(iii) Conducting the traditional ML classifiers as metalearning algorithms to be used as a preprocessor in the feature selection process.

(iv) Applying the three-layered dimension reduction approach which includes the use of the surpassed metalearning algorithm as a preprocessor to measure the contribution of each feature in verifying the tweet's authorship and then ranking these features using the MCDM approach (ELECTRE), where the least influential feature set is disregarded. The remained features are assigned weights according to their ranks using the rank exponent weight method.

(v) Implementing different ML algorithms for message classification to achieve the highest performance.

The proposed model, shown in Figure 1, is detailed in Section 3.

3. The Model

The general flow of the proposed model is described in Figure 1, and the flowchart is illustrated in Figure 2.

The authorship verification process is passed through seven main subprocesses, as shown in Figure 1. The first step is to collect the users' tweets, tweets' history, and all available and related attributes. In the second step, the collected tweets are cleaned through eliminating the unused attributes and standardized tweets in a single format. Different textual features' vectors are extracted from the text and combined in one matrix to represent the tweets' corpus during the third step. In the fourth step, the surpassed classification algorithm among four different classifiers is selected as a preprocessor to quantify each feature's ability to verify the tweet's authorship. Then, the feature reduction is done by ranking the feature sets using the MCDM approach ELECTRE in the fifth step. In the sixth step, the remaining feature sets are assigned weights according to their ranks using the rank exponent weight method. Finally, the four different classifier methods are applied to the weighted feature sets to achieve the highest performance in verifying the message authorship. In the following subsections, a detailed explanation of model steps is given.

3.1. Collecting and Preprocessing the Data. With over 320 million active users, Twitter has become one of the most popular microblogging OSNs [5]. Unfortunately, there is not any publicly available standardized Twitter dataset for authorship verification studies [3]. So, it is crucial to introduce a Twitter dataset to facilitate authorship verification models. The way to obtain Twitter data is through its API [5]. However, Twitter allows retrieving a limited amount of data through the API, about 3200 tweets with 280 characters' block, and their related features through many batches [24]. Therefore, the crawler has been built using Python programming language and Tweepy library to form the dataset. The data is collected from different users with the same topic and same genre. Of course, in the real-life scenarios, the authors differ in the topics and genre (e.g., documents, e-mail, tweet, etc.), but the main challenge is to focus on the author's stylometry [25, 26]. Meanwhile, the information from the cross-topic or cross-genre could mislead the model [25], which makes the authorship verification difficult [27].

The Twitter dataset contains 16124 instances from different users, in which the retrieved objects related to the tweet are "ID" and "text" that represent the author's "ID" and the message, respectively. Considering the authorship verification as a one-class classification problem [28], the instances are labeled with 1 for the set of tweets from the known author and 0 for the set of tweets from the unknown author.

The collected data is preprocessed using simple regex to maintain the noise to precisely express the author's style. According to Rocha et al. [4], the dataset in the authorship verification should be preprocessed carefully. At the same time, eliminating or reshaping the corpus may impact the idiosyncrasies features of the author. In the preprocessing, the first phase is cleaning the retweets and replacing URLs with URL characters that do not affect the author's writing style [26]. The second phase is to replace punctuation marks, emojis, hashtags, percentages, and months with the metatags "!," "?," "#," "%," and "m," respectively. Moreover, the dataset is tokenized.

3.2. Feature Extraction. ML algorithms are designed to learn from numerical vectors with prespecified size but not text data containing characters' sequences with various sizes. Thus, the text data should be translated to numerical vectors before they progress. Herein, the extracted numerical features are illustrated below.

3.2.1. Lexical Feature. The lexical or linguistic attributes include all characters and word-based statistical measures extracted from the corpus [19, 22], independent from the language [29]. The lexical features could be extracted based on word level or character level [21]. In this study, the

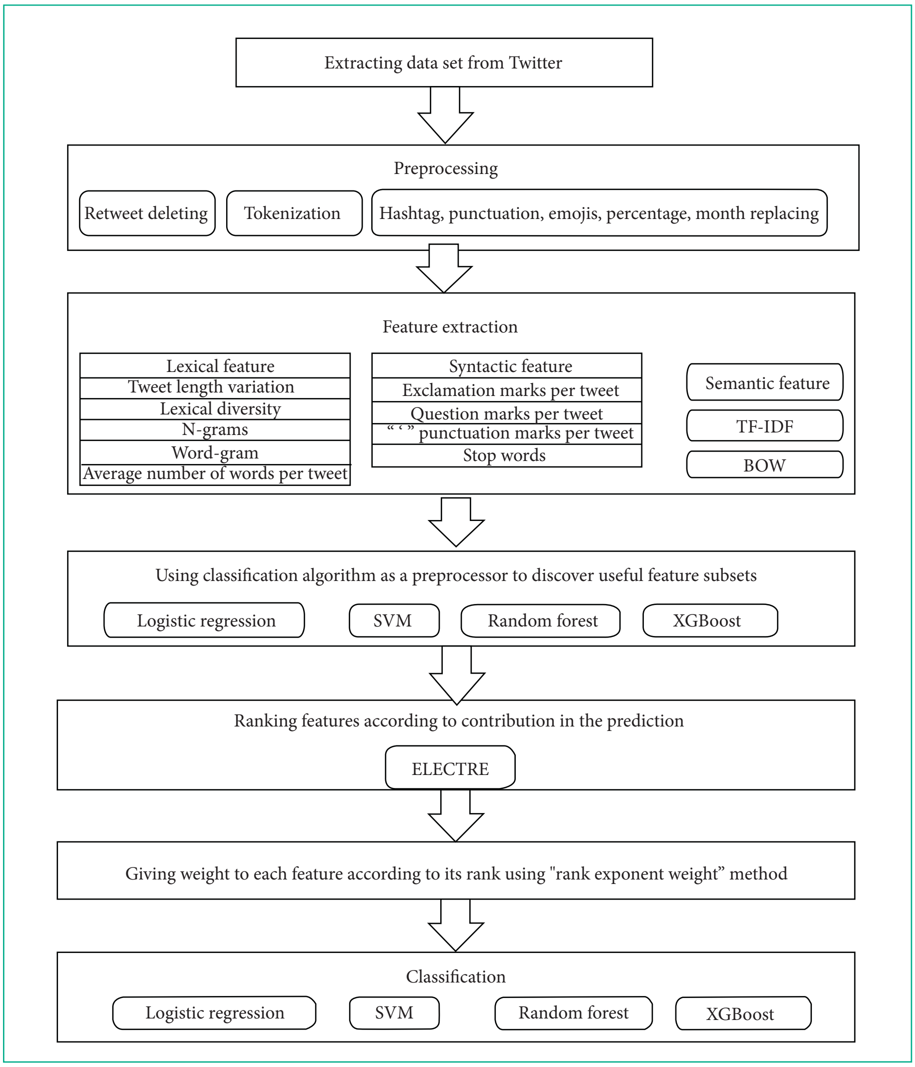
Extracting data set from Twitter
Preprocessing
Retweet deleting
Tokenization
Hashtag, punctuation, emojis, percentage, month replacing
Feature extraction
Lexical feature
Tweet length variation
Lexical diversity
N-grams
Word-gram
Average number of words per tweet
Syntactic feature
Exclamation marks per tweet
Question marks per tweet
" ' " punctuation marks per tweet
Stop words
Semantic feature
TF-IDF
BOW
Using classification algorithm as a preprocessor to discover useful feature subsets
Logistic regression
SVM
Random forest
XGBoost
Ranking features according to contribution in the prediction
ELECTRE
Giving weight to each feature according to its rank using "rank exponent weight" method
Classification
Logistic regression
SVM
Random forest
XGBoost

Figure 1: Graphical abstraction of the proposed model overview.

extracted character-level feature is tweet-length variation, while the word-level features are the average number of words per tweet and lexical diversities. One of the most important lexical features commonly used in the authorship verification literature is the *n*-gram [21]. This study used character 2-grams to consider the difference in the order of the characters among known versus unknown authors and word 2-grams to consider the multiword expressions [30], in addition, to keep the order of word pairs [20].

3.2.2. Syntactic Feature. The syntactic features describe author trait independent of context [21] via the punctuation that highlights the document boundaries to identify the sentences that could be tokenized [21]. Stop words and Part-of-Speech Tagger (POST) measures identify the function of the word in the context [29]. POST could be categorized as pronouns, prepositions, conjunctions, and auxiliary verbs, which grammatically describe the relationship between words in the corpus [31]. Function words, exclamation, question, and apostrophe marks per tweet features are used in this study.

3.2.3. Semantic Feature. The semantic features are used to understand the meaning of a word or sentence in the linguistic context and its relations with other linguistic units [29]. The semantic components extracted in the current study are word embedding, Bag of Words (BOW), and tf-idf.

Word embedding represents the text in numerical vectors, whereas the words with the same meaning have the exact representation. Thus, the words need to be vectored and combined to form the word embedding. Vector length is calculated using the features number that describes the word (e.g., suppose that there are 200 features; then the vector length is 200); the features number, then, is less than the total words number, and each feature value is between $[-1, 1]$; whenever the value is close to 1, it represents the word. The model's algorithm to make the word embedding is Word2vec, which is a pretrained word embedding technique that was developed by Mikolov in 2013 [32].

Mikolov et al. [33] have suggested two associated models that are used to represent vector of words from datasets. The first is the Continuous Bag-of-Words model which, unlike the traditional Bag-of-Words model, predicts the word

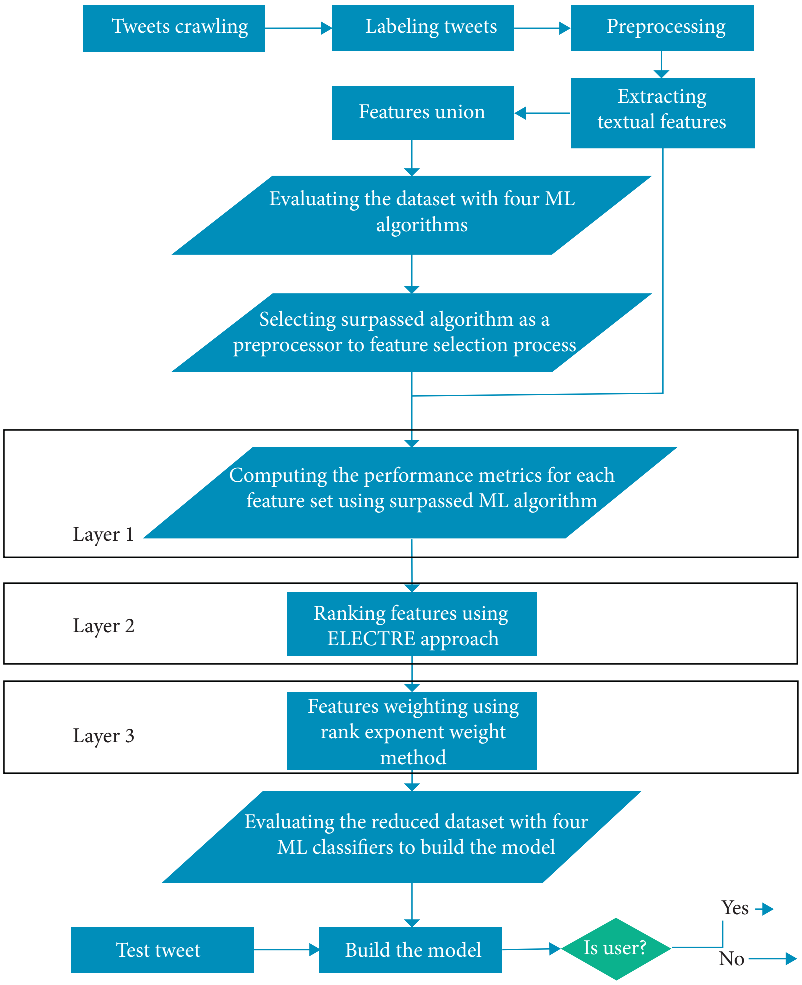
Tweets crawling
Labeling tweets
Preprocessing
Extracting textual features
Features union
Evaluating the dataset with four ML algorithms
Selecting surpassed algorithm as a preprocessor to feature selection process
Layer 1
Computing the performance metrics for each feature set using surpassed ML algorithm
Layer 2
Ranking features using ELECTRE approach
Layer 3
Features weighting using rank exponent weight method
Evaluating the reduced dataset with four ML classifiers to build the model
Test tweet
Build the model
Is user?
Yes
No

Figure 2: Flowchart for the proposed model.

according to the context, and the second is the continuous Skip-gram, which uses the input word to predict the surrounding words, as shown in Figure 3.

(i) Bag of Words. BOW is the count of token occurrences in the tweet, while tokenization splits the tweet into tokens and each token represents a word [34].

(ii) Term Frequency-Inverse Document Frequency. tf-idf gives more consideration to the importance of the word than its frequency in the corpus through calculating the term frequency in the tweet according to its frequency in all the tweets [35].

3.3. Selecting the Metalearning ML Algorithm. In the past decade, advances in computing power and available datasets are accompanied to lead to an increase in the variety of ML algorithms. Moreover, metalearning had made the algorithm selection and its parameters' tuning less complex [36]. According to Hospedales et al., metalearning transfers the experience to the machine learning model through many phases [37]. On the other hand, traditional ML algorithms, according to Lemke et al., such as SVM and K-Nearest Neighbor, are very successful in metalearning algorithm selection [36].

In order to lessen the feature size, which in turn improves the classifier accuracy, the extracted dataset has been evaluated using many metalearning algorithms such as Support Vector Machine (SVM), logistic regression, random forest, and XGBoost algorithms. On the other hand, different metrics might be used to measure the rate of recognition. Some are listed in Table 1; see [22, 23] for details.

Table 2 shows the experiments' results for the metalearning algorithms. The surpassed algorithm is used as a preprocessor in the feature selection process as well.

XGBoost algorithm has many parameters that could be tuned to avoid overfitting and get better accuracy such as Booster parameters, Max Depth, and Min Child Weights. More parameters descriptions can be found in [38].

To achieve the optimized accuracy of XGBoost algorithm through parameters' tuning, the experiments have increased the value of Max Depth parameter from 7 to 9. Table 3 reflects the change in the XGBoost performance.

3.4. Feature Selection. According to Table 3, the accuracy of XGBoost algorithm has surpassed those of other algorithms.

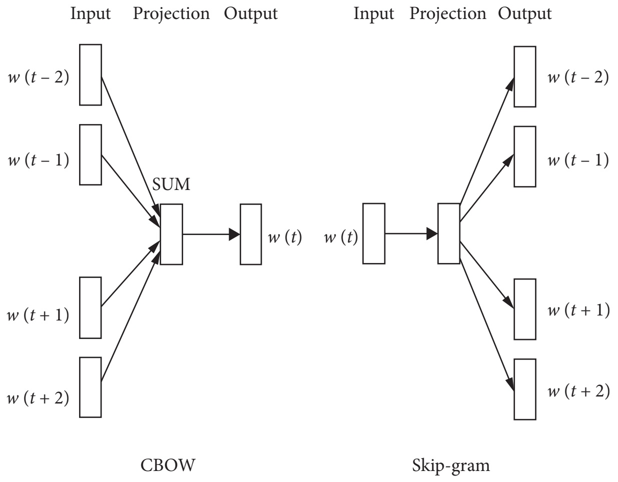

Figure 3: The CBOW and Skip-gram models' architecture [33].

Table 1: Some performance metrics.

Metric	Formula
Accuracy	(TP + TN)/ (TP + TN + FP + FN)
Recall	TP/ (TP + FN)
Precision	TP/ (TP + FP)
F-score	2 × (Precision × Recall)/ (Precision + Recall)

Table 2: The performance of some ML algorithms.

	Runtime (min)	Accuracy	Recall	Precision	F-score
Logistic regression	1.06	0.9029	0.962	0.90	0.939
SVM	0.99	0.9029	0.947	0.90	0.938
Random forest	0.63	0.7903	0.965	0.760	0.877
XGBoost	21.71	0.8932	0.9599	0.8896	0.933

Table 3: The performance is obtained from XGBoost algorithm when Max Depth parameter equals 9.

	Runtime (min)	Accuracy	Recall	Precision	F-score
Logistic regression	1.06	0.9029	0.962	0.90	0.939
SVM	0.99	0.9029	0.947	0.90	0.938
Random forest	0.63	0.7903	0.965	0.760	0.877
XGBoost	27.23	0.9049	0.967	0.902	0.94

Kotsiantis et al. have given the experimental recommendation to use ML algorithms as an introductory stage to discover the features' importance [39] Thus, XGBoost algorithm can be used to predict the contribution of each feature set in recognizing a tweet's authority, which is reflected in Table 4.

The ELECTRE method is used to select the most powerful features from Table 4. ELECTRE is an MCDM approach to rank many alternatives in the Multicriteria Decision-Making problems [40]. The ELECTRE method means the elimination and selection that reflects the truth [41]. It is developed as a philosophy to solve the complex decision-making problems with many alternatives and few criteria [42]. It is based on binary superiority comparisons between alternatives according to the appropriate criteria [43]. ELECTRE's [44] stepwise implementation is demonstrated as in the following parts.

3.4.1. Preparation of Decision Matrix. Table 4 represents the decision matrix, where the textual features in the first column are $m = 8$ alternatives to be ranked, and the measures in the first row are $n = 5$ criteria that are used to rank the alternatives.

3.4.2. Calculate the Normalized Decision Matrix. In order to make interattribute comparisons, the variation in the data scale between elements should be lesser, and normalization techniques scale down the elements to fall between 0 and 1. The following formula has been used to normalize decision matrix in Table 4:

$$x_{ij} = \frac{r_{ij}}{\sqrt{\sum_{i=1}^{n} r_{ij}^2}} \quad i = 1, 2, \cdots m, \quad j = 1, 2, \ldots, n. \tag{1}$$

Table 4: Comparison between the extracted textual features' performances.

	Runtime (min)	Accuracy	Recall	Precision	F-score
BoW	10.32	0.7867	0.91359	0.7789	0.8556
tf-idf	10.51	0.7844	0.9168	0.7766	0.8548
Word 2-gram	20.96	0.7257	0.9347	0.7070	0.8251
Char 2-gram	18.61	0.87363	0.9380	0.8718	0.9112
Stop words	73.64	0.7867	0.9135	0.7789	0.8556
Lexical	27.88	0.7748	0.8997	0.7520	0.8609
Syntactic	28.88	0.7748	0.8997	0.7520	0.8609
Semantic	0.66	0.7848	0.9006	0.7653	0.8674

3.4.3. Definition of the Criteria's Weights. The decision-maker assigns weight to each criterion to express its importance according to other criteria, but Saaty has developed a scale to gain the weight [45] as shown in Table 5.

The matrix of criterion importance according to other criteria shown in Table 6 is developed using Kaur et al.'s experiment [22] in addition to the runtime criterion, which gained strongly important weight.

Normalizing the matrix of the criteria importance gives the resultant matrix obtained in Table 7. Criteria weights shown in Table 8 are calculated by averaging the normalized importance for each criterion according to the other criteria in the exact row.

Table 5: Saaty's scale.

Meaning	Scale
Equally important	1
Slightly important	2
Moderately important	3
Above moderate	4
Strongly important	5
Above strong	6
Very strong	7
Highly important	8
Extremely important	9

3.4.4. The Calculation of the Weighted Normalized Decision Matrix. Table 8 contains $W = (w_1, w_2, \ldots, w_n)$, the weight vectors for the criteria, where $w_j \geq 0, \sum_{j=1}^{n} w_j = 1$, and the weighted normalized decision matrix is calculated by multiplying the normalized decision matrix obtained in Step 2 with criteria weights.

3.4.5. Determine the Discordance and Concordance Set. Elements of every pair in the weighted normalized decision matrix are compared and the concordance set will contain the best or equal elements of each alternatives pair, determined by the following relationship:

$$C(p,q) = \left\{ j, \quad v_{pj} \geq v_{qj} \right\}. \tag{2}$$

And the discordance set will contain the worst elements of each alternatives pair, determined by the following relationship:

$$D(p,q) = \left\{ j, \quad v_{pj} < v_{qj} \right\}. \tag{3}$$

3.4.6. Calculate the Concordance and Discordance Matrix. The concordance matrix is calculated by adding the elements weights of concordance set:

$$C_{pq} = \sum_{j^*} w_j^*. \tag{4}$$

The discordance matrix is calculated by dividing the sum difference of discordance set elements by the sum difference of criteria elements.

$$D_{pq} = \frac{\left(\sum_{j^0} \left| v_{pj^o} - v_{qj^0} \right| \right)}{\left(\sum_{j} \left| v_{pj} - v_{qj} \right| \right)}. \tag{5}$$

3.4.7. Make Calculations of the Advantages. The averages of discordance and concordance are calculated, stating "yes" for the values in the concordance matrix either bigger than or equal to concordance average that obtains the concordance index matrix, while stating "no" for the values in the discordance matrix less than or equal to discordance average that obtains the discordance index matrix.

3.4.8. Calculate Net Superior and Inferior Values. The alternatives are ranked according to the net superior and inferior values. The following formulas are used to calculate them:

$$C_p = \sum_{\substack{k=1 \\ k \neq p}}^{m} C_{pk} - \sum_{\substack{k=1 \\ k \neq p}}^{m} C_{kp}. \tag{6}$$

$$D_p = \sum_{\substack{k=1 \\ k \neq p}}^{m} D_{pk} - \sum_{\substack{k=1 \\ k \neq p}}^{m} D_{kp}. \tag{7}$$

Table 9 demonstrates the textual features' superior ranking based on the ELECTRE method.

ELECTRE's ranking result has demonstrated the best ranking for the semantic feature. Furthermore, the last rank is the stop words feature, while the others obtain ranking based on their priority to verify the tweet's authorship as, respectively, exposed in Table 9.

Table 6: The importance of each criterion according to other criteria.

	F-score	Precision	Recall	Accuracy	Runtime
F-score	1	2	3	4	5
Precision	0.5	1	2	3	4
Recall	0.33	0.5	1	2	4
Accuracy	0.25	0.33	0.5	1	5
Runtime	0.2	0.25	0.25	0.2	1

Table 7: Normalized criteria importance matrix.

	F-score	Precision	Recall	Accuracy	Runtime
F-score	0.43796	0.489795918	0.44444444	0.39215686	0.26316
Precision	0.21898	0.244897959	0.2962963	0.29411765	0.21053
Recall	0.14599	0.12244898	0.14814815	0.19607843	0.21053
Accuracy	0.10949	0.081632653	0.07407407	0.09803922	0.26316
Runtime	0.08759	0.06122449	0.03703704	0.01960784	0.05263

Table 8: Criteria weights.

Metric	Value
F-score	0.405502
Precision	0.252963
Recall	0.164637
Accuracy	0.125279
Runtime	0.051618

Table 9: Ranking table.

Alternatives	Ranking
BoW	5
tf-idf	4
Word 2-gram	2
Char 2-gram	7
Stop words	8
Lexical	3
Syntactic	6
Semantic	1

4. Results and Discussion

While the features selection determines the used features in the prediction features, weighting indicates the importance of the selected features by assigning different weights according to their priority [46]. Feature weighting is used in [15, 47].

Ranking textual features using the ELECTRE approach highlighted the differentiated performance of them. Furthermore, to eliminate the stop words features in the Feature Selection section, a weight is assigned to each feature based on its rank using the rank exponent weight method [48], which is defined by

$$w_j(\mathrm{RE}) = \frac{\left(n - r_j + 1\right)^p}{\sum_{k=1}^{n} \left(n - r_k + 1\right)^p}, \tag{8}$$

where r_j is feature rank, n is number of features, and P is the most important criterion weight.

To build the dataset, 16124 tweets from different Twitter users have been crawled, and the extracted features from cleaned tweets have represented the dataset attributes. Computers with 16 GB RAM and Python 3 have been used to perform the experiments. K-fold is used as a cross-validation type to index the training and test sets in each fold iteration.

Combining the weighted ranked values of the features to train classifiers has demonstrated the prediction to the tweet's authorship verification in Table 10.

Table 10 illustrates the performance of the logistic regression, SVM, random forest, and XGBoost algorithms in verifying the tweets' authorship based on the measures runtime, accuracy, recall, precision, and F-score. The comparative analysis of classifiers, after the feature selection and weighting using the ELECTRE approach and rank exponent weight method, highlights the fact that the performance of the logistic regression algorithm outperforms those of the other ML classifiers. Hence, using the logistic regression in the classification step in our model, it can be relied upon as a constructive approach to verify the tweet's authorship.

Figure 4 illustrates the performance evaluation of logistic regression classifier before applying the suggested model after the feature selection process using the ELECTRE approach and after feature selection and weighting. It is evident from the figure that applying the suggested model yields better performance in comparison to logistic regression without feature selection and weighting and logistic regression with just feature selection. With a higher value of accuracy, recall, precision, and F-score successively than the others, it helps to justify the superiority of the suggested model, while the proposed model improves the accuracy from 90.29% to 91.1%, which means that more tweets are recognized correctly.

Figure 5 illustrates the application of feature selection using the ELECTRE approach and rank exponent weight method lessening the training time for the logistic regression classifier to verify the tweet's authorship.

Figure 6 compares the performance in verifying the tweet's authorship with the suggested model versus the UbCadet model in [15]. It is observed that, for the measures (accuracy, precision, and F-score), the suggested model

Table 10: Prediction measures for each algorithm without stop words feature and after weighting.

Method	Runtime (min)	Accuracy	Recall	Precision	F-score
Logistic regression	0.51	0.911	0.9699	0.911	0.944
SVM	0.47	0.897	0.94	0.897	0.934
Random forest	0.72	0.79	0.98	0.79	0.879
XGBoost	14.12	0.895	0.967	0.895	0.935

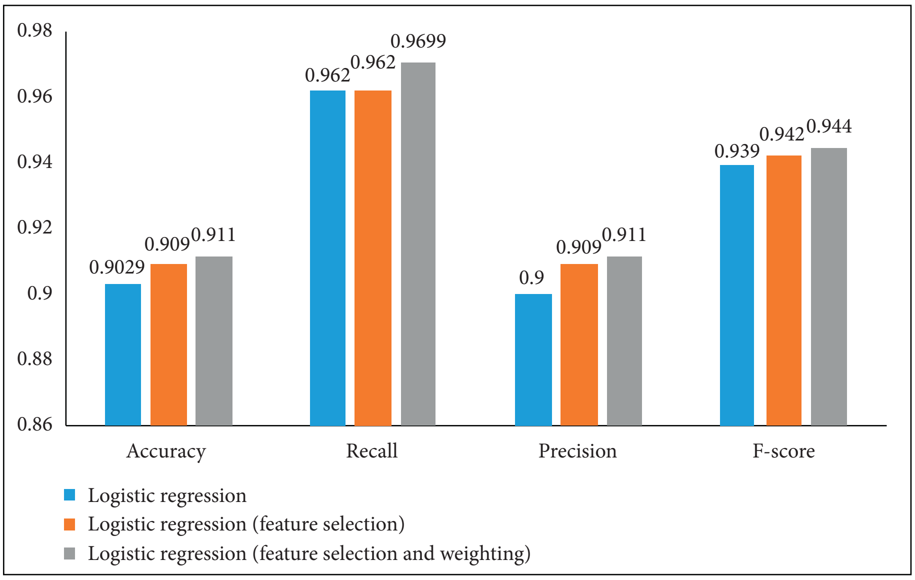

Figure 4: Performance evaluation of logistic regression classifier.

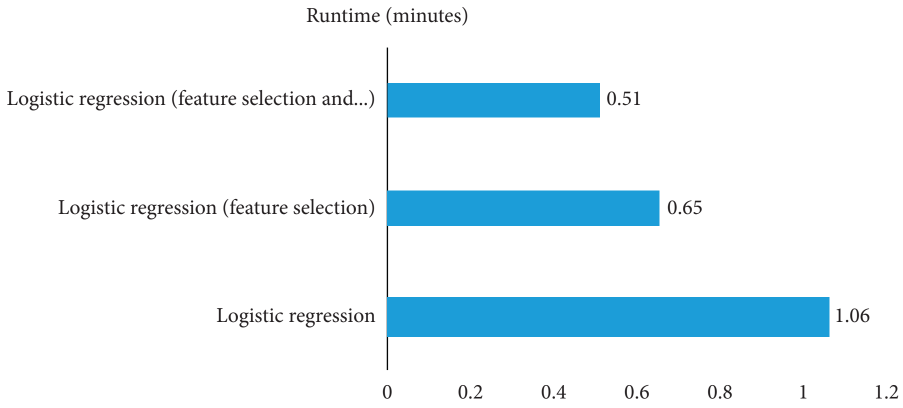

Figure 5: Training time comparison for logistic regression classifier in the suggested model.

yields higher values in comparison to the UbCadet model. However, in case of recall in the UbCadet model, the highest value is 1 compared to 96.99% with reference to the suggested model.

For five user accounts with anomalous tweets, UbCadet correctly recognizes 83.1% of the tweets, whereas the suggested model correctly recognizes 91.1%, reflecting the reliability of the suggested model in decision-making about verification of the authorship of the tweets.

The proposed model highlights the following pertinent results:

(i) The implementation of ELECTRE approach, in addition to rank exponent weight method, enhances the performance of the logistic regression and XGBoost algorithms and also slightly deteriorates the performance measures for SVM and random forest algorithms, as illustrated in Table 10.

(ii) The model implementation lessens the training time for ML classifiers.

(iii) The implementation of the ELECTRE approach alone is to highlight the most important features,

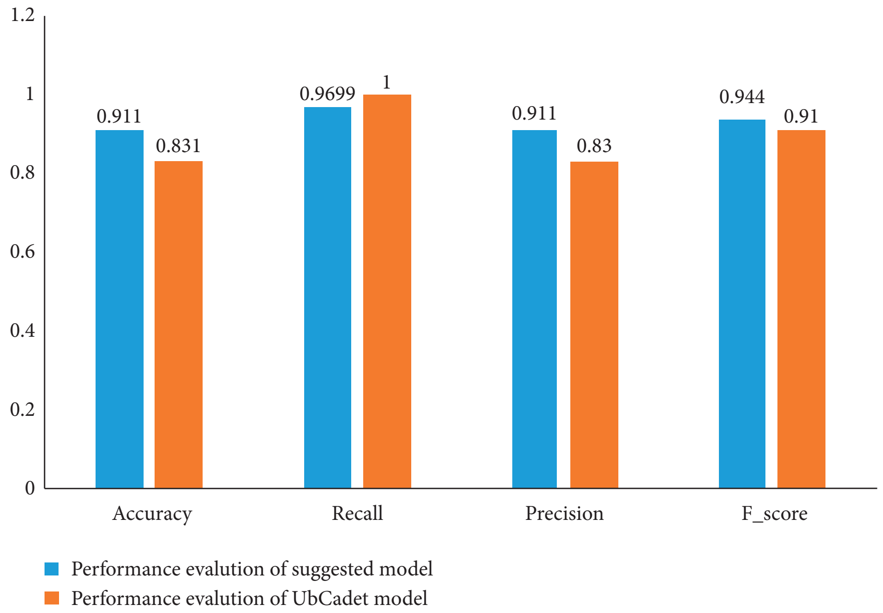

Figure 6: Comparison of the suggested model performance with the UbCadet model [15].

slightly enhancing the performance of the logistic regression as compared to before applying the suggested model as illustrated in Figure 4.

While the ELECTRE approach yields high performance rate for certain ML classifiers, it is considered a limitation of our study, where the weight assigned by the decision-maker to determine the criteria importance relies on subjective inputs by the decision-maker [43]. Incentivized collusion networks mentioned in [17] are considered as a limitation as they impede the authorship verification process, where users get paid to publish promotional messages on their accounts, modifying the textual features in the text, indicating the personality of the author.

5. Conclusion

The present study proposes a hybrid model for authorship verification in OSNs. Due to the nonavailability of standardized Twitter dataset publicly for authorship verification purpose, a same-genre and same-topic Twitter-based dataset is crawled and manually annotated with the authorship information. Successive preprocessing steps were performed to prepare the dataset for features extraction. Hence, a three-layered dimension reduction approach has been initiated. At the outset, XGBoost algorithm was selected as a preprocessor to calculate the textual features' performance in verifying the tweet's authorship on each criterion. Hence, the MCDM approach ELECTRE is used to solve the features selection problem. At the methodological level, it is the first application of ELECTRE to this domain, which is expected to be useful for similar problems. Based on Kaur et al.'s experiment [22], most criteria and their relative weights are obtained. ELECTRE uses the pairwise comparisons to rank eight textual features extracted from tweets. Further, the rank exponent weight method has been used for weighting the selected features.

The reduced dataset performs evaluation with four ML classifiers, wherein two of them reflect enhancing performance compared with traditional ML in terms of the runtime, accuracy, recall, precision, and F-score. The experiment analysis for the proposed model with the logistic regression classifier reflects a high result of F-score reaching 94.4% for block sizes of 280 characters in verifying the tweet's authorship, which can extend the model implementation on another classification problem with high feature number as future work.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Disclosure

The author Suleyman Alterkavı (Sleman AlTerkawi) has a dual citizenship, so his name is written in two different ways.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] D. Trång, F. Johansson, and M. Rosell, "Evaluating algorithms for detection of compromised social media user accounts," in *Proceedings of the 2015 Second European Network Intelligence Conference*, pp. 75–82, IEEE, Karlskrona, Sweden, September 2015.

[2] M. Alazab, R. Layton, R. Broadhurst, and B. Bouhours, "Malicious spam emails developments and authorship attribution," in *Proceedings of the 2013 Fourth Cybercrime and*

Trustworthy Computing Workshop, pp. 58–68, IEEE, Sydney, Australia, November 2013.

[3] B. Boenninghoff, S. Hessler, D. Kolossa, and R. M. Nickel, "Explainable authorship verification in social media via attention-based similarity learning," in *Proceedings of the 2019 IEEE International Conference on Big Data (Big Data)*, pp. 36–45, IEEE, Los Angeles, CA, USA, December 2019.

[4] A. Rocha, W. J. Scheirer, C. W. Forstall et al., "Authorship attribution for social media forensics," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 1, pp. 5–33, 2016.

[5] Y. Singh and S. Banerjee, "Fake (sybil) account detection using machine learning," 2019.

[6] F. Benevenuto, G. Magno, T. Rodrigues, and V. Almeida, "Detecting spammers on twitter," in *Proceedings of the Collaboration, Electronic Messaging, Anti-Abuse and Spam Conference (CEAS)*, vol. 6, p. 12, Perth, Australia, September 2010.

[7] P. Gao, B. Wang, N. Z. Gong, S. R. Kulkarni, K. Thomas, and P. Mittal, "Sybilfuse: combining local attributes with global structure to perform robust sybil detection," in *Proceedings of the 2018 IEEE Conference on Communications and Network Security (CNS)*, pp. 1–9, IEEE, Beijing, China, June 2018.

[8] N. Z. Gong, M. Frank, and P. Mittal, "Sybilbelief: a semi-supervised learning approach for structure-based sybil detection," *IEEE Transactions on Information Forensics and Security*, vol. 9, no. 6, pp. 976–987, 2014.

[9] M. Singh, D. Bansal, and S. Sofat, "Who is who on twitter-spammer, fake or compromised account? A tool to reveal true identity in real-time," *Cybernetics and Systems*, vol. 49, no. 1, pp. 1–25, 2018.

[10] C. VanDam, J. Tang, and P. N. Tan, "Understanding compromised accounts on twitter," in *Proceedings of the International Conference on Web Intelligence*, pp. 737–744, Thessaloniki, Greece, October 2017.

[11] K. Okereafor and O. Adelaiye, "Randomized cyber attack simulation model: a cybersecurity mitigation proposal for post covid-19 digital era," 2020.

[12] J. S. Li, J. V. Monaco, L. C. Chen, and C. C. Tappert, "Authorship authentication using short messages from social networking sites," in *Proceedings of the 2014 IEEE 11th International Conference on e-Business Engineering*, pp. 314–319, IEEE, Guangzhou, China, November 20.

[13] M. Egele, G. Stringhini, C. Kruegel, and G. Vigna, "Compa: detecting compromised accounts on social networks," in *Proceedings of the NDSS Symposium*, San Diego, CA, USA, February 2013.

[14] M. Nauta, "Detecting hacked twitter accounts by examining behavioural change using twtter metadata," in *Proceedings of the 25th Twente Student Conference on IT*, Twente, Netherlands, May 2016.

[15] P. Savyan and S. M. S. Bhanu, "Ubcadet: detection of compromised accounts in twitter based on user behavioural profiling," *Multimedia Tools and Applications*, vol. 79, pp. 19349–19385, 2020.

[16] D. Seyler, L. Li, and C. Zhai, "Identifying compromised accounts on social media using statistical text analysis," 2018.

[17] B. Viswanath, M. A. Bashir, M. Crovella et al., "Towards detecting anomalous user behavior in online social networks," in *Proceedings of the 23rd {USENIX} Security Symposium ({USENIX} Security 14)*, pp. 223–238, San Diego, CA, USA, August 2014.

[18] C. Suman, S. Saha, P. Bhattacharyya, and R. S. Chaudhari, "Emoji helps! a multi-modal siamese architecture for tweet user verification," *Cognitive Computation*, vol. 7, pp. 1–16, 2020.

[19] S. Barbon, R. A. Igawa, and B. Bogaz Zarpelão, "Authorship verification applied to detection of compromised accounts on online social networks," *Multimedia Tools and Applications*, vol. 76, no. 3, pp. 3213–3233, 2017.

[20] F. Lagerholm, "Using artificial intelligence to verify authorship of anonymous social media posts," 2017.

[21] M. L. Brocardo, I. Traore, I. Woungang, and M. S. Obaidat, "Authorship verification using deep belief network systems," *International Journal of Communication Systems*, vol. 30, no. 12, p. e3259, 2017.

[22] R. Kaur, S. Singh, and H. Kumar, "Authcom: authorship verification and compromised account detection in online social networks using ahp-topsis embedded profiling based technique," *Expert Systems with Applications*, vol. 113, pp. 397–414, 2018.

[23] T. R. Gadekallu, N. Khare, S. Bhattacharya et al., "Early detection of diabetic retinopathy using pca-firefly based deep learning model," *Electronics*, vol. 9, no. 2, p. 274, 2020.

[24] Z. C. Steinert-Threlkeld, *Twitter as Data*, Cambridge University Press, Cambridge, UK, 2018.

[25] G. Barlas and E. Stamatatos, "Cross-domain authorship attribution using pre-trained language models," in *Proceedings of the IFIP International Conference on Artificial Intelligence Applications and Innovations*, pp. 255–266, Springer, Berlin, Germany, 2020.

[26] A. Usha and S. M. Thampi, "Authorship analysis of social media contents using tone and personality features," in *Proceedings of the International Conference on Security, Privacy and Anonymity in Computation, Communication and Storage*, pp. 212–228, Springer, Guangzhou, China, December 2017.

[27] M. Kestemont, K. Luyckx, W. Daelemans, and T. Crombez, "Cross-genre authorship verification using unmasking," *English Studies*, vol. 93, no. 3, pp. 340–356, 2012.

[28] O. Halvani, C. Winter, and A. Pflug, "Authorship verification for different languages, genres and topics," *Digital Investigation*, vol. 16, pp. S33–S43, 2016.

[29] K. A. Maria, "Authorship Attribution Forensics: feature selection methods in authorship identification using a small e-mail dataset," Master's thesis, Technoglossia University, Athens, Greece, 2016.

[30] G. Sidorov, F. Velasquez, E. Stamatatos, A. Gelbukh, and L. Chanona-Hernández, "Syntactic n-grams as machine learning features for natural language processing," *Expert Systems with Applications*, vol. 41, no. 3, pp. 853–860, 2014.

[31] P. Juola, "Authorship attribution," *Foundations and Trends® in Information Retrieval*, vol. 1, no. 3, pp. 233–334, 2008.

[32] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," *Advances in Neural Information Processing Systems*, vol. 26, pp. 3111–3119, 2013.

[33] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," 2013, https://arxiv.org/abs/1301.3781.

[34] Y. Zhang, R. Jin, and Z. H. Zhou, "Understanding bag-of-words model: a statistical framework," *International Journal of Machine Learning and Cybernetics*, vol. 1, no. 1–4, pp. 43–52, 2010.

[35] G. Forman, "Bns feature scaling: an improved representation over tf-idf for svm text classification," in *Proceedings of the 17th ACM Conference on Information and Knowledge Management*, pp. 263–270, Singapore, November 2008.

[36] C. Lemke, M. Budka, and B. Gabrys, "Metalearning: a survey of trends and technologies," *Artificial Intelligence Review*, vol. 44, no. 1, pp. 117–130, 2015.

[37] T. Hospedales, A. Antoniou, P. Micaelli, and A. Storkey, "Meta-learning in neural networks: a survey," 2020, https://arxiv.org/abs/2004.05439.

[38] A. Castro and B. Lindauer, "Author identification on twitter," 2012.

[39] S. Kotsiantis, D. Kanellopoulos, and P. Pintelas, "Data preprocessing for supervised leaning," *International Journal of Computer Science*, vol. 1, no. 2, pp. 111–117, 2006.

[40] A. Jahan, K. L. Edwards, and M. Bahraminasab, *Multi-Criteria Decision Analysis for Supporting the Selection of Engineering Materials in Product Design*, Butterworth-Heinemann, Oxford, UK, 2016.

[41] G. G. Mesran, G. Ginting, G. Suginam, and R. Rahim, "Implementation of elimination and choice expressing reality (electre) method in selecting the best lecturer (case study stmik budi darma)," *International Journal of Engineering Research and Technology*, vol. 6, no. 2, pp. 141–144, 2017.

[42] A. Yanie, A. Hasibuan, I. Ishak et al., "Web based application for decision support system with electre method," *Journal of Physics: Conference Series*, vol. 1028, p. 12054, 2018.

[43] L. Botti and N. Peypoch, "Multi-criteria electre method and destination competitiveness," *Tourism Management Perspectives*, vol. 6, pp. 108–113, 2013.

[44] M. G. Yücel and A. Görener, "Decision making for company acquisition by electre method," *International Journal of Supply Chain Management*, vol. 5, no. 1, pp. 75–83, 2016.

[45] T. L. Saaty, "Decision making with the analytic hierarchy process," *International Journal of Services Sciences*, vol. 1, no. 1, pp. 83–98, 2008.

[46] M. A. Hall, "Correlation-based feature selection for machine learning," 1999.

[47] M. Alazab, S. Huda, J. Abawajy et al., "A hybrid wrapper-filter approach for malware detection," *Journal of Networks*, vol. 9, no. 11, pp. 2878–2891, 2014.

[48] E. Roszkowska, "Rank ordering criteria weighting methods–a comparative overview," 2013.