IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received 22 June 2023, accepted 1 July 2023, date of publication 10 July 2023, date of current version 19 July 2023.

*Digital Object Identifier 10.1109/ACCESS.2023.3293641*

RESEARCH ARTICLE

# A Multilingual Spam Reviews Detection Based on Pre-Trained Word Embedding and Weighted Swarm Support Vector Machines

**ALA' M. AL-ZOUBI[1], ANTONIO M. MORA[2], AND HOSSAM FARIS[3,4]**
[1]School of Computer Sciences and Telecommunications (ETSIIT), University of Granada, 18071 Granada, Spain
[2]Department of Signal Theory, Telematics and Communications, ETSIIT and CITIC-UGR, University of Granada, 18071 Granada, Spain
[3]Department of Computer Engineering, Automatics, and Robotics, ETSIIT and CITIC-UGR, University of Granada, 18071 Granada, Spain
[4]King Abdullah II School for Information Technology, The University of Jordan, Amman 11942, Jordan

Corresponding author: Ala' M. Al-Zoubi (alzoubi@correo.ugr.es)

This work has been partially funded by the UGR Visiting Scholars Programme. It has also been funded by projects TED2021-129938B-I0, PID2020-113462RB-I00, PDC2022-133900-I00, and PID2020-115570GB-C22, granted by Ministerio Español de Ciencia e Innovación MCIN/AEI/10.13039/501100011033; as well as project PDC2022-133900-I00, funded by MCIN/AEI and Next GenerationEU/PRTR.

**ABSTRACT** Online reviews are important information that customers seek when deciding to buy products or services. Also, organizations benefit from these reviews as essential feedback for their products or services. Such information required reliability, especially during the Covid-19 pandemic which showed a massive increase in online reviews due to quarantine and sitting at home. Not only the number of reviews was boosted but also the context and preferences during the pandemic. Therefore, spam reviewers reflect on these changes and improve their deception technique. Spam reviews usually consist of misleading, fake, or fraudulent reviews that tend to deceive customers for the purpose of making money or causing harm to other competitors. Hence, this work presents a Weighted Support Vector Machine (WSVM) and Harris Hawks Optimization (HHO) for spam review detection. The HHO works as an algorithm for optimizing hyperparameters and feature weighting. Three different language corpora have been used as datasets, namely English, Spanish, and Arabic in order to solve the multilingual problem in spam reviews. Moreover, pre-trained word embedding (BERT) has been applied alongside three-word representation methods (NGram-3, TFIDF, and One-hot encoding). Four experiments have been conducted, each focused on solving and demonstrating different aspects. In all experiments, the proposed approach showed excellent results compared with other state-of-the-art algorithms. In other words, the WSVM-HHO achieved an accuracy of 88.163%, 71.913%, 89.565%, and 84.270%, for English, Spanish, Arabic, and Multilingual datasets, respectively. Further, a deep analysis has been conducted to investigate the context of reviews before and after the COVID-19 situation. In addition, it has been generated to create a new dataset with statistical features and merge its previous textual features for improving detection performance.

**INDEX TERMS** Security, detection, spam reviews, pre-trained, word embedding, weighted SVM, Covid-19, multilingual.

## I. INTRODUCTION

Due to the evolution of technology, the Internet, and mobile devices, information has become accessible to virtually anyone. Information is usually provided in various forms, including images, sounds, videos, and text. On the internet, this information can be used to verify, inform, and explain various things. Such information can be critical and important to numerous individuals and organizations across different domains [1].

For example, in the medical and especially in the pharmaceutical field, it involves gathering, processing, and disseminating information about medications, as well as an explanation of how to safely and correctly use them. As for

The associate editor coordinating the review of this manuscript and approving it for publication was Jad Nasreddine.

economics, goods and services are produced primarily as a result of information-intensive activities that augment scientific and technical innovation. Hence, a society's integration and development are determined by the flow of information within it. For instance, [2] stated that information has three different styles: information as a process, information as knowledge, and information as a thing. Therefore, information may consider more than just informed data that exist on the web, it also can be categorized as a communication approach between people.

Information can also be found in reviews, which are known as feedback written about the customer experience of products or services. According to De et al. [3], reviews, or online reviews, are a form of electronic Word-Of-Mouth (eWOM), which involves communicating information about product usage, services, or the behavior of sellers to consumers using internet-based technology. On the other hand, Lo and Yao [4] define reviews as user-generated content (UGC) used by consumers to make decisions regarding various activities such as purchasing, traveling, renting, and signing up for services. Reading such reviews offers several benefits, including minimizing risk, simplifying decisions, generating new ideas, and providing consumers with a range of options for a product. Hu et al. [5] interpret online reviews as a method for previous users to critique specific products.

These opinions can be helpful for other users in deciding whether to purchase these products or not. Additionally, online reviews are regarded as a valuable source of information for retailers, manufacturers, and decision-makers. They can use these reviews to gauge the demand for certain products, adjust production quantities accordingly, and identify any issues or problems associated with the products. Consumers' feedback on the internet has become the most effective means of understanding consumer responses to specific services or products worldwide [6].

Various websites allow users to post reviews about products, services, places, flights, etc. These websites have high internet traffic in comparison to other e-commerce websites without reviews feature, for example, Google My Business has 158.03 million average monthly accesses in the US, while Amazon, Yelp, and Trip Advisor have 85.44 million, 40.47 million, and 28.27 million, respectively.

During 2019 and 2020, the COVID-19 pandemic forced more people to stay at home, leading to an increased reliance on remote technology by governments, organizations, and individuals. This resulted in a significant surge in data traffic and internet usage. For instance, in the first months of 2020, e-commerce sales in the US increased by 32% compared to the same period in 2019, while retail e-commerce sales grew by 13.7% in 2021 [7]. According to [8], international internet traffic grew by an annual rate of 29% between 2017 and 2021, with the peak occurring in 2019-2020 due to the COVID-19 pandemic. Consequently, the number of online reviews reached record highs. As reported by PowerReviews [9], review interactions increased by 50% compared to the pre-pandemic period. Specifically, review engagement rose by 89% during the initial months of the pandemic. Therefore, it is not surprising that 41% of beauty shoppers, for example, now rely much more on ratings and reviews than they did prior to the global health crisis [9].

Furthermore, the crisis also prompted people to change their spending habits and prioritize different products. For instance, over 54% of beauty shoppers now prefer to use less makeup and prioritize skincare instead. Additionally, there is an increased focus on purchasing vitamins, antibiotics, and medical supplies such as face masks, gloves, hand sanitizer, and medical oxygen. Currently, the largest community for online reviews and feedback exists within online social networks, where users have the freedom to express their opinions at any time and in any manner they choose. Online social networks are widely regarded as the most impactful and influential platforms on the internet, providing user-friendly and convenient web-based interfaces for publishing user-generated content. These platforms offer governmental and commercial entities various avenues to study and analyze customer behavior, promote their products, and extract valuable insights from user opinions [10].

Therefore, it is crucial for all parties involved to determine the authenticity of reviews. For example, a user may maliciously write feedback to trick consumers into buying a certain product where in fact it is not good enough and has flaws. Such feedback and reviews are known as spam reviews, where spammers manipulate the reviews to either promote or devalue services and products [11].

The impact of false reviews can be significant for individuals and businesses, as they can lead to financial losses and even job insecurity if people rely on them. Given that online reviews are often unreliable and untruthful, it becomes crucial to identify and mitigate the presence of such undesired reviews. By doing so, it becomes possible to create a safer environment for others and maintain a more positive business image.

To address this issue, numerous researchers in the field have proposed various detection approaches. One example is duplicate detection, which involves identifying frequently repeated reviews that are posted by either the same or different individuals for the same or different products [12]. Two perspectives can be employed to analyze similarities between reviews: conceptual similarity and text duplication. Generating multiple fake reviews with diverse content is time-consuming and costly. Consequently, spammers often opt to copy the text from existing fake reviews rather than creating multiple distinct ones. Therefore, identifying similar reviews plays a crucial role in detecting spam reviews.

Additionally, there have been several attempts in the literature to detect spam reviews based solely on their content. The content-based detection method focuses more on the context of the reviews and ignores the reviews' metadata. Thus, it aims to distinguish between spam and real reviews by analyzing the review content. There are three types of content-based detection approaches: genre identification, detection of psycho-linguistic deception, and text categorization. This method exhibits excellent performance compared to other techniques. Therefore, in this work, we will utilize the content-based detection method to detect spam reviews.

Applying this method requires dealing with the text more carefully compared to other approaches. Therefore, several Natural Language Processing (NLP) procedures need to be performed to fully exploit the content's text. NLP is considered a branch of artificial intelligence that enables computers to manipulate, understand, and interpret human language [13]. The term NLP refers to a combination of computational linguistics with deep learning, statistical, and machine learning models. These technologies allow computers to process text data in human language, identify the intended meaning and sentiment, and comprehend its full implications. This includes Feature Extraction (FE), a process that converts raw data into multidimensional data that can be processed by the aforementioned models [14]. Incorporating this technique would significantly improve the performance of machine learning models by extracting useful features.

One of these techniques is Word Embedding (WE) [15], which is also known as distributed word representation or word representation. It works as a language model whose purpose is to convert words or textual phrases to continuous spaces with low dimensions, as described in [15]. According to [16], WE is based on the distribution hypothesis ($w$, $w^{\sim}$), where w denotes the words and $w^{\sim}$ are semantically similar words. Therefore, we aim to collect both syntactic and semantic information, as semantics have to do with meaning, while syntactic has to do with structure.

One of these techniques is Word Embedding (WE) [15], also known as distributed word representation or word representation. It functions as a language model that aims to convert words or textual phrases into continuous spaces with low dimensions, as described in [15]. According to [16], WE is based on the distribution hypothesis ($w$, $w^{\sim}$), where w denotes the words and $w^{\sim}$ are semantically similar words. Therefore, our objective is to capture both syntactic and semantic information, as semantics relate to meaning, while syntax relates to structure.

However, sometimes the use of WE requires a large amount of data to perform well. Therefore, to address this challenge when dealing with medium to small-sized data, a pre-trained WE model can be employed. This approach is applied when the data is insufficient for training and to reduce training time [17]. Furthermore, pre-trained WE models have shown improvements in various applications across different domains and data sizes. According to [18] and [17], the two most commonly used pre-trained embeddings are Word2Vec and FastText embeddings. Word2Vec employs fast word representation techniques such as Continuous Bag-Of-Words or Skip-grams. In contrast, FastText utilizes both a sample set and character-level techniques.

The second part of using content-based detection depends on the Machine Learning (ML) models. Several ML approaches have been used to perform this task in the literature such as [19], [20], [21], and [22]. In [19], the authors proposed an algorithm to detect fake reviews using the unigram and bigram feature extraction methods. Moreover, review filtering is performed using supervised learning techniques. Then, multiple ML algorithms are combined together to achieve better prediction performance. Whereas, in [20] they used Twitter's online social network to collect the reviews. Additionally, they applied Support Vector Machine (SVM) in order to determine the spam reviews.

The previously mentioned approaches achieve somehow good results in detecting spam reviews. However, they lack to obtain similar performance on other data (problem). Such an issue can be solved by reducing the dimensionality of the data (feature selection) and doing a parameter optimization based on the confronted problem. Through using their dynamic ability to be customized according to the challenges faced, metaheuristic algorithms can achieve this in various ways. These algorithms can perform many tasks simultaneously. Hence, more space to improve and solve the encountered problems, and spam reviews detection is not an exception. Few researchers took this path of research in the literature to tackle spam reviews. For instance, [23] presented clustering spiral cuckoo search methods for spam detection. Reference [24] proposed Cuckoo and Harmony features combination in an effective hybrid feature selection technique, and Naive Bayes is used as a classification method for classifying reviews as spam or ham. While, the work in [25], presented an investigation into the detection of fake reviews in the big data environment, using parallel biogeography optimization. Further, feature selection is used by the binary flower pollination algorithm for spam review detection [26].

Therefore, all the aforementioned works lack several aspects that this study aims to address. These aspects include the utilization of advanced classifiers to optimize their parameters and improve performance, exploration of different word representation methods, investigation of spam review detection following the Covid-19 pandemic, and handling a multilingual context. Thus, in this study, we propose the utilization of a weighted SVM combined with a Harris Hawks Optimization (HHO) approach and pre-trained WE for the detection of spam reviews in a multilingual environment during the pandemic. Specifically, three different languages, namely English, Spanish, and Arabic, have been extracted for each product review. Additionally, various word representation methods have been employed to analyze and determine the most effective technique for these datasets.

Further, this study has explored several findings related to review characteristics. For example, the nature of reviews has evolved in the context of the pandemic, with a stronger emphasis on health and technological products rather than fashion or beauty-related things. This shift in review structure and context has also had an effect on spam reviews, which have developed and gotten more sophisticated. These changes have had an impact on both the features and language used in spam reviews. Customers' preferences are shifting toward medical and entertainment offerings in various languages. As a result, spam reviews have followed suit in an attempt to fool consumers. Notably, our research highlighted the most important aspects of these themes, including immediate treatment, COVID-19 protection, health advantages, safety, entertainment value, weight loss, and more. It is critical to adapt to changes in the review context and structure. The abbreviations used in this study are summarized in Table 1 for easy reference.

The experimental phase of this work consists of four stages:

- Experiment I: A comparison of well-known classification algorithms on our generated datasets. The purpose of this study is to investigate the initial results of the new data.
- Experiment II: To determine the best metaheuristic algorithm for this problem, we have hybridized different metaheuristic algorithms with SVM.
- Experiment III: We compare the proposed WSVM-HHO approach with HHO-SVM with Feature Selection (HHO-FsSVM).
- Experiment IV: The proposed WSVM-HHO algorithm is compared with other well-known classifiers in the literature combined with different metaheuristic algorithms.

In this regard, the contribution of the proposed approach can be summarized as follows:

- A weighted Support Vector Machine (SVM) combined with a Harris Hawks Optimization (HHO) for hyperparameters optimization and feature weighting.
- Spam reviews datasets have been generated taking into account two main aspects: the multilingual environment and the Covid-19 pandemic. Thus, four datasets are created and labeled, including, English, Spanish, Arabic, and all combined (multilingual).
- A pre-trained WE approach has been used together with other word representation methods of each data.
- A deep analysis of the reviews before and after the Covid-19 pandemic. To put it another way, analyze how the reviews' structure, style, and content have changed.

The rest of the paper is organized as follows: Section II discusses previous work on spam reviews detection methods. The methodology and description of the proposed approach are presented in Section III. Experiments and results are provided in Section IV. Conclusions and future work are addressed in Section V.

## II. RELATED WORK

Detection of spam has been an important method to mitigate security issues [27]. In academic and industrial institutions, detection methods have been developed for identifying and controlling the problem of spam reviews. There are many detection techniques that have emerged in the last decade. The behavior-detection-based method is an example of one of these techniques. In order to detect spam reviews, this type of detection relies on the behavior of the users to distinguish such reviews.

**TABLE 1.** List of abbreviations.

| Abbreviation | Full Form |
|---|---|
| NLP | Natural Language Processing |
| WE | Word Embedding |
| ML | Machine Learning |
| FE | Feature Extraction |
| SVM | Support Vector Machine |
| HHO | Harris Hawks Optimization |
| Covid-19 | Coronavirus Disease 2019 |
| UGC | User-Generated Content |
| GAN | Generative Adversarial Network |
| CRFD | Cumulative Relative Frequency Distribution |
| CNN | Convolutional Neural Network |
| PV-DBOW | Paragraph Vector Distributed Bag-of-Words |
| LSTM | Long Short-Term Memory |
| ACB | Self Attention-based CNN BiLSTM |
| LIWC | Linguistic Inquiry Word Count |
| PCA | Principal Component Analysis |
| CS | Cuckoo Search |
| NB | Naive Bayes |
| iBPSO | Hybrid Optimized Binary Particle Swarm Optimization |
| WEs | Word Embeddings |

An example of such detection type is the study in [28] that explored spam reviews by employing the Generative Adversarial Network (GAN) to create synthetic features based on the user's behavior. By selecting six fundamental features based on the normal users' behavior, the GAN trains on the data for creating synthetic features ( i.e. text features, rating, and attributes). Furthermore, novel generators and discriminators are implemented for optimal training. This way, the GAN can create synthetic behavior features for new users. Using the Yelp dataset as the basis for their experiments, the results indicate that the framework outperforms other approaches.

### A. SUPERVISED LEARNING

The work [29] is another example of behavior detection. In the paper, a model was presented to distinguish genuine reviews from spam. Random forest and autoencoder are exploited in this model using end-to-end training. Further, in order to identify the global parameter of a learning model, the decision tree model is applied. When compared to other methods on the Amazon review dataset, the experiments in this paper show that the proposed model produces better results.

The literature has also presented another method of detecting spam reviews, namely supervised learning-based detection [30]. As part of the machine learning methods, this type relies on labeled examples to learn. A variety of studies have used supervised learning for detecting spam reviews, including that discussed in [31]. They used supervised classification models to detect fake online reviews that could have a serious impact on users' decisions. In this study, various features were extracted to be re-engineered using the Cumulative Relative Frequency Distribution (CRFD) method to enhance the detection process.

Meanwhile, the authors in [32] have also adapted the supervised learning technique to address online review falsification. A neural approach was used to classify the reviews using syntactic and lexical patterns in order to solve such a problem. A comparison is made between several supervised classification models and their approaches. In order to examine all approaches, they used Google's WE architecture (BERT) [33]. Also, a method for detecting fake reviews in the Tourism environment was presented in [34] (HOTFRED). These types of reviews hamper hoteliers and guests when planning or selecting the best hotels for their trip. HOTFRED system uses different analytical procedures to detect fake hotel reviews. Based on the system's efficient performance, hoteliers can use it as an automated tool to guarantee they choose the proper hotel and to prevent spam reviews to trick them.

### B. DEEP LEARNING

Many researchers have proposed different approaches to solve spam review detection using deep learning thanks to its capacity to learn from big data. As an example, the work presented in [35], where a Convolutional Neural Network (CNN) is employed. In order to improve the detection of deceptive characteristics, CNN is used to identify the semantic details of reviews. In terms of detecting spam reviews, the proposed CNN is more effective than other neural network architectures. Moreover, Fahfouh et al. [36] also discuss how deep learning improves the extraction of semantics from the context of reviews. By using their new method (PV-DBOW), they can determine the global meaning of reviews. Additionally, the representation of reviews can be transferred to a neural network model for spam reviews detection. Experimentally, PV-DBOW shows better performance than existing state-of-the-art methods.

Meanwhile, Dang et al. [37] presented a novel method for detecting spam reviews using multi-dimensional features. In order to classify the user-product relationship, the standard component was used to acquire low-dimensional features. The spatial structure and textual context features are identified using the long short-term memory (LSTM) technique and a capsule network. Furthermore, the model combines both users' behavioral and textual features into a single module for classification detection. Results of the approach show that it is more effective than existing methods at detecting spam reviews. Additionally, the study [38] raises the issue of labeled datasets being unavailable in spam reviews. In order to distinguish spam reviews, the authors offered LSTM networks based on unsupervised learning. By training it with real reviews without labels, the model discovers patterns in the reviews. The experiment results indicate that the framework is capable of distinguishing real reviews from spam.

In its study of spam reviews, [39], the authors also applied LSTM to investigate the semantic features of the reviews. Additionally, the authors used CNN for detecting discrete features as well as Deep Belief Network to specify the credibility of product reviews. Given that, in order to build a DBN model, standard features are combined with semantic features.

Similarly, in [40], the authors discuss the shortcomings of traditional machine-learning approaches for detecting spam reviews. In the following, a deep learning framework is used to overcome these drawbacks. In other words, Self Attention-based CNN BiLSTM (ACB) is used for extracting and identifying the representations of reviews. A weighted combination of words is calculated, along with the identification of spamming indications in the sentences and documents. After the sentence representation is analyzed, a CNN is trained to discover the higher-level n-gram features. By using Bi-directional LSTM, the vectors of sentences are merged based on contextual information to detect spam reviews. In terms of accuracy, the ACB approach attained the best results compared with other variants techniques.

### C. MULTILINGUAL

The contextual structure of the reviews is usually taken into account in linguistic and multilingual studies. Lingual detection approaches indicate that it is also a problem in other languages and regions. Moreover, this type of detection relies heavily on linguistic characteristics, so the majority of features are text-based.

As reported in [41], the majority of spam review detection works have focused more on languages such as English, Chinese, and Arabic. Thus, the authors of that study intend to design a spam review detection system that is based on Roman Urdu. Two types of features are incorporated in various classification models, including linguistic and behavioral features. Evaluation of performance was conducted from three different perspectives. The first perspective utilized only linguistic features in the classification models. While in the second, the level of accuracy of each model is dependent on how the behavioral features are intertwined with the distributional and non-distributional aspects. Thirdly, behavioral and linguistic characteristics are combined and used as evaluation criteria. According to the experimental evaluations, the best method was obtained by the third perspective that combined both features, behavioral and linguistic.

Additionally, a new methodology based on behavioral and linguistic features was also applied in a recent study [42]. In a similar fashion to the prior study, two different procedures were employed for spam, reviews detection. In the first case, they used the Behavioral Method (SRD-BM), and in the second case, they used the Linguistic Method (SRD-LM). In the SRD-BM, 13 features were considered for the detection phase, whereas the RD-LM focused on textual features. SRD-BM and SRD-LM showed higher overall performance than other state-of-the-art approaches at 93.1% and 88.5%, respectively.

In their paper [43], the authors studied the detection of spam reviews using Word Count (LIWC) and linguistic content. The Principal Component Analysis (PCA) technique is employed to reduce the high number of dimensions in the data. To determine the effectiveness of the evaluation process, a total of five variances were used, both with and without PCA models. With regards to accuracy, the ensemble Bagged classifier exceeds all other supervised methods. Additionally, [44] presents an unsupervised method for detecting spam reviews on Chinese texts, images, videos, and other media. Following the evaluation, they found a number of results: 1. Video and text spam are less common than image spam; 2. Preferring to steal from reviews rather than a marketing campaign; 3. In order to influence customers, spammers are more likely to use pseudo-rare incidents than any other type of trick; 4. spammers frequently use the same techniques to manipulate text, images, and video.

According to [45], customers' understanding of spam reviews is still being investigated insufficiently. In order to describe the reviewer's intentions, they designed a theoretical model based on a linguistic approach. By applying the fractional logit model, 120 reviews were analyzed. From the results, the speaker has shown his intention based on the method he used. Additionally, customers are more likely to perceive reviews with fewer arguments, flattering, and contextual embeddings as spam reviews. Two studies for the detection of spam were presented in [46]. As part of the first study, the researchers used features of Linguistic Inquiry Word Count (LIWC) to analyze Yelp data. On the other hand, 660 participants were considered to label reviews as confident or doubtful in the second study. The results of both studies indicated that positive reviews were more trustworthy than negative ones.

A machine-learning model is developed in study [47] in order to identify trustworthiness in an opinion. During the course of the study, the authors gathered a large dataset of spam reviews consisting of 869 false and 866 truthful reviews. In such a case, this was the first attempt at reviewing data in the Korean language. According to the results, the model achieves an accuracy of approximately 81%.

### D. METAHEURISTIC ALGORITHMS

Additionally, more advanced methods have been investigated during the development and maturation of spam reviews detection analysis. One of these methods was using metaheuristic algorithms to enhance the machine learning classifiers for detecting spam reviews. For example, For the detection of spam reviews, the authors have developed a spiral cuckoo search-based clustering approach [23]. In this method, a spiral is used to resolve the cuckoo search method's convergence issue, while also taking advantage of the strengths of its search mechanism. In addition to four spam datasets, one Twitter spammer dataset was also used to prove the effectiveness of the proposed method. To validate the capability of the proposed clustering method, six metaheuristic clustering approaches are compared with it. Both experimental results and statistical analysis prove that the proposed method is faster and more accurate than existing methods. An algorithm based on the military dogs' squad is presented in paper [48]. Through this algorithm, the military dogs' searching abilities are mimicked. A performance test is conducted with 17 benchmark functions followed by a comparison with five other meta-heuristics. In this study, the fitness value is validated by calculating the mean and standard deviation. Also, a fake review detection problem was solved using the proposed algorithm. In the experiments, the proposed algorithm outperformed the other algorithms and achieved the highest results from the benchmark functions.

An optimization-based parallel biogeography-based method is presented in [25] to unravel the fake review detection in the big data environment. This study examines the comparison between K-means and 4 state-of-the-art methods using two standard fake review datasets. In both datasets, it was found that the proposed method outperformed all the other methods. Moreover, the existing method is evaluated for its parallel performance potency by analyzing speedup results.

With the help of flower pollination, gray wolf, and moth flame, the authors propose a novel framework for the detection of opinion spam using a meta-heuristic and k-means clustering approach [49]. The dataset for analysis was selected from Amazon's automotive products. In comparison to moth flame and flower pollination algorithms, the gray wolf algorithm performs better. The algorithm obtained the best results in terms of run time, convergence speed, variance, mean and standard deviation. In another work [24], an effective hybrid feature selection technique using Cuckoo Search (CS) with Harmony search is proposed and Naive Bayes is used for classifying the review into spam and ham. A global optimization technique called Adaptive Binary Flower Pollination Algorithm (BFPA) is presented in paper [26] to extract features. As an objective function, they use the accuracy of the Naive Bayes classifier (NB). Comparing the proposed method to other competitive methods, the experimental results indicate that this approach selected only the informative features and gave higher classification accuracy as compared to the others.

Another work for spam reviews detection is presented by [50], wherein the feature selection phase, a hybrid optimized Binary Particle Swarm Optimization (iBPSO) method is combined with CS. In order to classify reviews as spam or ham, Naive Bayes and k Nearest Neighbors are used. According to the experimental results, the proposed algorithm consistently outperforms other Binary Particle Swarm Optimization (BPSO) algorithms.

### E. WORD EMBEDDING

Additionally, other authors used advanced natural language processing techniques to generate feature representation in order to enhance the detection of spam reviews. For example, the authors of [51] proposed two neural network models that successfully detect fake reviews by taking into account the word context and the consumer's emotional state as well as traditional bag-of-words methods. Particularly, three sets of features are used to construct document-level representations: (1) WEs; (2) different lexicon-based sentiment indicators; and (3) n-grams. Fake reviews are classified into four domains using this high-dimensional feature representation. They compare the classification performance of the suggested detection methods with several state-of-the-art approaches for fake review detection in order to demonstrate their effectiveness. Regardless of sentiment polarity or product category, the proposed approach performs well on all datasets. Paper [52] proposes a novel content-based approach for review spam detection that takes into account both the bag-of-words and the word context. To build a vector model, they exploit n-grams and a method called skip-gram WE. Thus, high-dimensional features are formed. In the second step, a deep feed-forward neural network is used to represent and classify the spam reviews accurately. Two hotel review datasets are used to test their approach, including positive and negative reviews. Using the proposed detection approach for spam detection, they demonstrate that it outperforms existing algorithms in both accuracy and area under the ROC curve.

As a result of deep learning methods such as Word2Vec, one can acquire more accurate vector representations of words and enhance the training of standard machine learning algorithms [53]. The disadvantage of deep learning, in this case, is that only 1600 and 2000 reviews were used from Ott and Yelp datasets, respectively. This small number of data might put the detection accuracy at risk of an overfitting problem. As a result of hardware limitations, word embeddings had to be limited in dimension. Deep learning has been applied only to Word2Vec WEs, which are not pre-trained. As part of that process, they have used both labeled and unlabeled data, as well as deep learning methods for spam review detection including, Convolutional Neural Network (CNN), Long Short-Term Memory (LSTM), Multi-Layer Perceptron (MLP) and Recurrent Neural Network (RNN).

Table 2 summarizes the various research works based on their approaches employed and key findings.

Therefore, this work differs from previous ones using metaheuristic algorithms for parameter optimization and feature weighting for spam review detection. In addition, pre-trained WEs are used for multilingual contexts. Study the context of the reviews and structure before and after the covid-19 situation.

## III. METHODOLOGY

In this section, we will describe the four stages that have been undertaken to design the methodology for our work. These stages include Preliminaries, Data Description, Data Preparation and Labeling, Proposed Approach, and Evaluation Measures.

### A. PRELIMINARIES

#### 1) SVM

Support Vector Machine (SVM) was developed by Vladimir Vapnik as a non-linear binary classifier [54], [55], [56]. It is used to classify non-linear data and transformed it into a linear space. The algorithm can be used when the target function is not linearly separable. SVM will try to find a hyperplane that separates as many points of one class from the other as possible, which means that SVM tries to maximize the margin between different classes. It also means that for any specific input, if there exists such a hyperplane, it must have zero inner product with this input vector. A support vector machine is an example of a supervised learning algorithm and is used to fit a linear or non-linear model to data. In short, SVM tries to find the best hyperplane that can be used as a separating line between all the samples classified into two different classes. SVM also cable to perform on regression problems.

In more detail, the SVM generates linear separating hyperplanes in a vector space with high-dimensional characteristics, where each data point is seen as a $x_i, y_i)$ pair, where the feature vector $(x_i)$ is part of $\left(x_{i1}, x_{i2}, \ldots, x_{ip}\right)$, the number of features denote with $p$, $i = 1, \ldots, n$, where $n$ denotes the number of training instances and the class label presented by $y_i$ [57].

As long as the feature space is well separated, the data points belonging to each labeled class will be separated into distinct areas within it by hyperplanes. The finest hyperplane is determined by its ability to maximize the margin (distance) of the nearest training data point. The observations with more than one margin away from the hyperplane are known as support vectors (SV). Such vectors can be found in the feature space and the hyperplane location relies on these vectors. Therefore, if the location of the observations SV changed. The hyperplane, based on these observations, will also change its location [57].

Hyperplanes that separate linearly will provide optimal classification. Nevertheless, most of the time, these data points are not clearly divided into different classes. Thus, it is highly likely that the linear classification will lead to significant misclassification. In order to solve this frequent problem, mapping the initial feature space to more higher-dimensional space ($\varphi$). This way it becomes easier to distinguish between data points that belong to different classes (turn into linearly separable). Using kernel functions, we enlarge the original space in a non-linear way. There are several kernel functions can be used, namely:

- Linear kernels :

$$k\left(x_i, x_i'\right) = \sum_{j=1}^{p} x_{ij} x_{ij}' \tag{1}$$

- Polynomial kernels :

$$k\left(x_i, x_i'\right) = (1 + \sum_{j=1}^{p} x_{ij} x_{ij}')^d \tag{2}$$

**TABLE 2.** Summary of mentioned works.

| Paper | Approach | Key Findings |
|---|---|---|
| [27] | Behavior-detection-based | Detection methods developed for spam reviews in academic and industrial institutions |
| [28] | Generative Adversarial Network (GAN) | GAN generates synthetic behavior features for detecting spam reviews |
| [29] | Random Forest and Autoencoder | Proposed model produces better results than other methods on Amazon review dataset |
| [30] | Supervised Learning | Supervised learning models effectively detect fake online reviews |
| [31] | Supervised Classification Models | Various features extracted and re-engineered using CRFD method for enhanced detection |
| [32] | Supervised Learning | Neural approach using syntactic and lexical patterns for classification |
| [34] | HOTFRED system | Analytical procedures used to detect fake hotel reviews |
| [35] | Convolutional Neural Network (CNN) | CNN effectively detects spam reviews by identifying semantic details |
| [36] | PV-DBOW | PV-DBOW outperforms existing methods for extracting semantics from reviews |
| [37] | LSTM and Capsule Network | Multi-dimensional features combined with LSTM and Capsule Network for detection |
| [38] | LSTM | Unsupervised learning-based LSTM model effectively distinguishes real reviews from spam |
| [39] | LSTM, CNN, Deep Belief Network | LSTM and CNN used for semantic and discrete feature detection, DBN for credibility |
| [40] | Self Attention-based | Self Attention-based approach achieves high performance in detecting spam reviews |

- RBF kernels :

$$k\left(x_i, x_i'\right) = \exp(-\gamma \sum_{j=1}^{p} (x_{ij} - x_{ij}')^2) \tag{3}$$

where $k\,(.)$ denotes the kernel function, while $x_i$ and $x_i'$ are the observation of the two vectors for the inner product. As a result, the inner product for the transformed feature space $(\varphi)$ can be illustrated as $\varphi\,(x_i).\varphi\left(x_i'\right)$.

Accordingly, SVM performance depends entirely on its parameters and the selection of the kernel function. In this work, we utilized the RBF kernel in order to deal with non-linear decision boundaries and variables. The RBF kernel is calculated by Eq. 3, where the gamma coefficient denotes $\gamma$. While the other parameter is the cost $(C)$, the penalty parameter, which is used to improve the classification accuracy of the new data points. As a consequence, the inappropriate setting of $\gamma$ and $C$ could lead to inadequate generalization causing overfitting or underfitting of the data.

### 2) HARRIS HAWKS OPTIMIZATION (HHO)

A population-based optimization algorithm named Harris Hawks Optimizer (HHO) was developed by Heidari et al. [58]. Harris' hawks use an intelligent strategy known as the surprise pounce to chase their prey in HHO. Using this method, hawks surprise their prey by pouncing from different directions as portrayed in Figure 1. The HHO, like any optimization algorithm, has two primary phases, exploration and exploitation processes, alongside a state called the transition between exploitative behaviors.

A hawk observes and waits in the desert to detect its prey as one of the candidate solutions, and the selected prey is the best solution for each step. As part of their exploration phase, Harris' hawks choose random locations and wait to see if prey might be present. During the haunting, two strategies are used: the first takes into account the placement of other hawks who are participating in the haunting, and the second takes into account the existence of encountered tall trees within the range of the haunt.

Both strategies are explained by equation 4, which considers an equal chance $q$ for each positioning (perching) strategy. Therefore, $q$ must be greater than or equal to 0.5 in order to select the first strategy, otherwise, the second strategy is selected. The vector of hawks' positions are represented by $X(t+1)$, the position of the prey denoted by $X_{rabbit}(t)$ at iteration $t$. The randomly selected hawk is indicated by $X_{rand}(t)$, while the hawk's positions vector is represented by $X(t)$ at the current iteration. Numbers of random are generated ($r_1$, $r_2$, $r_3$, $r_4$ and $q$) in the interval (0,1) and then updated each iteration. Further, the upper and lower bounds are denoted by $UB$ and $LB$, respectively.

$$X(t+1) = \begin{cases} X_{rand}(t) - r_1 |X_{rand}(t) - 2r_2 X(t) \\ \qquad q \geq 0.5 \\ (X_{rabbit}(t) - X_m(t)) - r_3(LB + r_4(UB - LB)) \\ \qquad q < 0.5 \end{cases} \tag{4}$$

In the current population, $X_[m](t)$ represents the average position of hawks, based on equation 5, where, in the current

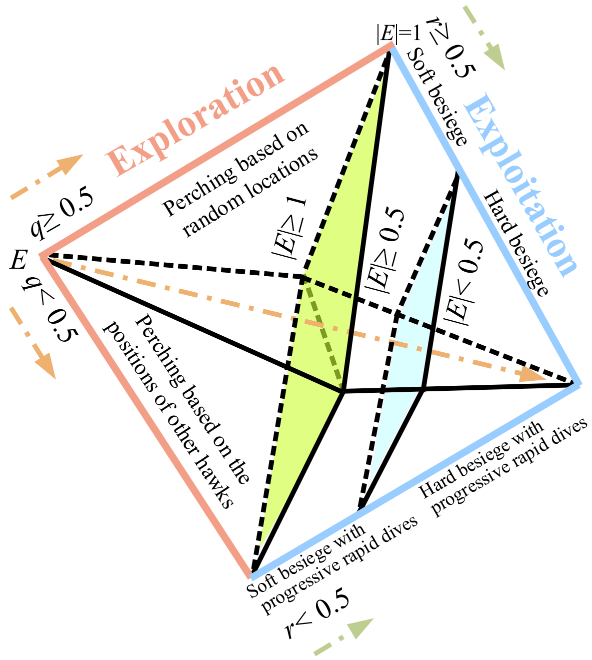



**FIGURE 1.** **HHO different phases [58].**

iteration, $X_i(t)$ represents the position of the hawk $(i)$, whereas the entire number of hawks denoted by $N$.

$$X_m(t) = \frac{1}{N}\sum_{i=1}^{N} X_i(t) \tag{5}$$

As part of the exploitation phase, Harris hawks perform surprise pounces on their prey. When the prey tries several times to escape, hawks adapt to the prey's escape behavior by changing their chasing strategies. As a result, hawks follow four different chasing techniques, including, Hard Besiege, Soft Besiege, Hard Besiege involving progressive rapid dives, and Soft Besiege involving progressive rapid dives. Preys lose energy during the escape from haunts, so picking either of the four strategies depends on their energy $E$. Therefore, it can be interpreted as a change in exploitative behavior. Modeling the energy of prey can be accomplished using equations 6, where the prey's initial energy is $E_0$, the maximum number of iterations represented by $T$.

$$E = 2E_0(1 - \frac{t}{T}) \tag{6}$$

$$X(t+1) = \Delta X(t) - E|JX_{rabbit}(t) - X(t)| \tag{7}$$

$$X(t+1) = \begin{cases} Y & if\ F(Y) < F(X(t)) \\ Z & if\ F(Z) < F(X(t)) \end{cases} \tag{8}$$

$$X(t+1) = X_{rabbit}(t) - E|\Delta X(t)| \tag{9}$$

### 3) WORD EMBEDDING

An embedding is a dense, chunk of text, document, or low-dimensional representation of words presented as raw vectors containing real values [59]. There are two types of embedding models: predictive-based and frequency-based. So, a WE represents words as vectors of real numbers by mapping their semantic properties to a latent feature vector. Thus, each latent feature of a word is represented by a different dimension of the embedding.

Usually, the textual data encoded in a string format cannot be used by machine learning algorithms unless it is converted (encoded) into a suitable representation that can be comprehended by the algorithms. There have been several text representation techniques proposed in the literature, including frequency-based techniques (e.g., TFIDF), and prediction-based techniques (e.g., FastText, Word2Vec, and BERT). Previous studies have demonstrated that prediction-based techniques are more effective than frequency-based ones in capturing the meaning of words. Hence, in the learning phase, it is essential to numerically represent the raw textual data. That is to say, in the process of WE, words are encoded as dense vectors by representing them as numerical vectors.

In an example, an embedding representation of the word "car" can be described by the following vector: [0.40, 2, - 1, 0.5, 3.2, - 9.7, 62, 44.6, 1, 0], in which the length of the vector is determined beforehand. An option for creating WEs is to use shallow neural networks, where the WE vectors are represented by the weights of the hidden layers in these networks. By using the embedding method, smarter features can be produced to express implicit meaning (e.g., questions) in texts. Therefore, by using neural networks, the embedding vectors can be tuned in order to be similar when the words they include have similar meanings. In short, similar words with the same vector representations will have high similarity scores when they appear in the same context.

There are many types of WE techniques, in this study, we will use the BERT method:

- BERT
  The Bidirectional Encoder Representations from Transformers (BERT) is known as one of the most powerful word representations and contexts. BERT used the attention mechanism to perform and depend on the transformers' methodology. Through the attention mechanism, it can identify the word's relationship in a particular sentence. Thus, considering the various context for an exact word. Due to the fact that some words can have several embeddings in accordance with the context. For instance, the word 'bank' can be used in different contexts and also possess diverse embedding. Using the word 'bank' in a 'bank account' has a different context and embedding from using it in 'bank of the river'. Furthermore, BERT employs word-piece tokenization, where the word walk is possible to have two-word pieces, walk and ****ing. The benefit of such tokenization is, that if the word does not exist in the BERT vocabulary, then it can be divided into different word pieces. Accordingly, might have embeddings for irregular and rare words. Another advantage of using

the BERT WE is that it can be applied to multilingual data. In this study, three languages will be used, namely English, Spanish, and Arabic.
The BERT model has two modes that can be used:

- Creating embeddings for words and using them as input.
- Fine-tuning the pre-trained model through applying task-precise corpus.

#### 4) PRE-TRAINED WE (TRANSFER LEARNING)

Transfer learning has been a major factor in driving growth in NLP adoption [60]. Transfer learning can be defined, in the context of NLP, as a trained model on a given dataset and adapting and applying that model to another dataset (problem). In other words, it is about transferring the learning of a specific task to another. This can happen thanks to the pre-trained concept. Such learnings might be either embeddings or weights. Hence, if the learning was for embeddings then it is known as pre-trained WEs, whereas if it was weighted learning it is identified as a pre-trained model. In our case, the focus will be on pre-trained WEs.

The pre-trained WEs cable of capturing the syntactic and semantic meaning of a word since they are trained on bigger datasets, thus, boosting the NLP models' performance. Moreover, made it simple for others, particularly individuals with no resources or time to create their own NLP model from scratch. So, reusing existing embeddings is more appealing because the process doesn't require extensive training or expertise on how to learn embeddings. Especially if there were problems such as the lack of training data and the existence of many trainable parameters. There are many pre-trained models that are trained on different domains (multilingual data) and ready to be used on other similar problems such as FastText and BERT.

### B. DATA DESCRIPTION AND COLLECTION

This stage provides a description of the data and outlines the data collection procedures. The data used in this study consists of reviews for various products and services, written in different languages and obtained from the Twitter online social network. Twitter was chosen as the data source due to its status as the largest online community for writing reviews. Online social networks are user-friendly web-based platforms that attract a larger user base compared to other review websites. As a result, they have a greater potential to influence individuals' opinions on various subjects, particularly when it comes to providing feedback on products and services. The data collection process was conducted using the twitteR package in RStudio, which enables access to Twitter's API.

The data was collected using a term-based procedure, where the names of the products or services were utilized. The data collection process also considered different languages. Specifically, three languages were focused on: English, Spanish, and Arabic. Each language was gathered and saved in a separate text file for each product, as specific preprocessing steps were required based on the characteristics of each language. Subsequently, the reviews were consolidated into CSV files to prepare them for further processing.

**TABLE 3.** List of generated datasets.

| Data | Instances |
|---|---|
| English | 9900 |
| Spanish | 2000 |
| Arabic | 468 |
| Multilingual | 12368 |

Taking into account different languages will aid in understanding the feedback from various regions regarding a particular product. This, in turn, allows for tailored decisions and solutions based on the specific requirements of each region. The dataset comprises over 10,000 reviews covering medical, food, and entertainment products and services. Table 3 provides an overview of the dataset details.

### C. DATA PREPARATION

After gathering the data into separate CSV files, the labeling process will commence. This step involves creating two copies of each file: one for preprocessing the original content and another for translating the text during the labeling procedure. The labeling process entails three experts who will review and annotate the data. Their task is to review the data and assign appropriate labels. The labels used for the data are 'spam' or 'real'. Once the labeling is complete, the labels will be saved and subsequently added to the data.

Furthermore, the original file will undergo various cleaning and formatting processes and sometimes may have an imbalanced problem [61]. Firstly, stop words will be removed, as they have no impact on the meaning of the text. Different sets of stop words will be used for each language. Thus, they will be "the, a, are, be", "la, de, para, que" and "من ، لم ، كان ، في ، ال" for English, Spanish and Arabic languages, respectively. Also, distinct stemming methods have been employed for each language [62]. The selection of the best method was determined through various tests. For English and Spanish, the Snowball stemming method [63] was utilized, while for the Arabic language, the Arabic Light stemming algorithm [64] was used. The application of text stemming and stop-word removal serves to eliminate numerous unnecessary data features, ultimately reducing the total number of extracted features and enhancing the feature selection process.

In the final step, the word representation process is carried out through linguistic analysis. In this study, WE is used as a word representation technique. The BERT WE method has been applied in this phase [65]. The reason for selecting such a method is that it allows for multilingual WE.

As for the second phase, due to the size of the data, pre-trained word embeddings (WE) were also utilized. There are several advantages to using pre-trained WE, including,

**TABLE 4.** Number of features.

| Data | English | Spanish | Arabic |
|---|---|---|---|
| NGram-3 | 3828 | 2485 | 9298 |
| TFIDF | 3752 | 1219 | 1829 |
| WE100 | 100 | 100 | 100 |
| WE400 | 400 | 400 | 400 |
| Words | 1697 | 1233 | 1829 |

but not limited to, saving time and resources, achieving better performance, and gaining access to specific knowledge. In this phase, two different procedures were employed: first, separate pre-trained WE models were used for each language, and second, a multilingual pre-trained WE model was applied to the entire dataset.

Each pre-trained WE model was already created with a specific number of dimensions. Then, after selection, the pre-trained word embeddings were fine-tuned using the data. In other words, an embedding matrix was generated by assigning the terms to the pre-trained WE vocabulary. For the multilingual approach, BERT multilingual embeddings were utilized.

To compare with other word representation methods, five versions of each dataset were generated along with the WE technique. These versions include NGram-3, TFIDF, WE100, WE400, and Words (One-hot encoding).The number of features for each dataset can be found in Table 4.

### D. PROPOSED APPROACH (WSVM-HHO)

To optimize SVM parameters and feature weighting before utilizing the HHO algorithm, it is necessary to first design the solution representation. This includes designing the individual representation as well as choosing the fitness function. After briefly discussing each of these design problems, we will present and explain the overall system architecture of our proposed model.

#### 1) SOLUTION REPRESENTATION

In addition to its usage as a search algorithm, HHO is designed to solve complex problems. In our case, it is utilized to address two specific issues. The first issue involves searching for the optimal C (Cost) and $\gamma$ (Gamma) parameters, while the second issue focuses on weighting the data features.

As a result, HHO produces elements for both the parameters (C and $\gamma$) along with the D number of features per dataset. Based on a vector of D + 2 real numbers, an initial interval of [0,1] can be generated. The 2 real numbers in the vector correspond to C and $\gamma$ parameters. These parameters' search space differs from their original scale and, accordingly, the parameters are converted and scaled to [0,35000] and [0,32] for C and $\gamma$, respectively. In order to perform this transformation, Min-max normalization is used as shown in Eq. 10, while the elements of the features (the second part of the vector) will be weighted. For each instance in the vector, the weighting is determined by multiplying each element in the vector by its corresponding feature. In the case of a simple dataset of five instances, the values of the first feature of all instances will be multiplied by the value of the first element of the HHO solution.

As a result, HHO produces elements for both the parameters (C and $\gamma$) along with the D number of features per dataset. By using a vector of D + 2 real numbers, an initial interval of [0,1] can be generated. The 2 real numbers in the vector correspond to the C and $\gamma$ parameters. However, the search space for these parameters differs from their original scale. To accommodate this, the parameters are converted and scaled to [0,35000] and [0,32] for C and $\gamma$, respectively. To perform this transformation, Min-max normalization is applied, as shown in Eq.10. Additionally, the elements of the features (the second part of the vector) are weighted. Each instance in the vector is multiplied by its corresponding feature to determine its weighting. For example, in a simple dataset with five instances, the values of the first feature of all instances would be multiplied by the value of the first element of the HHO solution.

$$B = \frac{A - min_A}{max_A - min_A}(max_B - min_B) + min_B \tag{10}$$

#### 2) FITNESS EVALUATION

HHO receives feedback based on the fitness function in each iteration. In this case, the evaluation is proportional to the SVM classification accuracy. Therefore, it can be calculated as follows:

$$fitness(I_i^t) = \frac{1}{k}\sum_{k=1}^{k}\frac{1}{N}\sum_{j=1}^{N}\delta(c(x_j), y_j) \tag{11}$$

where $c(x_j)$ represents the accuracy of the $j$th instance and the label of the actual class for that instance denoted with $y_j$. The relationship between $c(x_j)$ and $y_j$ is denoted by $\delta$, thus, $\delta = 1$ when $c(x_j) = y_j$, otherwise $\delta = 0$. In the testing set, $N$ is the number of instances, while $K$ represents the number of folds (data parts).

#### 3) SYSTEM ARCHITECTURE

Every dataset is divided into training and testing sets as part of our proposed approach. The splitting criterion depends on the number of experiments conducted. For each experiment, the dataset is divided into $k$ parts, where $k$ represents the number of experiments. During training, $k - (1/k)$ parts are used for training the model, while $1/k$ parts are used for testing the results. The aim is to ensure that both the training and testing sets are as diverse as possible to produce the best possible model.

The next step involves the involvement of the HHO algorithm. In this step, a random vector of real numbers is generated at the beginning of each iteration using HHO. These numbers correspond to C, $\gamma$, and the feature weights. The training process of the SVM classifier starts using the weighted training set.

To enhance the robustness of the model, both internal and external validity measures have been implemented. For internal validity, an inner 3-cross-validation is utilized during the training phase to generate a stable model. Regarding external validity, the algorithm is run 10 times, with each run using a different part of the data based on a 10-cross-validation criterion. In other words, the testing phase is carried out on unseen data to produce a reliable model.

HHO receives the accuracy from the SVM classifier as its fitness value after the completion of the training process. In our case, we repeat the aforementioned steps until the HHO termination criterion is satisfied (number of iterations). The best individual is generated by HHO during the testing phase when the maximum number of iterations is reached. Ultimately, all the mentioned stages are repeated $k$ times, and then the average of the values is computed. The entire process of the proposed approach is illustrated in Figure 2.

### E. EVALUATION METRICS

As part of this stage, we will evaluate the predictive performance of the best-obtained SVM. The measurement below will be used to assess the detection problem, which is a binary classification problem. The confusion matrix is used as a reference for calculating the accuracy. Spam refers to the positive classes, while real refers to the negative classes.

- Accuracy: determined by dividing the number of spam and real reviews correctly classified by the total number of classified reviews.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{12}$$

## IV. EXPERIMENTS AND RESULTS

This section aims to provide an in-depth comparison of the classification performance of WSVM. All experiments were conducted using a PC equipped with an AMD Ryzen 5 5600X 3.7GHz CPU and 16.0GB RAM. MATLAB 2016 software was utilized to run all the algorithms. Additionally, 10-fold cross-validation was employed for both training and testing sets. The following experiments were conducted:

- Experiment I: Focuses on comparing the performance of well-known classification algorithms on the generated datasets. The goal is to examine the initial results of the data and evaluate the performance of various classification algorithms on the dataset.
- Experiment II: The purpose of this analysis is to examine the impact of applying various metaheuristics (MVO, GA, PSO, GOA, SSA, and HHO) to SVM in order to determine the most effective metaheuristic algorithms for this problem. The objective is to assess how the application of these metaheuristics affects the performance of SVM and identify the best metaheuristic algorithms that yield optimal results for solving this specific problem.
- Experiment III: In this comparison, we will evaluate the proposed WSVM-HHO approach and another version of SVM combined with HHO, specifically HHO-FsSVM (HHO-SVM with Feature Selection). The aim is to assess the performance of both approaches and determine their effectiveness in solving the problem at hand.
- Experiment IV: This experiment aims to compare the performance of the proposed WSVM-HHO algorithm with other well-known classifiers from the literature, which are combined with various metaheuristic algorithms. The goal is to assess the effectiveness of the WSVM-HHO algorithm in comparison to existing classifiers when paired with different metaheuristic algorithms.

After the experiments, we deeply explain and analyze the reviews' context before and after the pandemic, as well as generate a new dataset with statistical features.

The importance of selecting appropriate parameters has a significant impact on the efficiency of the model [66]. Further, the parameter settings of the used algorithms can be found in Table 6. Moreover, it is worth noting that the algorithm parameters were obtained from the original papers.

### A. EXPERIMENT I: RESULTS OF TRADITIONAL CLASSIFICATION MODELS

The first experiment depicts the performance of traditional or classic classification models in analyzing the newly generated datasets. In other words, three well-known classifiers, namely J48, k-NN, and NB, have been run on different versions of the data. Each classifier runs on five versions of each linguistic data (Arabic, English, Spanish, and Multilingual). This approach allows us to observe the pattern of the results on each dataset for well-known and already familiar models, thereby enhancing our understanding of the new data.

Table 7 presents the results of the first language data, which is the Arabic dataset, considering five versions of the dataset. As can be seen, the J48 algorithm achieved the highest accuracy for all versions, while k-NN obtained the second-best accuracy for the WE100 version. In general, the best result for NB was obtained on the NGram-3 version, k-NN obtained it for the WE100 version, while all results were the same for J84. This occurred because the algorithm failed to recognize all the classes for this data due to the limited number of instances.

In Table 8, the results depict the performance of the algorithms on the English dataset. Unlike the Arabic data, this dataset contains a large number of instances due to the popularity of the language in product reviews. The best results were achieved by k-NN, while the second-best performance was attained by J48. The algorithms achieved their best results in different versions: J48 on the NGram-3 version, k-NN on the WE400 version, and NB on the WE100 version. Similar to the previous dataset, the English dataset also achieved the highest accuracy when the PT-WE version (WE400) was used.

Regarding the Spanish dataset, the results of the five versions are shown in Table 9. As illustrated, the highest accuracy was achieved by J48, followed by k-NN and NB,

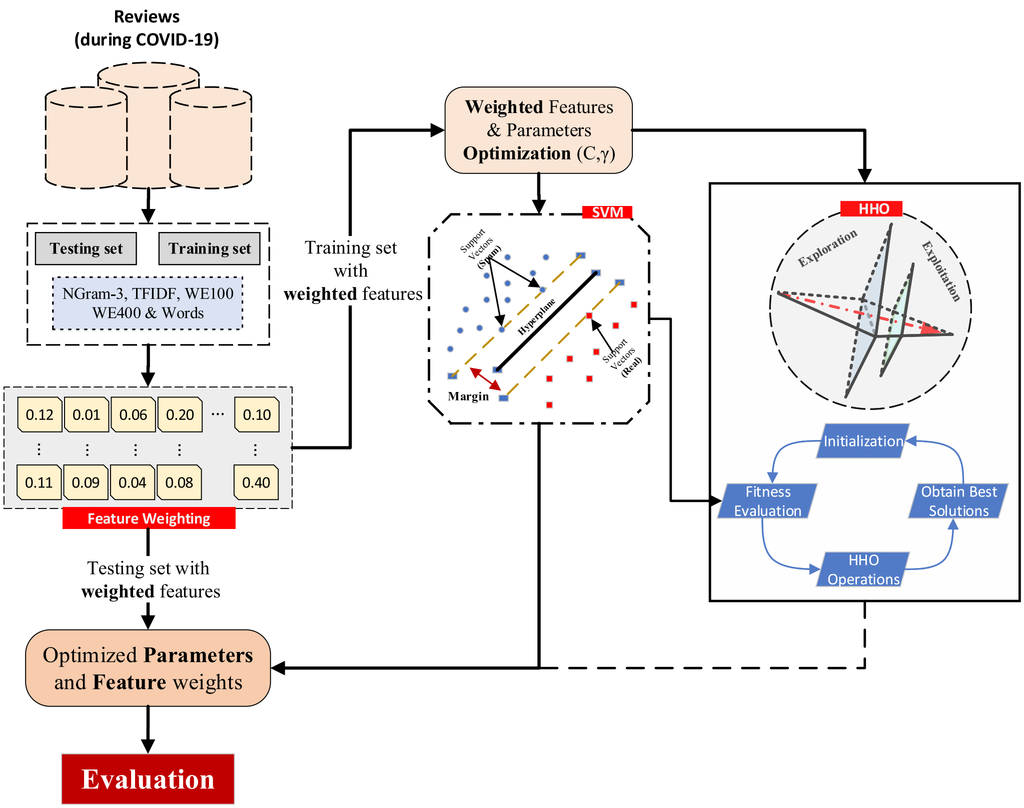



**FIGURE 2.** A schematic diagram illustrates the flow of the proposed approach.

**TABLE 5.** Meaning of symbols.

| Symbol | Meaning |
|---|---|
| $k(.)$ | Kernel function |
| $x_i,\ x_i'$ | Observations of two vectors for inner product |
| $\sum_{j=1}^{p}$ | Summation over the index $j$ from 1 to $p$ |
| $d$ | Degree parameter for polynomial kernel |
| $\gamma$ | Parameter for RBF kernel |
| $\exp(x)$ | Exponential function with base $e$ |
| $(x_{ij} - x_{ij}')^2$ | Squared difference between $x_{ij}$ and $x_{ij}'$ |
| $\varphi(x_i)$ | Transformation of $x_i$ in the feature space |
| $X(t+1)$ | Vector of hawks' positions at time $t+1$ |
| $X_{rabbit}(t)$ | Position of the prey at iteration $t$ |
| $X_{rand}(t)$ | Randomly selected hawk |
| $X(t)$ | Hawk's positions vector at the current iteration |
| $r_1,\ r_2,\ r_3,\ r_4,\ q$ | Random numbers generated in the interval (0, 1) |
| $UB$ | Upper bound |
| $LB$ | Lower bound |
| $X_m(t)$ | Average position of hawks in the current population |
| $X_i(t)$ | Position of hawk $i$ in the current iteration |
| $N$ | Total number of hawks |

respectively. As for the best results for each algorithm, J48 attained its highest result when running on the NGram-3 version, k-NN when running on the NGram-3 version as well, and NB for the Words version.

Furthermore, Table 10 displays the results for the combination of the previous datasets (Multilingual-data). As shown in the table, J48 achieved the highest accuracy of 81.05, while k-NN obtained the second-highest accuracy of 78.18. The

**TABLE 6.** Initial parameters of the metaheuristic algorithms.

| Algorithm | Parameter | Value |
|---|---|---|
| MVO | Minimum wormhole existence ratio | 0.2 |
| | Maximum wormhole existence ratio | 1 |
| GA | Crossover ratio | 0.9 |
| | Mutation ratio | 0.1 |
| | Selection mechanism | Roulette wheel |
| PSO | Acceleration constants | [2.1, 2.1] |
| | Inertia w | [0.9, 0.6] |
| GOA | cMin | 0.00001 |
| | cMax | 1 |
| SSA | $c_1$ | [0-1] |
| | $c_2$ | [0-1] |

**TABLE 7.** Results of the traditional classifiers for the Arabic dataset processed with different vectorization methods.

| Version | J48 | k-NN | NB |
|---|---|---|---|
| NGram-3 | **87.2** | 38.37 | 85.92 |
| Tf-IDF | **87.2** | 61.4 | 83.36 |
| WE100 | **87.2** | 86.99 | 77.39 |
| WE400 | **87.2** | 86.78 | - |
| Words | **87.2** | 61.4 | 81.02 |

**TABLE 8.** Results of the traditional classifiers for the English dataset processed with different vectorization methods.

| Version | J48 | k-NN | NB |
|---|---|---|---|
| NGram-3 | 84.30 | 81.25 | 61.00 |
| Tf-IDF | 84.41 | 80.93 | 69.01 |
| WE100 | 81.86 | 85.02 | 78.09 |
| WE400 | 81.83 | **85.28** | - |
| Words | 84.33 | 82.58 | 69.00 |

**TABLE 9.** Results of the traditional classifiers for the Spanish dataset processed with different vectorization methods.

| Version | J48 | k-NN | NB |
|---|---|---|---|
| NGram-3 | 64.76 | 52.22 | 60.55 |
| Tf-IDF | **64.41** | 55.27 | 57.61 |
| WE100 | 56.22 | 54.22 | 56.87 |
| WE400 | 55.12 | 55.57 | - |
| Words | 65.11 | 55.72 | 58.70 |

results, in general, were relatively close, with J48 achieving its best results in the WE400 version, k-NN in the Words version, and NB when running on the Tf-IDF version.

From these experiments, we can draw several conclusions. Firstly, the generated datasets yield good results for traditional classification models, which validates the reliability of the data. Secondly, each dataset exhibits varying results across different versions. This highlights the importance of employing different word representation methods for each

**TABLE 10.** Results of the traditional classifiers for the Multilingual dataset processed with different vectorization methods.

| Version | J48 | k-NN | NB |
|---|---|---|---|
| NGram-3 | 79.71 | 76.66 | 63.19 |
| Tf-IDF | 78.05 | 77.04 | 72.02 |
| WE100 | 77.46 | 77.27 | 70.86 |
| WE400 | **81.05** | 76.14 | - |
| Words | 79.97 | 78.18 | 69.95 |

dataset, rather than relying on a single technique. Consequently, the WE method proved to be the most effective.

### B. EXPERIMENT II: COMPARISON OF THE PROPOSED APPROACH AGAINST OTHER METAHEURISTIC ALGORITHMS

Once we have studied the performance of basic classifiers, the next experiment aims to investigate the improvements that several optimization algorithms can bring to strong classifiers like SVM. Therefore, in the second experiment, various metaheuristic algorithms were compared using SVM as the main classifier. The initial parameters for the metaheuristic algorithms can be found in Table 6.

Table 11 illustrates the results of the Arabic data and its versions. The WSVM-HHO achieved the best results among all other combinations for the TF-IDF version, while the second-best result was obtained by SSA-WSVM for the NGram-3 version. The comprehensive performance indicates a significant increase compared to the previous experiment (using traditional classifiers). For instance, the best result for the NGram-3 data was 87.20, which increased to 89.565 here, showing an improvement of 2.36.

For the English data, Table 11 presents the results for the data and its versions. The highest accuracy was achieved by WSVM-HHO for the WE400 version, followed by MVO-WSVM, GOA-WSVM, SSA-WSVM, PSO-WSVM, and GA-WSVM for the same version, respectively. Additionally, the overall results demonstrate the successful combination of HHO and WSVM for all versions, outperforming other combinations.

Regarding the Spanish data, the results demonstrate a decrease in performance compared to other languages, as shown in Table 11. Furthermore, WSVM-HHO outperforms the rest of the algorithms, while SSA-SVM obtains the second-highest performance. It can also be observed that the best results are achieved when using the WE400 version.

In the Multilingual-data version, the results demonstrate similar performance to the English and Spanish data, as illustrated in Table 11. The best result was achieved by WSVM-HHO for the WE400 version, followed by MVO-WSVM, PSO-WSVM, SSA-WSVM, GOA-WSVM, and GA-WSVM, respectively. The table also highlights that WSVM-HHO is the algorithm that consistently achieved the best results for most versions.

**TABLE 11.** Results of different metaheuristic algorithms and WSVM for all datasets.

| Versions | MVO-WSVM | | GA-WSVM | | PSO-WSVM | | GOA-WSVM | | SSA-WSVM | | WSVM-HHO | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Avg | Std | Avg | Std | Avg | Std | Avg | Std | Avg | Std | Avg | Std |
| Arabic | | | | | | | | | | | | |
| NGram-3 | 88.700 | 4.015 | 88.904 | 4.713 | 89.348 | 4.582 | 88.927 | 5.732 | 89.561 | 4.053 | 88.918 | 4.099 |
| TFIDF | 88.266 | 4.861 | 88.062 | 4.030 | 89.343 | 3.605 | 88.714 | 5.203 | 89.343 | 6.722 | **89.565** | 4.522 |
| WE100 | 87.012 | 4.516 | 86.998 | 5.791 | 86.776 | 3.876 | 86.776 | 5.668 | 86.785 | 7.348 | 86.776 | 7.568 |
| WE400 | 86.790 | 6.392 | 87.211 | 6.009 | 87.216 | 4.235 | 86.984 | 4.574 | 87.211 | 3.155 | 87.197 | 3.666 |
| Words | 88.904 | 4.265 | 89.112 | 3.880 | 88.488 | 4.504 | 88.918 | 2.783 | 88.686 | 4.761 | 87.410 | 5.291 |
| English | | | | | | | | | | | | |
| NGram-3 | 86.072 | 0.960 | 85.426 | 1.447 | 84.072 | 3.370 | 84.991 | 3.098 | 85.739 | 0.959 | 85.951 | 1.050 |
| TFIDF | 84.921 | 3.641 | 80.365 | 5.066 | 83.073 | 4.907 | 77.255 | 4.429 | 85.254 | 1.442 | 86.678 | 1.299 |
| WE100 | 86.385 | 1.446 | 85.769 | 0.774 | 86.193 | 1.059 | 86.486 | 1.189 | 85.719 | 1.091 | 87.153 | 1.146 |
| WE400 | 87.910 | 1.427 | 87.001 | 0.631 | 87.345 | 0.829 | 87.961 | 1.204 | 87.507 | 1.061 | **88.163** | 1.125 |
| Words | 86.486 | 1.424 | 85.486 | 1.795 | 86.456 | 1.151 | 85.012 | 2.968 | 85.850 | 1.067 | 85.820 | 1.148 |
| Spanish | | | | | | | | | | | | |
| NGram-3 | 55.576 | 10.986 | 51.725 | 8.641 | 58.277 | 12.197 | 54.014 | 10.236 | 70.863 | 9.084 | 70.514 | 9.487 |
| TFIDF | 57.226 | 13.867 | 56.513 | 13.581 | 59.965 | 14.676 | 51.526 | 8.283 | 63.560 | 14.472 | 70.726 | 13.207 |
| WE100 | 67.015 | 1.830 | 66.316 | 2.416 | 66.716 | 2.298 | 66.065 | 3.722 | 66.911 | 5.560 | 68.364 | 4.729 |
| WE400 | 70.068 | 4.863 | 69.366 | 4.751 | 70.516 | 3.970 | 70.514 | 2.231 | 69.262 | 4.929 | **71.913** | 3.392 |
| Words | 54.778 | 11.879 | 56.524 | 12.316 | 58.664 | 13.654 | 53.779 | 12.123 | 70.161 | 12.010 | 66.060 | 15.237 |
| Multilingual | | | | | | | | | | | | |
| NGram-3 | 81.028 | 1.240 | 79.032 | 1.970 | 80.600 | 1.855 | 78.555 | 2.256 | 80.543 | 1.309 | 81.125 | 1.159 |
| TFIDF | 81.667 | 1.160 | 77.116 | 3.291 | 81.820 | 1.057 | 77.343 | 4.287 | 80.179 | 2.965 | 81.885 | 1.185 |
| WE100 | 82.411 | 0.865 | 81.578 | 0.871 | 81.829 | 1.294 | 81.764 | 1.427 | 82.524 | 0.989 | 82.855 | 0.870 |
| WE400 | 83.639 | 1.670 | 83.025 | 1.041 | 83.284 | 1.192 | 83.171 | 1.555 | 83.267 | 1.345 | **84.270** | 1.429 |
| Words | 83.849 | 0.969 | 81.837 | 2.958 | 81.998 | 4.073 | 79.064 | 4.397 | 83.251 | 0.778 | 83.203 | 1.342 |

Figures 3, 4 and 5 illustrate the convergence curve results for all algorithms. In most datasets, WSVM-HHO exhibits the best convergence.

The results of all datasets indicate an improvement compared to the previous experiment. This demonstrates the quality and efficiency of the metaheuristic algorithms in enhancing detection accuracy.

### C. EXPERIMENT III: COMPARISON OF THE PROPOSED WSVM-HHO APPROACH AGAINST HHO-FsSVM

To demonstrate the efficiency of WSVM-HHO, we conducted another experiment using a new version that combines HHO and SVM together, known as HHO-SVM with Feature Selection (HHO-FsSVM). In HHO-FsSVM, we perform two tasks: optimizing the SVM parameters and feature selection. HHO-FsSVM differs from the proposed approach (WSVM-HHO) in the treatment of features. In other words, HHO-FsSVM selects and determines the best subset of features and tests the model using these features, unlike WSVM-HHO, where the features are weighted.

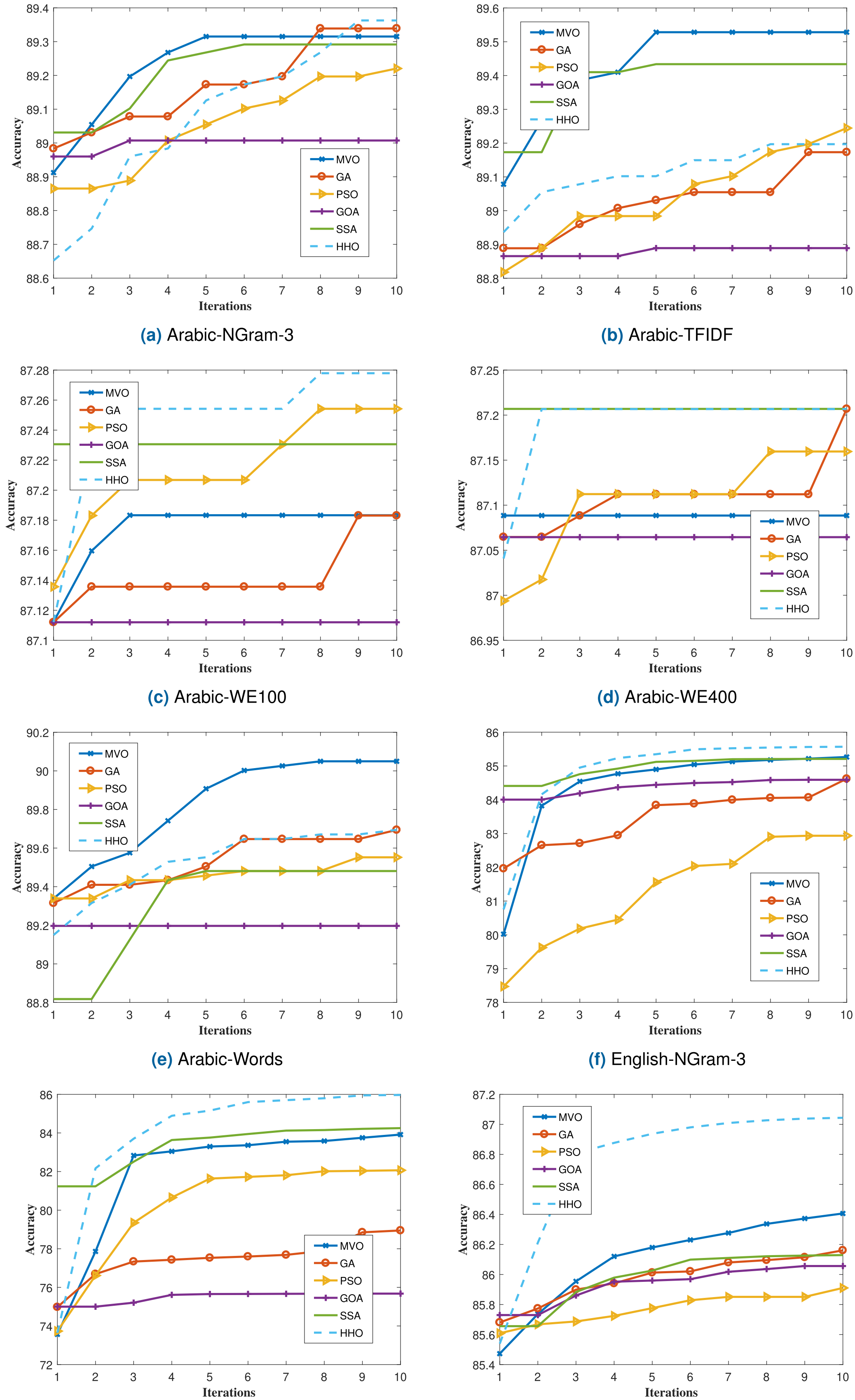
Table 12 illustrates the performance of the two approaches on the datasets versions. In the first data (Arabic), all results of the five versions show the superiority of the proposed approach to other methods. The WSVM-HHO achieved 88.81%, 89.56%, 86.77%, 87.19%, and 87.41 for NGram-3, TFIDF, WE100, WE400, and Words versions, respectively. Whereas in the English data, the WSVM-HHO obtained the best results in most versions with 85.70%, 86.67%, 87.15%, and 88.16% for NGram-3, TFIDF, WE100, and WE400, respectively, while the HHO-FsSVM has the best result in Words version with 86.78%. For the Spanish data, three versions (TFIDF, WE100, and WE400) accomplish the better results for the WSVM-HHO approach, whilst, NGram-3 and Words versions obtained the best results for HHO-FsSVM. On the other hand, the WSVM-HHO acquired the highest accuracy on all versions with 81.12%, 81.28%, 82.85%, 84.27%, and 83.20% for NGram-3, TFIDF, WE100, WE400, Words versions, respectively.

In summary, based on the above analysis, WSVM-HHO achieved the best results in 17 out of the 20 versions. On the other hand, HHO-FsSVM only obtained the best results in three versions. This clearly demonstrates the superiority of WSVM-HHO even when compared to HHO-FsSVM.

### D. EXPERIMENT IV: COMPARISON OF THE PROPOSED WSVM-HHO APPROACH AGAINST OTHER CLASSIFIERS COMBINED WITH METAHEURISTIC ALGORITHMS

In this phase, a comparison between well-known classifiers combined with different metaheuristic algorithms and our proposed approach is performed. The comparison is conducted on the WE400 version, which yielded the best results in previous experiments. The classifiers used are EVO-J48, EVO-RF, EVO-k-NN, EVO-NB, EVO-MLP, EVO-AdaBoost, and EVO-Bagging. These classification models have been chosen based on their good performance in the literature.

Figure 6 presents the results of these models along with the proposed approach. As shown in the table, WSVM-HHO achieved the highest accuracy of 89.56% for the Arabic data, followed by EVO-RF, EVO-k-NN, and EVO-MLP. For the




(a) Arabic-NGram-3

(b) Arabic-TFIDF

(c) Arabic-WE100

(d) Arabic-WE400

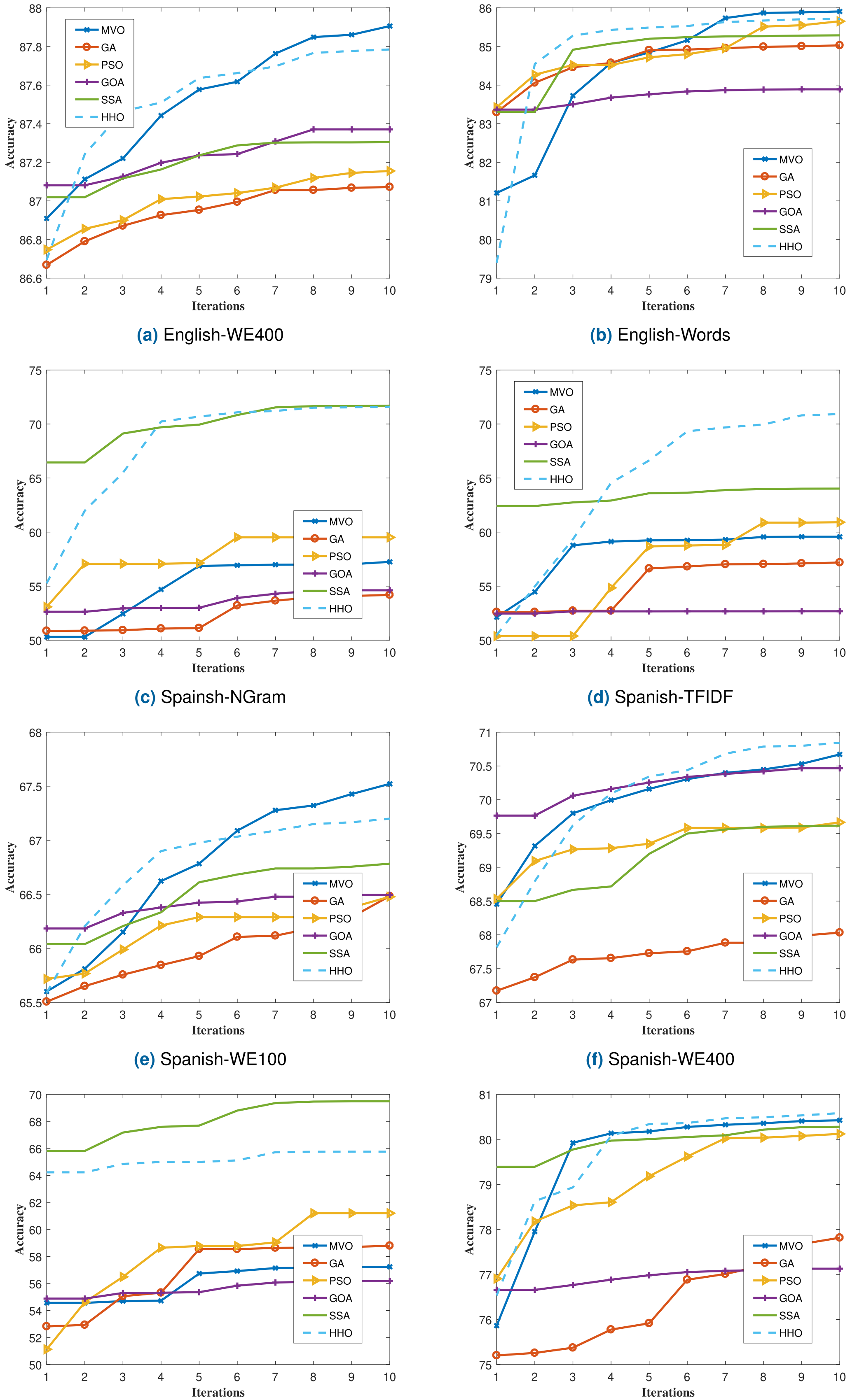
(e) Arabic-Words

(f) English-NGram-3

(g) English-TFIDF

(h) English-WE100

**FIGURE 3. Convergence curve charts for HHO and other algorithms based on 5 versions of Arabic and English datasets.**




(a) English-WE400 (b) English-Words

(c) Spainsh-NGram (d) Spanish-TFIDF

(e) Spanish-WE100 (f) Spanish-WE400

(g) Spanish-Words (h) Multi-NGram

**FIGURE 4. Convergence curve charts for HHO and other algorithms based on 5 versions of English, Spanish, and Multilingual datasets.**

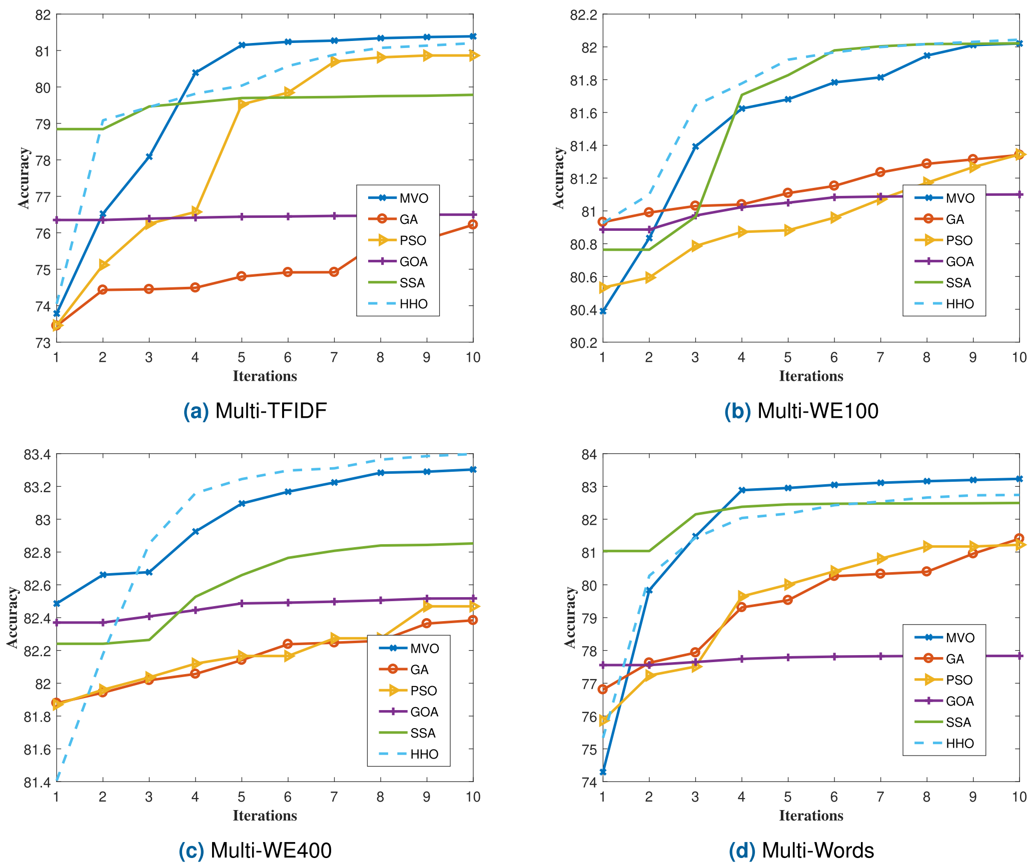



**FIGURE 5.** Convergence curve charts for HHO and other algorithms based on 5 versions of the multilingual datasets.

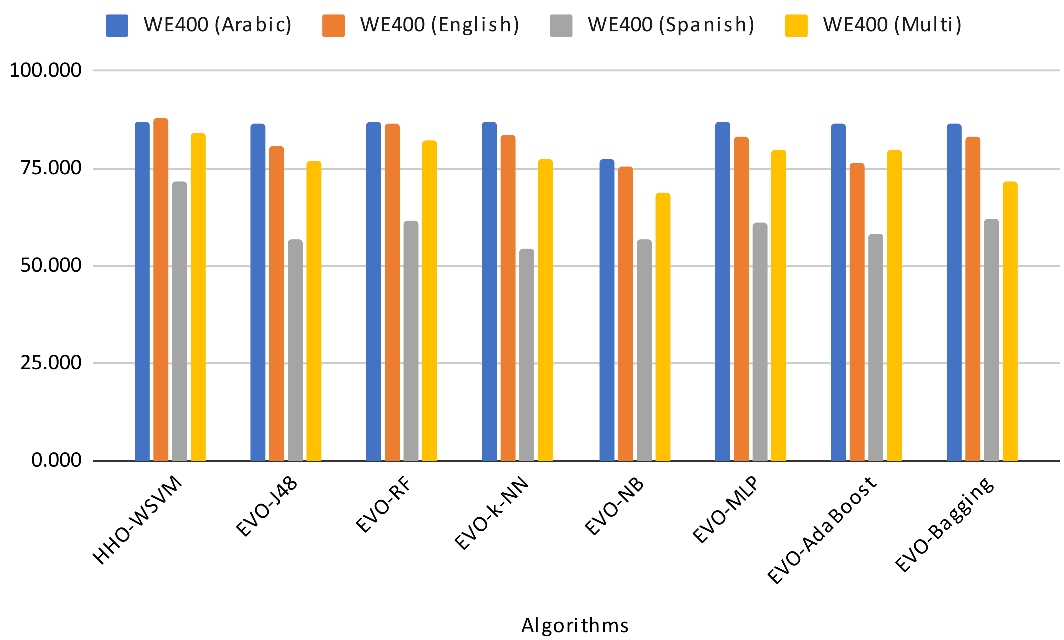



**FIGURE 6.** Results of the proposed approach (WSVM-HHO) and various classifiers combined with evolutionary algorithms.

English data, WSVM-HHO also achieved the highest results, followed by EVO-RF with an accuracy of 86.547%. In the case of the Spanish data, WSVM-HHO attained the best results with an accuracy of 71.913%, while EVO-Bagging

**TABLE 12. Results of the proposed approach (WSVM-HHO) and the feature selection approach (HHO-FsSVM).**

| Arabic | | | | |
|---|---|---|---|---|
| Versions | WSVM-HHO | | HHO-FsSVM | |
| | Avg | Std | Avg | Std |
| NGram-3 | 88.918 | 4.099 | 87.636 | 2.600 |
| TFIDF | **89.565** | 4.522 | 88.478 | 4.536 |
| WE100 | 86.776 | 7.568 | 86.554 | 3.832 |
| WE400 | 87.197 | 3.666 | 86.984 | 3.295 |
| Words | 87.410 | 5.291 | 86.341 | 3.977 |
| English | | | | |
| Versions | WSVM-HHO | | HHO-FsSVM | |
| | Avg | Std | Avg | Std |
| NGram-3 | 85.951 | 1.050 | 84.870 | 2.844 |
| TFIDF | 86.678 | 1.299 | 84.506 | 4.721 |
| WE100 | 87.153 | 1.146 | 86.416 | 1.294 |
| WE400 | **88.163** | 1.125 | 87.981 | 1.609 |
| Words | 85.820 | 1.148 | 86.870 | 1.479 |
| Spanish | | | | |
| Versions | WSVM-HHO | | HHO-FsSVM | |
| | Avg | Std | Avg | Std |
| NGram-3 | 70.514 | 9.487 | 70.822 | 9.716 |
| TFIDF | 70.726 | 13.207 | 70.114 | 2.178 |
| WE100 | 68.364 | 4.729 | 67.267 | 2.439 |
| WE400 | **71.913** | 3.392 | 70.515 | 2.525 |
| Words | 66.060 | 15.237 | 66.415 | 2.769 |
| Multilingual | | | | |
| Versions | WSVM-HHO | | HHO-FsSVM | |
| | Avg | Std | Avg | Std |
| NGram-3 | 81.125 | 1.159 | 80.713 | 2.831 |
| TFIDF | 81.287 | 1.471 | 81.198 | 2.267 |
| WE100 | 82.855 | 0.870 | 82.863 | 0.921 |
| WE400 | **84.270** | 1.429 | 83.906 | 0.972 |
| Words | 83.203 | 1.342 | 82.548 | 4.981 |

obtained the second-highest accuracy. Similarly, WSVM-HHO outperformed other algorithms with an accuracy of 84.270% for the multilingual data, while EVO-RF obtained the second-highest accuracy of 82.249%. Overall, WSVM-HHO demonstrated its superiority over other approaches by achieving the highest accuracy in these experiments.

### *E. ANALYSIS AND DISCUSSION*

Four experimental phases were conducted in this study, each serving a specific purpose. The first experiment aimed to provide initial results by using well-known classic classifiers. This allowed us to assess the reliability of the data before conducting more advanced experiments. In the second experiment, notable improvements were observed compared to the previous experiment. The proposed approach was compared with different metaheuristic algorithms to determine the best one. Furthermore, the third experiment focused on evaluating the performance of the WSVM-HHO with feature selection version (HHO-FsSVM) in comparison to the proposed approach. Lastly, the fourth experiment involved comparing the proposed approach with other classification models. Across all experiments, the superiority of the proposed approach was consistently demonstrated. This can be attributed to the efficiency of HHO in handling data with vectorization values.

Moreover, all the experiments further confirm that the best vectorization method is the WE method. In this case, a pre-trained WE model was used due to the limited size of the datasets, which may not be sufficient for training a new WE model from scratch. It was observed that the pre-trained WE model yielded better results compared to other word representation methods such as NGram, TFIDF, and Word split.

Additionally, the WE method exhibited excellent performance in the multilingual dataset, indicating the effectiveness of using BERT-based WE in such scenarios. However, it is worth noting that the Arabic data achieved its best result when utilizing the TFIDF method. This suggests that the embedding methods for Arabic are still not as mature as those for other languages, although they still show promising competitiveness.

#### 1) REVIEWS' CONTEXT BEFORE AND AFTER THE PANDEMIC

Consumers rely on reading reviews to gather information about products or services before making a purchase. These reviews not only serve as a means for organizations to moderate products and refine business strategies but also provide an opportunity for them to enhance their overall quality. In the context of the pandemic, the nature of reviews has shifted, with a greater emphasis on health and technology products rather than fashion or beauty-related items. This change in the structure and context of reviews has also had an impact on spam reviews, which have evolved and become more sophisticated.

These modifications have affected both the features and words used in spam reviews. Across all languages, there has been a shift in customer focus towards medical and entertainment products. Consequently, spam reviews have followed this trend in an attempt to deceive consumers. Notably, our approach has identified the most significant features related to these topics, such as urgent treatment, COVID-19 protection, health benefits, safety, entertainment value, weight loss, and more. Adapting to these changes in review context and structure is crucial.

The use of pre-trained WE models has greatly aided in spam review detection. By leveraging context words to map target words, the WE model facilitates a sub-linear relationship within the word vector space. Consequently, the relationship with context becomes more meaningful, enhancing the accuracy and effectiveness of spam review detection.

**TABLE 13.** Example of statistical features.

| # | Features |
|---|---|
| 1 | No. of characters |
| 2 | No. of capitalized words |
| 3 | Max ratio of uppercase to lowercase letters of each word |
| 4 | Count of Spam Words |
| 5 | Count of Function Words |
| 6 | Count of duplicate words |
| 7 | Count of lowercase letters |
| 8 | Average word length |
| 9 | No. of single quotes |
| 10 | No. of commas |
| 11 | Total No. of sentences |
| 12 | No. of exclamation marks |
| 13 | No. of ellipsis |
| 14 | No. of colons |
| 15 | Count of lines |

#### 2) ADDITIONAL FEATURES (NEW DATA)

The overall results demonstrate a strong performance across all datasets. However, we sought to explore the possibility of further improving the results for the multilingual data by incorporating additional layers of features. Consequently, a new version of the multilingual dataset was created, which includes a combination of statistical features and WE400 embeddings. These statistical features (shown in Table 13) provide a more numerical representation of the text.

The new dataset (WE400StFE) was generated using the WE400 embeddings since it exhibited favorable results compared to other versions. The bullet points below present the results of the proposed approach on both the old dataset and the new dataset (WE400StFE). Notably, the results demonstrate a significant improvement compared to the previous version, with a difference of 6.722%. This highlights the effectiveness of incorporating these additional features for multilingual data.

- With old data (WE400), the average accuracy is 84.270 with 1.429 std.
- The new data (WE400StFE), the average accuracy is 90.992 with 0.026 std.

Even though the proposed approach has an excellent performance, there are some limitations that could affect the results. One of these limitations is the selection of the first parameters of the metaheuristic algorithms. While the other limitation is the huge effort to prepare the datasets.

## V. CONCLUSION AND FUTURE WORK

Due to the evolving nature of spam reviews during the pandemic, it has become essential to employ more advanced approaches for their detection. This paper presents a hybrid approach combining the Harris Hawks Optimization (HHO) algorithm with Weighted Support Vector Machine (SVM) for spam detection. The HHO algorithm is utilized to optimize the hyperparameters of the SVM and perform feature weighting. The study focuses on multilingual datasets (English, Spanish, and Arabic) during the Covid-19 pandemic period.

Several word representations are compared in this research, including NGram-3, TFIDF, WE100, WE400, and Words (One-hot encoding). The study is divided into four experimental phases: the first phase explores well-known classic classification algorithms, the second phase compares different evolutionary algorithms with SVM, the third phase introduces another version of the HHO-SVM approach, and the final phase examines the combination of other classifiers with metaheuristic algorithms.

The results consistently demonstrate the superior performance of the proposed approach across all stages of the study. It outperforms other methods in terms of spam detection accuracy and showcases its effectiveness in addressing the challenges posed by evolving spam reviews.

For future work, the authors plan to develop new word embeddings specifically tailored for reviews related to the Covid-19 pandemic. This will assist other researchers in capturing the unique context and language used in such reviews. Additionally, the authors intend to explore the application of the Twin Support Vector Machine (Twin SVM), Extreme Learning Machines, and Back-Propagation Neural Network in combination with metaheuristic algorithms to further enhance the spam detection phase. By incorporating these algorithms, they aim to improve the accuracy and efficiency of the detection process.

## REFERENCES

[1] A. M. Al-Zoubi, A. M. Mora, and H. Faris, "Spam reviews detection in the time of COVID-19 pandemic: Background, definitions, methods and literature analysis," *Appl. Sci.*, vol. 12, no. 7, p. 3634, Apr. 2022.

[2] M. K. Buckland, "Information as thing," *J. Amer. Soc. Inf. Sci.*, vol. 42, no. 5, pp. 351–360, Jun. 1991.

[3] P. De Pelsmacker, S. van Tilburg, and C. Holthof, "Digital marketing strategies, online reviews and hotel performance," *Int. J. Hospitality Manag.*, vol. 72, pp. 47–55, Jun. 2018.

[4] A. S. Lo and S. S. Yao, "What makes hotel online reviews credible?" *Int. J. Contemp. Hospitality Manag.*, vol. 31, no. 1, pp. 41–60, Jan. 2019.

[5] N. Hu, L. Liu, and J. J. Zhang, "Do online reviews affect product sales? The role of reviewer characteristics and temporal effects," *Inf. Technol. Manag.*, vol. 9, no. 3, pp. 201–214, Sep. 2008.

[6] M. Crawford, T. M. Khoshgoftaar, J. D. Prusa, A. N. Richter, and H. Al Najada, "Survey of review spam detection using machine learning techniques," *J. Big Data*, vol. 2, no. 1, pp. 1–24, Dec. 2015.

[7] A. Phaneuf. (2021). *Ecommerce Statistics: Industry Benchmarks and Growth*. [Online]. Available: https://www.insiderintelligence.com/insights/ecommerce-industry-statistics/

[8] A. Mauldin. (2021). *Global Internet Traffic and Capacity*. [Online]. Available: https://blog.telegeography.com/internet-traffic-and-capacity-return-to-their-regularly-scheduled-programming

[9] V. Muller. (2021). *How COVID-19 Changed Online Reviews*. [Online]. Available: https://www.powerreviews.com/blog/how-covid-19-changed-online-reviews/

[10] J. Alqatawna, H. Faris, K. Jaradat, M. Al-Zewairi, and O. Adwan, "Improving knowledge based spam detection methods: The effect of malicious related features in imbalance data distribution," *Int. J. Commun., Netw. Syst. Sci.*, vol. 8, no. 5, pp. 118–129, 2015.

[11] K. Dhingra and S. K. Yadav, "Spam analysis of big reviews dataset using fuzzy ranking evaluation algorithm and Hadoop," *Int. J. Mach. Learn. Cybern.*, vol. 10, no. 8, pp. 2143–2162, Aug. 2019.

[12] A. Heydari, M. A. Tavakoli, N. Salim, and Z. Heydari, "Detection of review spam: A survey," *Exp. Syst. Appl.*, vol. 42, no. 7, pp. 3634–3642, May 2015.

[13] E. Cambria and B. White, "Jumping NLP curves: A review of natural language processing research [review article]," *IEEE Comput. Intell. Mag.*, vol. 9, no. 2, pp. 48–57, May 2014.

[14] S. Hakak, M. Alazab, S. Khan, T. R. Gadekallu, P. K. R. Maddikunta, and W. Z. Khan, "An ensemble machine learning approach through effective feature extraction to classify fake news," *Future Gener. Comput. Syst.*, vol. 117, pp. 47–58, Apr. 2021.

[15] Y. Li and T. Yang, "Word embedding for understanding natural language: A survey," in *Guide to Big Data Applications*. Berlin, Germany: Springer, 2018, pp. 83–104.

[16] I. Kaibi, E. H. Nfaoui, and H. Satori, "Sentiment analysis approach based on combination of word embedding techniques," in *Embedded Systems and Artificial Intelligence*. Berlin, Germany: Springer, 2020, pp. 805–813.

[17] S. Kumar, S. Kumar, D. Kanojia, and P. Bhattacharyya, "'A Passage to India': Pre-trained word embeddings for Indian languages," in *Proc. 1st Joint Workshop Spoken Lang. Technol. Under-Resourced Lang. (SLTU) Collaboration Comput. Under-Resourced Lang. (CCURL)*. Marseille, France: European Language Resources Association, May 2020, pp. 352–357. [Online]. Available: https://aclanthology.org/2020.sltu-1.49

[18] M. M. Ashi, M. A. Siddiqui, and F. Nadeem, "Pre-trained word embeddings for Arabic aspect-based sentiment analysis of airline tweets," in *Proc. Int. Conf. Adv. Intell. Syst. Inform.* Cham, Switzerland: Springer, 2018, pp. 241–251.

[19] S. Mani, S. Kumari, A. Jain, and P. Kumar, "Spam review detection using ensemble machine learning," in *Proc. Int. Conf. Mach. Learn. Data Mining Pattern Recognit.* Cham, Switzerland: Springer, 2018, pp. 198–209.

[20] C. Visani, N. Jadeja, and M. Modi, "A study on different machine learning techniques for spam review detection," in *Proc. Int. Conf. Energy, Commun., Data Analytics Soft Comput. (ICECDS)*, Aug. 2017, pp. 676–679.

[21] Z. Tian, C. Luo, J. Qiu, X. Du, and M. Guizani, "A distributed deep learning system for web attack detection on edge devices," *IEEE Trans. Ind. Informat.*, vol. 16, no. 3, pp. 1963–1971, Mar. 2020.

[22] M. Fayaz, A. Khan, J. U. Rahman, A. Alharbi, M. I. Uddin, and B. Alouffi, "Ensemble machine learning model for classification of spam product reviews," *Complexity*, vol. 2020, pp. 1–10, Dec. 2020.

[23] A. C. Pandey and D. S. Rajpoot, "Spam review detection using spiral cuckoo search clustering method," *Evol. Intell.*, vol. 12, no. 2, pp. 147–164, Jun. 2019.

[24] S. P. Rajamohana, K. Umamaheswari, and S. V. Keerthana, "An effective hybrid cuckoo search with harmony search for review spam detection," in *Proc. 3rd Int. Conf. Adv. Electr., Electron., Inf., Commun. Bio-Informatics (AEEICB)*, Feb. 2017, pp. 524–527.

[25] A. K. Tripathi, K. Sharma, and M. Bala, "Fake review detection in big data using parallel BBO," *Int. J. Inf. Syst. Manag. Sci.*, vol. 2, no. 2, pp. 288–298, 2019.

[26] S. P. Rajamohana, K. Umamaheswari, and B. Abirami, "Adaptive binary flower pollination algorithm for feature selection in review spam detection," in *Proc. Int. Conf. Innov. Green Energy Healthcare Technol. (IGEHT)*, Mar. 2017, pp. 1–4.

[27] G. Xu, M. Hu, and C. Ma, "Secure and smart autonomous multi-robot systems for opinion spammer detection," *Inf. Sci.*, vol. 576, pp. 681–693, Oct. 2021.

[28] X. Tang, T. Qian, and Z. You, "Generating behavior features for cold-start spam review detection with adversarial learning," *Inf. Sci.*, vol. 526, pp. 274–288, Jul. 2020.

[29] M. Dong, L. Yao, X. Wang, B. Benatallah, C. Huang, and X. Ning, "Opinion fraud detection via neural autoencoder decision forest," *Pattern Recognit. Lett.*, vol. 132, pp. 21–29, Apr. 2020.

[30] M. A. Khan, M. Mittal, L. M. Goyal, and S. Roy, "A deep survey on supervised learning based human detection and activity classification methods," *Multimedia Tools Appl.*, vol. 80, no. 18, pp. 27867–27923, Jul. 2021.

[31] M. Fazzolari, F. Buccafurri, G. Lax, and M. Petrocchi, "Experience: Improving opinion spam detection by cumulative relative frequency distribution," *J. Data Inf. Qual.*, vol. 13, no. 1, pp. 1–16, Mar. 2021.

[32] S. Kennedy, N. Walsh, K. Sloka, J. Foster, and A. McCarren, "Fact or factitious? Contextualized opinion spam detection," 2020, *arXiv:2010.15296*.

[33] N. Poerner, U. Waltinger, and H. Schutze, "E-BERT: Efficient-yet-effective entity embeddings for BERT," 2019, *arXiv:1911.03681*.

[34] M. Mohring, B. Keller, R. Schmidt, M. Gutmann, and S. Dacko, "HOTFRED: A flexible hotel fake review detection system," in *Information and Communication Technologies in Tourism 2021*. Berlin, Germany: Springer, 2021, pp. 308–314.

[35] F. Anass, R. Jamal, M. A. Mahraz, Y. Ali, and H. Tairi, "Deceptive opinion spam based on deep learning," in *Proc. 4th Int. Conf. Intell. Comput. Data Sci. (ICDS)*, Oct. 2020, pp. 1–5.

[36] A. Fahfouh, J. Riffi, M. A. Mahraz, A. Yahyaouy, and H. Tairi, "PV-DAE: A hybrid model for deceptive opinion spam based on neural network architectures," *Exp. Syst. Appl.*, vol. 157, Nov. 2020, Art. no. 113517.

[37] L. Deng, J. Wei, S. Liang, Y. Wen, and X. Liao, "Review spam detection based on multi-dimensional features," in *Proc. Int. Conf. AI Mobile Services*. Cham, Switzerland: Springer, 2020, pp. 106–123.

[38] S. Saumya and J. P. Singh, "Spam review detection using LSTM autoencoder: An unsupervised approach," *Electron. Commerce Res.*, vol. 22, no. 1, p. 113, 2020.

[39] L. Zhou and Q. Zhang, "Recognition of false comments in e-commerce based on deep learning confidence network algorithm," *Inf. Syst. e-Business Manag.*, pp. 1–18, Jan. 2021.

[40] P. Bhuvaneshwari, A. N. Rao, and Y. H. Robinson, "Spam review detection using self attention based CNN and bi-directional LSTM," *Multimedia Tools Appl.*, vol. 80, pp. 18107–18124, Feb. 2021.

[41] N. Hussain, H. T. Mirza, F. Iqbal, I. Hussain, and M. Kaleem, "Detecting spam product reviews in Roman Urdu script," *Comput. J.*, vol. 64, no. 3, pp. 432–450, Mar. 2021.

[42] N. Hussain, H. T. Mirza, I. Hussain, F. Iqbal, and I. Memon, "Spam review detection using the linguistic and spammer behavioral methods," *IEEE Access*, vol. 8, pp. 53801–53816, 2020.

[43] D. P. Jayathunga, R. I. S. Ranasinghe, and R. Murugiah, "A comparative study of supervised machine learning techniques for deceptive review identification using linguistic inquiry and word count," in *Proc. Int. Conf. Comput. Intell. Inf. Syst.* Cham, Switzerland: Springer, 2021, pp. 97–105.

[44] J. Li, P. Zhang, and L. Yang, "An unsupervised approach to detect review spam using duplicates of images, videos and Chinese texts," *Comput. Speech Lang.*, vol. 68, Jul. 2021, Art. no. 101186.

[45] S. Ansari and S. Gupta, "Customer perception of the deceptiveness of online product reviews: A speech act theory perspective," *Int. J. Inf. Manag.*, vol. 57, Apr. 2021, Art. no. 102286.

[46] A. M. Evans, O. Stavrova, and H. Rosenbusch, "Expressions of doubt and trust in online user reviews," *Comput. Hum. Behav.*, vol. 114, Jan. 2021, Art. no. 106556.

[47] Y. W. Oh and C. H. Park, "Machine cleaning of online opinion spam: Developing a machine-learning algorithm for detecting deceptive comments," *Amer. Behav. Scientist*, vol. 65, no. 2, pp. 389–403, Feb. 2021.

[48] A. K. Tripathi, K. Sharma, and M. Bala, "Military dog based optimizer and its application to fake review," 2019, *arXiv:1909.11890*.

[49] M. S. Rani and S. Sumathy, "Online social networking services and spam detection approaches in opinion mining—A review," *Int. J. Web Based Communities*, vol. 14, no. 4, pp. 353–378, 2018.

[50] S. P. Rajamohana and K. Umamaheswari, "Hybrid optimization algorithm of improved binary particle swarm optimization (iBPSO) and cuckoo search for review spam detection," in *Proc. 9th Int. Conf. Mach. Learn. Comput.*, Feb. 2017, pp. 238–242.

[51] P. Hajek, A. Barushka, and M. Munk, "Fake consumer review detection using deep neural networks integrating word embeddings and emotion mining," *Neural Comput. Appl.*, vol. 32, no. 23, pp. 17259–17274, Dec. 2020.

[52] A. Barushka and P. Hajek, "Review spam detection using word embeddings and deep neural networks," in *Proc. IFIP Int. Conf. Artif. Intell. Appl. Innov.* Cham, Switzerland: Springer, 2019, pp. 340–350.

[53] G. M. Shahariar, S. Biswas, F. Omar, F. M. Shah, and S. B. Hassan, "Spam review detection using deep learning," in *Proc. IEEE 10th Annu. Inf. Technol., Electron. Mobile Commun. Conf. (IEMCON)*, Oct. 2019, pp. 0027–0033.

[54] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, Jul. 1995.

[55] V. N. Vapnik, "An overview of statistical learning theory," *IEEE Trans. Neural Netw.*, vol. 10, no. 5, pp. 988–999, 1999.

[56] V. Vapnik, *The Nature of Statistical Learning Theory*. Berlin, Germany: Springer, 2013.

[57] G. James, D. Witten, T. Hastie, and R. Tibshirani, *An Introduction to Statistical Learning*, vol. 6. Berlin, Germany: Springer, 2013.

[58] A. A. Heidari, S. Mirjalili, H. Faris, I. Aljarah, M. Mafarja, and H. Chen, "Harris hawks optimization: Algorithm and applications," *Future Gener. Comput. Syst.*, vol. 97, pp. 849–872, Aug. 2019.

[59] S. Lai, K. Liu, S. He, and J. Zhao, "How to generate a good word embedding," *IEEE Intell. Syst.*, vol. 31, no. 6, pp. 5–14, Nov. 2016.

[60] F. Zhuang, Z. Qi, K. Duan, D. Xi, Y. Zhu, H. Zhu, H. Xiong, and Q. He, "A comprehensive survey on transfer learning," *Proc. IEEE*, vol. 109, no. 1, pp. 43–76, Jan. 2021.

[61] W. Xiao, J. Zhang, Y. Li, S. Zhang, and W. Yang, "Class-specific cost regulation extreme learning machine for imbalanced classification," *Neurocomputing*, vol. 261, pp. 70–82, Oct. 2017.

[62] A. G. Jivani, "A comparative study of stemming algorithms," *Int. J. Comput. Tech. Appl.*, vol. 2, no. 6, pp. 1930–1938, 2011.
[63] M. F. Porter. (2001). *Snowball: A Language for Stemming Algorithms*. [Online]. Available: http://snowball.tartarus.org/texts/introduction.html
[64] L. S. Larkey, L. Ballesteros, and M. E. Connell, "Light stemming for Arabic information retrieval," in *Arabic Computational Morphology*. Dordrecht, The Netherlands: Springer, 2007, pp. 221–243.
[65] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," 2018, *arXiv:1810.04805*.
[66] G. Xu, J. Zhang, U. G. O. Cliff, and C. Ma, "An efficient blockchain-based privacy-preserving scheme with attribute and homomorphic encryption," *Int. J. Intell. Syst.*, vol. 37, no. 12, pp. 10715–10750, Dec. 2022.

**ALA' M. AL-ZOUBI** received the B.Sc. degree in software engineering from Al-Zaytoonah University, in 2014, and the M.Sc. degree in web intelligence from The University of Jordan, Jordan, in 2017. He is currently pursuing the Ph.D. degree with the School of Science, Technology, and Engineering, University of Granada. He was a Teacher and a Research Assistant on various projects funded by The University of Jordan. During the graduate studies, he has published numerous papers in reputable journals and conferences. His research interests include evolutionary computation, machine learning, and security in social network analysis, among other research fields. He is an active member of two research groups, such as the JISDF Research Group, which focuses on bridging the gap between academia and industry in the field of security, and the Evo-ML Research Group, which explores evolutionary algorithms, machine learning, and their applications for solving significant problems in diverse domains.

**ANTONIO M. MORA** was born in Granada, Spain, in 1977. He received the degree in computer sciences and the Ph.D. degree in researching on multi-objective ant colony optimization for solving a bi-criteria path finding problem in a military environment from the University of Granada, in 2001 and 2009, respectively. He is currently an Associate Professor with the Department of Signal Theory, Telematics and Communications, University of Granada, where he was a Contract Researcher and a Substitute Professor for 14 years. He has published more than 25 articles in international journals and more than 100 papers in top-rated international conferences. His research interests include bio-inspired algorithms and their applications to data analysis, network security, and video games. He is a Founder Member of the Spanish Society on Videogames Research (SECiVi). He has been an active member of review boards in IEEE journals, such as IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION and IEEE TRANSACTIONS ON COMPUTATIONAL INTELLIGENCE AND AI IN GAMES, and a PC and an Organization Member of IEEE conferences, such as IEEE Conference on Computational Intelligence and Games and IEEE Evolutionary Computation Conference.

**HOSSAM FARIS** received the B.A. degree in computer science from Yarmouk University, Jordan, in 2004, the M.Sc. degree in computer science from Al-Balqa' Applied University, Jordan, in 2008, and the Ph.D. degree in e-business from the University of Salento, Italy, in 2011. In 2016, he worked as a Postdoctoral Researcher with the GeNeura Team, Information and Communication Technologies Research Center (CITIC), University of Granada, Spain. He is currently a Professor with the Department of Information Technology, King Abdullah II School for Information Technology, The University of Jordan, Jordan. He is also the Chief Data Science Officer with Altibbi. He co-founded The Evolutionary and Machine Learning (Evo-ML.com) Research Group. His research interests include applied computational intelligence, evolutionary computation, knowledge systems, and data mining.