Spam Reviews Detection Models in Multilingual Contexts applying Sentiment Analysis, Metaheuristics, and Advanced Word Embedding

UNIVERSIDAD DE GRANADA

ALA' MAHMOUD MOHAMMED AL ZOUBI

Supervisors: ANTONIO MIGUEL MORA GARCÍA
HOSSAM FARIS

To obtain the Ph.D. degree as part of the
El Programa de Doctorado en Tecnologías de la Información y la Comunicación

February 2024

Editor: Universidad de Granada. Tesis Doctorales
Autor: Ala' Mahmoud Mohammed Al Zoubi
ISBN: 978-84-1195-276-7
URI: https://hdl.handle.net/10481/91051

I would like to dedicate this thesis to my loving parents ...

Declaration

I hereby declare that except where specific reference is made to the work of others, the contents of this dissertation are original and have not been submitted in whole or in part for consideration for any other degree or qualification in this, or any other university. This dissertation is my own work and contains nothing which is the outcome of work done in collaboration with others, except as specified in the text and Acknowledgements. This dissertation contains almost 61,000 words including, the main text, footnotes, tables, equations, figures, and bibliography.

ALA' MAHMOUD MOHAMMED AL ZOUBI

February 2024

Acknowledgements

I would like to express my sincere gratitude to all those who have supported me during the course of this research. First, I would like to express my deepest appreciation to my two advisors, Dr. Antonio Miguel Mora García and Prof. Hossam Faris, for their guidance, support, and encouragement throughout the entire research process. Their expertise, patience, and unwavering support have been invaluable in helping me complete this thesis. They have been instrumental in shaping my research and have provided me with invaluable feedback and suggestions that greatly improved the quality of this thesis. Furthermore, I would like to extend my heartfelt gratitude to my family, for their unwavering support, encouragement, and understanding throughout the entire research process. I dedicate this thesis to my parents, who has been a constant source of inspiration, motivation, and love.

This work would not have been possible without the support and encouragement of all of the above-mentioned individuals. I am truly grateful to have had the opportunity to work with such dedicated and supportive individuals.

Abstract

Online reviews are a type of information that comes from consumers' experiences after using products or services. Many individuals and organizations consider these reviews as an evaluation source that can be used for their sake. For example, customers with the desire to purchase read reviews in order to know the quality of products and services, while organizations view them as feedback about their products. As a result, reviews can easily impact businesses' reputations either in a positive or negative way. The COVID-19 pandemic led to a marked increase in the number of online reviews, with a 50% uptick observed between 2020 and 2023. This can be attributed to the widespread adoption of remote technology and increased time spent at home, as compared to pre-pandemic levels. In addition to the increase in numbers, the COVID-19 pandemic also led to changes in the style, structure, context, and preferences of online reviews. However, not all reviews are genuine, some of them are written for the purpose of misleading consumers to make a profit or damage competitors. Such reviews are known as *spam reviews*, and they can lead to a lot of harm, including financial losses and job loss. Therefore, it is a necessity to identify these fraudulent reviews. Spam reviews detection is an important aspect of online security as it aims to mitigate the potential negative effects of such reviews, including manipulation of online reputation and consumer fraud. Additionally, it helps to protect the integrity of online marketplaces and prevents the promotion of malicious sites through spam reviews. One of the well-recognized methods used to detect spam reviews is machine learning, specifically through the application of Natural Language Processing (NLP). Machine learning algorithms, incorporating NLP techniques, prove to be efficient methods for text classification across various tasks, such as sentiment analysis and spam reviews detection. In this thesis, three different approaches are presented in order to identify spam and the sentiment of reviews.

The first applied approach is the Weighted Support Vector Machine (WSVM) and a swarm intelligence algorithm supported by the prey and predator behaviors of soft and hard besiege for spam review detection, where the optimization algorithm was incorporated for feature weighting and hyperparameter optimization simultaneously. The combination of SVM and metaheuristic algorithms showed excellent performance in the literature for solving different problems. To enhance their performance, a weighted method has been implemented

in conjunction with a metaheuristic algorithm. The first approach is applied to test the performance of the proposed WSVM in spam reviews detection and compare it with other existing approaches.

Furthermore, the second approach employed Twin SVM (TwinSVM) and a swarm intelligence algorithm based on the special chain movement method to address a different problem, specifically in analyzing the sentiment of reviews. Due to its superior performance compared to traditional SVM, particularly in mitigating overfitting, improving generalization, handling imbalanced datasets, enhancing robustness, and adapting to complex structures, the TwinSVM model has been employed. Moreover, the swarm-based algorithm has been utilized in this approach for the parameter optimization of TwinSVM. TwinSVM is a newer version of the standard SVM, which offers several advantages compared to SVM and incorporates additional parameters. Further, we used the second approach for sentiment prediction of the same reviews and then compared it with the previous approach (WSVM) and other approaches.

In the final approach, we amalgamated the findings from the previous two approaches to pursue the optimal outcome. That is to say, sentiment analysis was applied to improve spam review detection based on the combination of the prey and predator behaviors of the swarm-based algorithm and TwinSVM. Here, the metaheuristic algorithm is applied to optimize TwinSVM parameters and feature selection. In other words, in this third approach, we combine the previous two applications (spam reviews detection and sentiment analysis) with the optimized version of SVM (TwinSVM) and feature selection technique to enhance the spam detection performance.

Given that there are reviews in several languages, three different languages have been taken into account when extracting reviews, namely English, Spanish, and Arabic. In order to convert text into data that can be interpreted by machine learning algorithms, word representation methods have been applied. Due to their ability to deal with multilingual text, three pre-trained word embedding models were applied, namely, Bidirectional Encoder Representations from Transformers (BERT), FastText, and Multilingual Unsupervised and Supervised Embeddings (MUSE). Consequently, twenty-four datasets were generated, each comprising three languages, and a multilingual dataset was created by combining these three languages with two different dimensions (rows) (100 & 400).

The experimental results of all approaches demonstrate the superiority of the proposed methods compared with other algorithms from the State of the Art. Specifically, the prey and predator-based algorithm with WSVM obtained the best results (Accuracy) in the first approach, with 88%, 71%, 89%, and 84%, for English, Spanish, Arabic, and Multilingual datasets, respectively. As for the second approach, the special chain movement algorithm

combined with TwinSVM outperformed the other algorithms on most datasets. More precisely, it has the best results in three out of the main four datasets with 87%, 86%, and 87% for Arabic, English, and Multilingual datasets, respectively. In the final approach, the prey and predator algorithm with TwinSVM-Fs achieved the highest accuracy when compared with other algorithms, with 92%, 89%, 80%, and 85% for Arabic, English, Spanish, and Multilingual datasets, respectively. Further, with the help of sentiment analysis, the results improved by 1.0994%, 2.6674%, 9.4430%, and 8.7448% for Arabic BERT-100, English MUSE-400, Spanish Fast-400, and Multi MUSE-400, respectively. At the end of each approach, a comprehensive analysis of the review's style, context, and preferences was discussed.

Overall, the thesis proposes three effective approaches to detect spam reviews, study the sentiment of the reviews, and ultimately, improve the detection of spam reviews using sentiment analysis, all within a multilingual environment.

Table of contents

List of figures

List of tables

Acronyms

ADASYN Adaptive Synthetic Sampling

AFSA Artificial Fish Swarm Algorithm

BA-3+ Ternary Bees Algorithms

BERT Bidirectional Encoder Representations from Transformers

BFPA Adaptive Binary Flower Pollination Algorithm

BPSO Binary PSO

CBoW Continuous Bag Of Words

CMA Competition and Markets Authority

CNN Convolutional Neural Network

CRFD Cumulative Relative Frequency Distribution

CRO Chemical Reaction Optimization

CS Cuckoo search

DE Differential Evolution

DT Decision Tree

eWOM Electronic word-of-mouth

FOA Fruit Fly Optimization

GA Genetic Algorithm

GAN Generative Adversarial Network

GEPSVM Generalized Eigenvalues PSVM

GRNN (Gated Recurrent Unit

GSA Gravitational Search Algorithm

GWO grey wolf optimizer

HHO Harris Hawks Optimization

iBPSO optimized BPSO

IWV Improved Word Vectors

K* KStar

k-NN k-nearest neighbors

LIWC f Linguistic Inquiry Word Count

LR , Logistic Regression

LSTM Long Short-term Memory

LSTwinSVM Least Squares TwinSVM

mBERT Multilingual BERT

MLP Multilayer perceptron

MUSE Multilingual Unsupervised and Supervised Embeddings

NB Naïve Bayes

NLP Natural Language Processing

PCA Principal Component Analysis

POS Part-of-Speech

PSO Particle Swarm Optimization

PSVM Proximal SVM

RC Ridge Classifier

RF Random Forest

RNN Recurrent Neural Network

RSS Ring Seal Search

SGD Stochastic Gradient Descent

SLR Systematic Literature Review

SMOTE Synthetic Minority Oversampling Technique

SSA Salp Swarm Algorithm

SVM Support Vector Machine

TwinSVM Twin Support Vector Machine

UGC User-Generated Content

UGRNN Update GRNN

W2WGs Word-to-Word Graphs

WOA Whale Optimization Algorithm

WSVM Weighted Support Vector Machine

WTWSVM Wavelet Analysis Techniques TwinSVM

XGBoost eXtreme Gradient Boosting

Chapter 1

Introduction

1.1 Introduction

Nowadays, people use information for different aspects of their daily life such as learning, updating, informing, and instructing purposes. Basically, it is a piece of knowledge or facts describing a specific event or subject. Information can be acquired through various means, namely, through websites, books, personal conversations, and experiences. Similarly, there are multiple ways in which it can be shared, including writing, speaking, or the media. One example of such information is online reviews, which are a type of feedback written by consumers about their experience with products or services. Furthermore, the authors of [208] state that reviews provide a subjective assessment of a service, product, or experience. They are generally written from the reviewer's perspective and based on his/her own personal experiences. As stated in [14], reviews are a type of word-of-mouth that includes personal recommendations. So, a review constitutes a critical evaluation of a product by exploring its functionality, aesthetics, and tactile properties. The assessment focuses on various significant aspects such as the product's quality, worth, dimensions, longevity, and other pertinent features.

According to Consumer Centers in Europe, more than 80% of consumers check online reviews before purchasing a product [280]. Hence, reviews are considered significant for many individuals, as they can provide consumers with valuable information regarding product or service quality and performance. In this way, they are able to make better purchasing decisions. Furthermore, reviews can play a critical role in influencing consumer purchasing decisions, as they can help establish trust in a product or service. Additionally, businesses can benefit from online reviews by gaining valuable insight and feedback. This can assist them in improving their services and better meeting their customers' needs.

On the one hand, positive reviews can increase business profits and are likely to lead to enhanced customer loyalty. On the other hand, negative reviews, for example, may damage their reputation. People can also lose their jobs and money due to negative reviews. In this sense, it can be concluded that reviews have a significant impact on a wide range of issues.

Reviews can be found in several places such as online shopping websites, movie reviews, food services, and others. Some examples of these sites include TripAdvisor, Metacritic, Yelp, IMDB, Just-Eat, and Amazon [303]. However, with the rise of social networks, they have become the largest community with written reviews and they give their users the ability to post whenever and whatever they want. Social networks like Facebook and Twitter are among the most influential and effective places on the Internet, due to their web-based platform features that are user-friendly and convenient [15].

As technology, such as social networks, has evolved, it has brought people from diverse backgrounds together. The internet has also made information more accessible to everyone, breaking down national boundaries. In this globalized society, it is important that our representations are unbiased and inclusive, including language. As a result, multilingualism is becoming more relevant. Multilingual reviews are becoming increasingly important as they allow for a more inclusive representation of different cultures and languages. They also enable others to understand the nuances of language and how it affects the interpretation of information. This is particularly important in fields such as NLP and machine learning, where language plays a crucial role. Additionally, multilingual reviews also provide valuable insights into the cultural context of the information being reviewed. This is especially important in the field of marketing, where understanding the cultural context of a product or message can greatly impact its global effectiveness. Overall, the increasing importance of multilingual reviews is a reflection of the globalized society we live in and the need for inclusive and unbiased representation. It highlights the importance of understanding the role of language in shaping our perceptions and interactions with the world around us.

The recent years have seen a change in online reviews due to COVID-19, including an increase in their quantity and a shift in their preferences for the customers. In general, consumers now tend to focus more on medical and entertainment products. For instance, over 54% of consumers stated to give skincare a higher priority than makeup during the pandemic [196]. According to a study by PowerReviews, review interactions have increased by 50% compared to pre-pandemic and 89% in the first months. As a result, 41% of beauty shoppers rely more on reviews and ratings than before the pandemic [196].

1.2 Motivation

In addition to their impact on our lives, the massive growth in review interactions also causes an increase in spam reviews. A spam review can be seen as a fraud, a fake, or a misleading review that is meant to trick consumers. The authors in [171], suggested that spam reviews are opinions that are often deceptive or false, and which are published online in order to manipulate the perception or decisions of consumers.

These consumers may be misled by spam reviews and their purchasing decisions are likely to be affected as a result. This contributes to unfair competition and biased market benefits for organizations that use spam practices. Further, it could cause reputational damage to a business or its products. Also, by using spam reviews it is possible to reduce the credibility of all online reviews, thus, consumers may be unable to trust the review system and reviews will become less valuable.

Search engine rankings can also be affected by spam reviews, when determining the relevance of the search, search engines may take into account the quantity and quality of reviews. Organizations that engage in spam review practices may be liable for legal action, for example, fines or lawsuits. According to The Washington Post report, approximately 61% of Amazon reviews in electronic categories were determined to be fraudulent [75]. Further, an investigation conducted by the UK consumer group 'Which?' found that about one in seven TripAdvisor reviews are fraudulent [41]. Approximately £23 billion of UK customer spending is potentially influenced by fake reviews every year, as it is estimated by the Competition and Markets Authority (CMA) [55]. Moreover, the YELP website has also experienced an increase in spam reviews from 5 to 20 percent, as reported in [181].

The field of Computer Science, specifically Artificial Intelligence (AI), holds a crucial position in addressing the challenge of spam reviews. Such reviews have become a widespread issue across numerous online platforms. AI domain is instrumental in combating this issue using NLP, Machine Learning Algorithms, User Behavior Analysis, Image and Multimedia Analysis, Content Matching, and Duplicate Detection.

Therefore, it is imperative that the issue of fake or spam reviews detection will be addressed. Researchers proposed various types of detection approaches to minimize the impact of such malicious activity [82]. To identify spam reviews, content-based detection is becoming increasingly popular in the literature. Especially through using machine learning algorithms for text classification. Text classification is a key component of NLP, which involves categorizing text documents based on predefined characteristics [151]. Text classification can be used for a variety of purposes, including sentiment analysis, spam filtering, and topic modeling. Additionally, it can be used to organize and filter large amounts of text information [186].

As a branch of NLP, sentiment analysis is responsible for designing, implementing, and evaluating methods, techniques, models, and determining whether a text consists of objective or subjective information [141]. Then determine whether that information is expressed positively, neutrally, or negatively, as well as whether it is strongly or weakly expressed.

There have been successful applications of sentiment analysis techniques in detection studies. For example, the ability to detect irony, satire, and sarcasm in formal and informal writing [242, 306]; in detecting instances of hate speech, racism, harassment, bullying and sexism [20, 166, 127, 6, 65]; in reputation and influence analysis [284, 28]; in the analysis of health and well-being [313, 9] and a part of security monitoring [261].

Sentiment analysis is also considered one of the innovative approaches for improving spam detection [53, 137]. Detecting spam reviews through sentiment analysis is a technique for identifying fake or fraudulent reviews on online platforms, such as e-commerce websites, social media, and review sites. The basic idea of detecting spam reviews through sentiment analysis is to use the sentiment, or overall attitude, expressed in a review as a signal of its authenticity. For instance, spam reviews can manipulate the ratings or rankings of products by writing reviews with positive or neutral content. In contrast, genuine reviews tend to express a mix of negative, neutral, and positive sentiments, depending on each reviewer's particular experience.

Expression of sentiment plays an integral role in spam reviews. Using a combination of sentiment indicators, Dickerson et al. [68] suggests that sentiment-related behavior is adequate to distinguish between social bots and human accounts. Thus, sentiment analysis is able to provide critical information about the content of reviews in order to determine whether they are trustworthy or should be regarded as spam. Additionally, the authors of [260] explained that sentiment analysis was found to be useful in detecting spam reviews, since positive sentiments in fake reviews tend to be exaggerated, whereas negative sentiments in social media replies tend to be strongly expressed. Therefore, they believe sentiment analysis is an effective approach for illustrating the emotions, opinions, and attitudes displayed on online social media, as well as identifying suspicious accounts based on sentiment-related factors. On the one hand, a number of spam review detection approaches use sentiment analysis as their fundamental foundation [202]. In addition, other information is typically gathered from reviewer content as well as from review metadata. On the other hand, in a number of models, sentiment is considered a feature of the review item.

Aside from sentiment analysis features, textual features also require more consideration to extract, especially when the multilingual context is taken into account. A variety of methods can be used to convert text into features. Despite this, word embedding is considered the preferred technique when dealing with extracting multilingual features [58]. By representing

words from different languages in a single unified vector space, word embeddings are a powerful method for working with multilingual data. With this method, it is possible to deal with words from various languages within the same model, rather than training separate models for each language.

The main idea behind word embedding is the conversion of words into real numbers by mapping them to related vectors. The use of similar vectors can allow the representation of words with the same meaning, where the cosine similarity between vectors can be used to identify similar words. In addition, word embedding models can be learned from huge amounts of data and then used for various tasks later on. Such embeddings are known as pre-trained word embeddings. With the benefit of pre-trained models, several shortcomings can be resolved that are related to word embedding such as starting from scratch and learning from small datasets. To put it another way, it can save time and resources as well as increase the efficiency of subsequent tasks, including text classification and language translation.

A variety of machine learning algorithms are used in the field of text classification, such as Random Forest (RF), k-nearest neighbors (k-NN), Naïve Bayes, Neural Networks, and SVM. For text classification and, in particular, for spam review detection using sentiment analysis, SVM is one of the most effective machine learning techniques. For example, Eranpurwala et al. [78, 79] presented two studies based on a sentiment analysis approach for spam review detection using SVM. According to both studies, SVM outperforms the other algorithms when it comes to detecting spam reviews.

However, when using SVM, tuning its parameters is essential to achieving effective performance. To solve this problem, metaheuristic algorithms can be employed. Metaheuristic algorithms are high-level methods for finding the best possible solution. A variety of optimization problems can be solved using these approximate algorithms, such as parameter optimization and feature selection. Through metaheuristic algorithms, SVM can be improved by determining optimal values for its hyperparameters [85, 12]. The text classification problem has been studied by many researchers using metaheuristic algorithms and SVM. For instance, in order to perform textual data classification, Particle Swarm Optimization (PSO) is employed to optimize the parameters of SVM [150]. Moreover, in order to efficiently classify text, the Ring Seal Search (RSS) algorithm was combined with an SVM [259]. Further, a hybrid method combining SVM and PSO was applied for text classification task [149].

Though SVM with metaheuristic algorithms provided excellent performance, it also exhibited several limitations under certain conditions. To overcome these limitations, Twin Support Vector Machines (TwinSVM) have been applied. TwinSVM is considered an extension of the standard SVM algorithm used for classification tasks. A "twin" structure can be constructed using two hyperplanes, offering a more precise representation of the data.

The majority is separated from the minority by the first hyperplane, whereas the minority is separated from the rest by the second hyperplane. Moreover, compared to standard SVM algorithms, it offers several advantages. For instance, TwinSVMs are better at handling data with imbalance problems than standard SVMs. TwinSVM also provides enhanced functionality when employing different kernel functions. Therefore, TwinSVM is capable of adapting to a wide range of kernel functions, while it's more sensitive to standard SVM when it comes to selecting between kernel functions.

In this thesis, three efficient approaches have been proposed to deal with spam and sentiment reviews identification in a multilingual context. The first approach consists of using WSVM and swarm intelligence algorithm based on the prey and predator behaviors to detect spam reviews. In this approach, the swarm-based algorithm is used to optimize the SVM parameters and apply feature weighting. Second, the TwinSVM and swarm intelligence algorithm based on the special chain movement method were employed for sentiment analysis. Here, the optimization algorithm was responsible for optimizing the TwinSVM parameters. In the third approach, we combine prey and predator swarm algorithm and TwinSVM to improve spam reviews detection through sentiment analysis. During this phase, the metaheuristic algorithm optimized the hyperparameters and selected features to enhance the performance. To deal with the multilingual context that reviews could face, three different languages have been considered in all approaches, namely, English, Spanish, and Arabic. In the first approach, traditional word representation and BERT word embedding have been utilized in order to prove the superiority of word embedding when handling multilingual contexts. As for the other two approaches, various pre-trained word embedding models (BERT, FastText, and MUSE) were used for word representation. These pre-trained models were employed due to their capability to deal with multiple language datasets.

1.3 Objectives

The main goal of this thesis, is to propose an approach to perform efficient spam reviews detection and then demonstrate its value and validity. Throughout this thesis, various objectives are presented to deal with spam and sentiment reviews identification and ultimately improve their performance. These main objectives include:

- Design an approach for efficient spam review detection.
- Conduct and improve the performance of sentiment analysis for the same reviews.
- Utilize sentiment analysis to improve spam review detection.

In order to achieve these objectives, several steps must be completed and can be summarized as follows:

- Several approaches will be proposed to deal with spam and sentiment review identification.
- Swarm-based algorithms will be integrated with machine learning for hyperparameters optimization as well as feature weighted and selection to improve their performance.
- Multilingual context data issues will be addressed by extracting reviews in different languages, including English, Spanish, and Arabic.
- Several pre-trained word embedding models will be employed for word representation in a multilingual environment, namely, BERT, FastText, and MUSE.
- Deep analysis of the reviews' style, context, and product preference will be studied before and after the COVID-19 pandemic.

1.4 Thesis Structure

This thesis is structured as follows: Chapter two gives detailed information on the context and background of the methods used in this study. The third chapter, Literature Review, discusses previous research on spam review detection, sentiment analysis, spam review detection through sentiment analysis, and Twin SVM works including previous research studies, theories, and models. The fourth chapter describes the dataset used in the study, including the preparation and labeling criteria, as well as the datasets used in the literature, multilingual motivations, and word representation methods. The fifth chapter presents the first approach employed in the study for detecting spam reviews using Weighted Swarm Support Vector Machines. The sixth chapter describes the second approach used in the study for sentiment analysis using a swarm-based model and Twin-SVM. The seventh chapter outlines the third approach applied in the study for detecting spam reviews by using sentiment analysis and feature selection based on Twin SVM. The eighth chapter, Conclusions, and Future Work, summarizes the study's main findings, reiterates the research problem and objectives, and provides recommendations for future research. It also concludes the thesis and highlights the contributions made by the research.

Chapter 2

Background

This chapter covers background information and definitions of the techniques and methods applied in this thesis. It will be divided into two parts: the first part will define terms such as online reviews, spam reviews, detection, and sentiment analysis. The second part will discuss classification models and metaheuristic algorithms.

2.1 Preliminaries

2.1.1 Online Reviews

A review can be defined as feedback of specific products from specific users to inform particular individuals. De Pelsmacker et al. [62] described online reviews as Electronic word-of-mouth (eWOM). The eWOM is informal communications about usage or features of goods, services, or their sellers geared towards consumers through an internet-based technology. On the other hand, [172] stated that reviews are popular User-Generated Content (UGC) that consumers utilize when making different decisions related to travel, buying, renting, and using a service. Reading such reviews can reduce risk, make new ideas, streamline choices, and emphasize choices for consumers. While the work in [117], interprets online reviews as a process that previous consumers provide to criticize products.

That being said, reviews are known as an information source for marketers and consumers in order to learn product quality. Also, reviews are considered a useful metric to measure the loyalty of consumers in situations like product's success critical implications [40]. Therefore, providing reviews can benefit many related parties to study the products in different aspects to confirm the quality, sales, reputation, and market share of the product. Others consider reviews more credible and essential than the data presented by commercial sources [157].

Moreover, reviews could affect several facets of products such as revenue, prices, performance, and popularity. According to European Consumer Centres, more than the 80% of consumers read reviews before purchasing [280]. Online reviews have a primary and critical impact on e-commerce, particularly shopping decisions and the amount of money that could be spent. For that reason, reviews are known to be one of the important aspects of business performance [299].

Reviews usually have a heavy effect on consumers buying behavior in various product categories, including, games [311], movies [74], restaurants [146], and books [52]. Pelsmacker et al. [62] states that alongside reviews, also rating could impact consumers' attitudes. A number of studies take into account the rating as an element of reviews that also have some kind of effect on consumers [277, 37]. Therefore, [74] demonstrates that consumers now have access to any information and exchange opinions on products, services, and companies easily.

As a result, more businesses begin to offer online services for consumers to write their reviews. Examples of such services are Amazon, Yelp, IMDb, Metacritic, and so on. Further, ABC, CBS, and NBC television networks provided a place for viewers to discuss their shows and programs online [74]. Hence, making the web more like a medium to reach people in order to generate more awareness of their products and services. Also, with the different types of reviewers' argument quality, readers could easily differentiate with certain reviews. Thus, many websites offer the ability for readers to rate reviews to be useful or not for them [51]. All these features made reviews more crucial every day.

In summary, online reviews become more needed than ever due to different aspects, such as knowing the quality of the products, and making the right decisions about what to buy. Moreover, feedback for companies, manufacturers, and owners, as well as staying at home due to the COVID-19 pandemic situation (quarantine). These aspects or reasons increased the importance and popularity of online reviews. Therefore, the number of platforms that support online reviews increased. However, due to the widespread and simplicity of users who post and read reviews, it becomes more exposed to turn these reviews into negative opinions, namely, *spam reviews*. Furthermore, in light of the COVID-19 circumstances, individuals predominantly concentrate on the pandemic rather than other matters. Where most posts and tweets were about the feeling of the circumstances. Such behavior allows researchers to understand people's mental stress and illness, and then produces different types of studies such as sentiment analysis. Thus, helping people from panicking or deterioration of their mental state that could cause more increase in the number of bad decisions [266, 254].

2.1.2 Spam Reviews

With the huge achievement of reviews recently due to its important and great features for many people. This success attracts a large number of individuals (competitive parties) and their intention to spread false information and reviews for different objectives, including ruining or improving the product's reputation. This type of review is known as fake or spam review. Martens and Maalej [178] described spam reviews as a procedure done by fake reviewers who get paid for submitting such reviews. These reviews might or might not be actual users of the product. While, [116], stated that spam reviews are an act from writers, publishers, and vendors that post non-authentic online reviews to increase product sales. Further, Banerjee and Chua explained that spam reviews in tourism, for example, are similar to online reviews written by the imagination of several persons without the experience of going to that destination [29]. As for [293], they defined a spam review as a review that is inconsistent with the genuine evaluations of services or products. Therefore, spam reviews are considered deceptive, false, and bogus reviews could be posted by various types of individuals, namely, review platforms online merchants, and consumers.

According to [174, 251, 198, 245], the percentage of spam reviews ranges from 16%, 20%, and 25% to 33.3%, respectively. There are many situations where the spam reviews had a role in causing damage to several parties. For instance, the UK Advertising Standards Authority reported that the TripAdvisor website was involved in generating more than 50 million spam reviews in 2012 [237]. While, in 2013, the Taiwan Federal Trade Commission ordered Samsung to pay a fine for spreading negative spam reviews [33]. Moreover, Amazon sued more than 1000 reviewers for posting spam reviews [95]. Mafengwo's website for a tourism platform in China was also involved in review deception [309].

Spam reviews have the ability to manipulate the market effectively. This is done due to being complex in structure and similar to real ones. However, spam reviews tend to be more influential. Various platforms try to manipulate and add spam reviews in order to boost traffic and deceive consumers into getting involved in the argument [158]. Therefore, spam reviews become more challenging to identify day after day. Spammers have been employed novel techniques in order to increase the difficulty of preventing and detecting their spam reviews. Such techniques can be, for example, using a different style of writing periodically, duplicating real users' reviews on other products, and implementing complex bots that can spread spam reviews.

Both academia and industry tried several countermeasures to mitigate the spam reviews phenomenon. Also, many governments imposed a number of laws and penalized anyone who performed such an act. In 2013, for example, the Attorney General of New York State led the operation "Clean Turf" for identifying companies that generate and post spam

reviews [292]. In 2018, the Chinese government legislated the first E-commerce Law that prevents merchants from performing misleading and false promotions through posing spam reviews [54]. On the other hand, researchers proposed numerous attempts to detect spam reviews [262, 249, 135, 27]. However, these attempts did not have much impact on decreasing the spread of all spam reviews. They offer different methods that can detect an exact type of spam review, notwithstanding, it's hard to identify all types. Hence, more approaches have been proposed periodically to handle new and specific types of spam reviews.

2.1.3 Spam Reviews Detection

Detection can be described as a process to prevent and identify something that presents and further perform its purpose either being positive or negative [104, 8, 16, 56, 91, 92]. Klein et al. [148] states that detection is a procedure that required action after the individuals in control become concerned about an event that might cause undesirable and unexpected activity. Whereas, Cowan explained the detection as an accumulation of several discrepancies that reached the threshold [60]. Furthermore, detection in the security fields is more critical than detection in other fields. For example, the authors in [231] described security threat detection as a practice of examining the security ecosystem in order to distinguish any malicious activities. The main principle of the detection process relies on the technology perception to discover unique patterns among all behaviors [268]. Besides, the intention of security detection is to implement automatic systems that are capable of investigating the possibility of different kinds of threats [200].

The detection process might help us with an early warning of the possible conditions that can occur. Thus, when we need a countermeasure for an exact incident, there will be time to construct a plan and execute the steps of that plan. For example, when a person is involved in a routine activity like driving, he needs to observe the disturbances that could imply traffic jams or dangerous situations. Even in a stable circumstance (a tree branch could fall on a car or a maintenance procedure that might impact the safety of a plant operation). An alert of possible risk is desired in order to take the correct action to prevent the loss of important resources. As soon as the problem is detected, various procedures can be applied, including, collecting more information, monitoring the events closely, defining the problem, or discussing the situation with others.

Detection can be found in all fields, especially in security, such as malware, anomaly, intrusion, and spam detection. Malware detection, for instance, refers to the procedure of finding malicious software (e.g., viruses, spyware, and ransomware) on a system and trying to perform extensive damage or steal information of that system [138]. While intrusion detection is consists of a software application that attempts to monitor the system or network

for any policy violations or malicious activity [222]. In addition to that, anomaly detection is a process of recognizing the outlier or difference of normal activity, usually in a network. As for spam detection, it has more variants than the other detection methods. Spam can be in emails, social network profiles and posts, messages, or, reviews.

Spam reviews are characterized to be complicated due to their structure. Therefore, complex problems require complex solutions. In the literature, many detection methods for spam reviews have been presented. [262] introduced a network-based spam detection model for online reviews in the social network environment, while [101] study the detection of spam reviews using the reviews processing and the rating of those reviews. Deep learning also has been used to identify the spam reviews through [253], and the authors of [170] investigated the spam detection for reviews by integrating the probabilistic review graph with the multi-modal embedded representation. Such methods are two years old and the style of spam reviews has been improved. Consequently, new and enhanced approaches become imperative.

2.1.4 Sentiment Analysis

Sentiment analysis involves analyzing written text with the aim of identifying and categorizing opinions expressed therein; these opinions are generally classified as negative, positive, or neutral [152]. The employment of sentiment analysis was necessitated due to the vast quantity of data amassed, extracted, and utilized in both structured and unstructured formats, derived from various online sources. [234]. Sentiment analysis applies both NLP and text analysis techniques, either in combination or separately. In this way, reviews and opinions can be categorized according to various standards [152].

As sentiment analysis is a large, developing area of computational text analysis, there is no single best procedure for applying it. Researchers conducted performance evaluations using various methods. These methods included text classification using Naïve Bayes, decision tree classifiers, and N-fold Cross Validation [136]. Knowledge-based sentiment analysis and machine learning-based sentiment analysis are the two major approaches to sentiment analysis. The Knowledge-based approach differs from the machine learning-based approach in that the former is based on NLP algorithms or lexicon methods, [201]. Nevertheless, the latter analyzes data polarity based on previously labeled texts, such as whether a text is neutral, positive, or negative [100].

Business owners and application designers face a number of challenges due to the dramatic expansion of the use of applications and the availability of text data. The primary challenges are derived from analyzing customer reviews and identifying which reviews are positive and which are negative. In each type of polarity, a business owner works differently

to maintain good qualities and enhance bad ones. Furthermore, ratings alone do not provide enough insight for business owners to comprehend ratings, so they need to be combined with text reviews. The sentiment of the reviews should be clearly expressed, and business owners should be able to understand how they are labeled by automated procedures. As a result, developers and researchers must continuously improve the methods for the sentiment analysis of multi-language text and context.

It is possible to use sentiment analysis to infer users' attitudes toward a specific topic by investigating their implicit mindsets in their comments [21]. Furthermore, [258] depicted sentiment analysis as "analyzing people's sentiments, opinions, appraisals, attitudes, evaluations, and emotions towards such entities as organizations, products, services, individuals, topics, issues, events, and their attributes, as presented online via text, video and other means of communication". Opinion mining is another method for sentiment analysis that is based on computational techniques, text analysis, and NLP [301]. There are several kinds of resources for sentiment analysis to operate such as sentence level, document level, and aspect level [34]. Using comments, reviews, and posts from customers, it aims to place products and services in positive, negative, or neutral classes as indicated by the sentiment expressed in the comments [107]. A machine learning approach and a lexicon-based approach can both be used for sentiment analysis [96].

To mention a few, Xing Fang et al. [84] use sentiment polarity categorization to analyze sentiment. In order to categorize the data, RF, SVM, and naïve Bayesian classification methods were selected. Further, as part of other frameworks [42], the text is preprocessed and features are extracted. The tests compare their deep learning approach with the best previously identified classifiers for the same task. Moreover, based on the item's sentimental aspects, the authors of [290] presented a novel approach in this research study. Using pre-processing techniques, the system extracts meaningful information from datasets like positivity or negativity by boxing, stonecoating, tokenizing, and deleting stop-words. Additionally, they provide some insight into their future work on text classification.

2.2 Classification Models

2.2.1 Support Vector Machine (SVM)

SVM was developed by Vladimir Vapnik as a non-linear binary classifier [59, 282, 283]. It is used to classify non-linear data and transform 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 as shown in Figure 2.1. It also means that for any specific input, if there exists such a hyperplane, it must have zero inner product with this input vector. SVM 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.

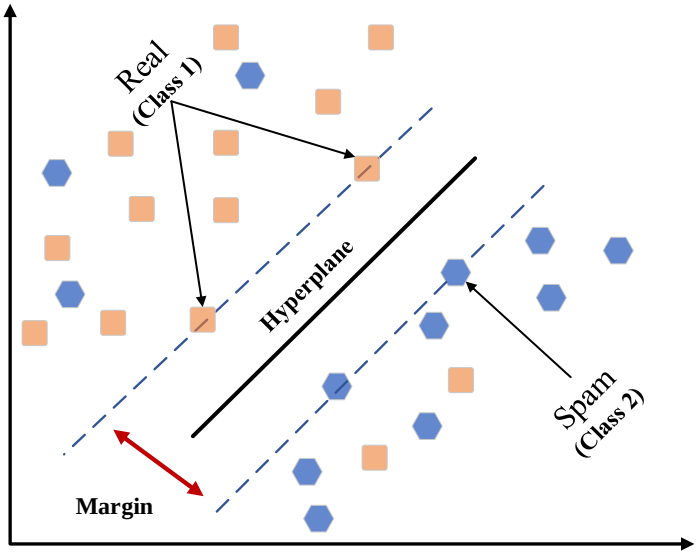


Fig. 2.1 Standard SVM.

In more detail, SVM generates a linear separating hyperplane 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 $(x_{i1}, x_{i2}, \ldots, x_{ip})$, the number of features are denoted with p, $i = 1, \ldots, n$, where n denotes the number of training instances and the class label presented by y_i [123].

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 a hyperplane. 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 [123].

The hyperplane 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, it is conducted a mapping of the initial feature space to more higher-dimensional space (φ). 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 :

$$k\left(x_i, x_i'\right) = \sum_{j=1}^{p} x_{ij} x_{ij}' \tag{2.1}$$

- Polynomial kernels :

$$k\left(x_i, x_i'\right) = (1 + \sum_{j=1}^{p} x_{ij} x_{ij}')^d \tag{2.2}$$

- RBF kernels :

$$k\left(x_i, x_i'\right) = \exp(-\gamma \sum_{j=1}^{p} (x_{ij} - x_{ij}')^2) \tag{2.3}$$

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

The pseudocode of the SVM technique is presented in Algorithm 1. The algorithm accepts the training data and then initializes the optimization problem with the training data. As a numerical optimization method, SVM uses a cost function similar to quadratic programming to find the best solution. Finally, The learning classifier is defined so it separates the data with the hyperplane to distinguish the different classes.

Accordingly, SVM performance depends entirely on its parameters and the selection of the kernel function. In this work, we have considered the RBF kernel in order to deal with non-linear decision boundaries and variables due to its effectiveness according to the literature. The RBF kernel is calculated by Eq. 2.3, where the gamma coefficient is noted as γ. While the other parameter is the cost (C) 1, the penalty parameter, which is used to improve the classification accuracy of the new data points. As a consequence, the inappropriate setting of γ and C could lead to inadequate generalization causing overfitting or underfitting of the data. A model that overfits the training data learns the data too well. In contrast, underfitting occurs when a model does not capture the underlying patterns.

Algorithm 1 Support Vector Machine (SVM) Pseudocode

Require: Training data $D = (x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$, where $x_i \in \mathbb{R}^d$ and $y_i \in -1, 1$
Ensure: Learned SVM classifier $f(x)$

1: Initialize the optimization problem with the training data:

$$\begin{aligned} \min_{\mathbf{w},b,\xi} \quad & \frac{1}{2}|\mathbf{w}|^2 + C\sum_{i=1}^{n}\xi_i \\ \text{s.t.} \quad & y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1 - \xi_i \quad \forall i = 1, 2, \ldots, n \\ & \xi_i \geq 0 \quad \forall i = 1, 2, \ldots, n \end{aligned}$$

2: Solve the optimization problem using a numerical optimization method (e.g., quadratic programming).

3: The learned classifier is defined as:

$$\text{f(x)} = \begin{cases} 1 & \mathbf{w}^T\mathbf{x} + b \geq 0 \\ -1 & \mathbf{w}^T\mathbf{x} + b < 0 \end{cases}$$

2.2.2 Twin Support Vector Machine (TwinSVM)

Unlike SVM, TWSVM generates nonparallel hyperplanes, one positive and the other is negative [145]. Following the same maximum interval idea, TWSVM requires each hyperplane as far away from one of the samples as possible, whereas the other hyperplane is to be close to the other class samples simultaneously as shown in Figure 2.2. As a result, TWSVM constructs hyperplanes for each class and tries to make each hyperplane as close to one class as possible, and as far away from the other class. According to their proximity to the two nonparallel hyperplanes, new instances will be classified into one of the classes.

In the n-dimensional real space *Dn*, assume that the binary classification of c_1 training samples into class 0 and c_2 training samples into class 1. Let matrix Y in $D^{c_1 \times \mathrm{n}}$ corresponds to the training samples of class 0 and matrix Z in $D^{c_1 \times \mathrm{n}}$ denotes the training samples of class 1. In TWSVM, the goal is to find two nonparallel hyperplanes in the n-dimension input space: $x^T w_1 + b_1 = 0, x^T w_2 + b_2 = 0$, where each hyperplane must be as close to its respective class samples as possible, while remaining as far away as possible from the other class samples as possible.

Therefore, the TWSVM formulation can be presented by the following equations:

$$\min \frac{1}{2}(Aw_1 + e_1 b_1)^T (Aw_1 + e_1 b_1) + c_1 e_2^T \xi - (Bw_1 + e_2 b_1) + \xi \geq e_2, \; \xi \geq 0 \qquad (2.4)$$

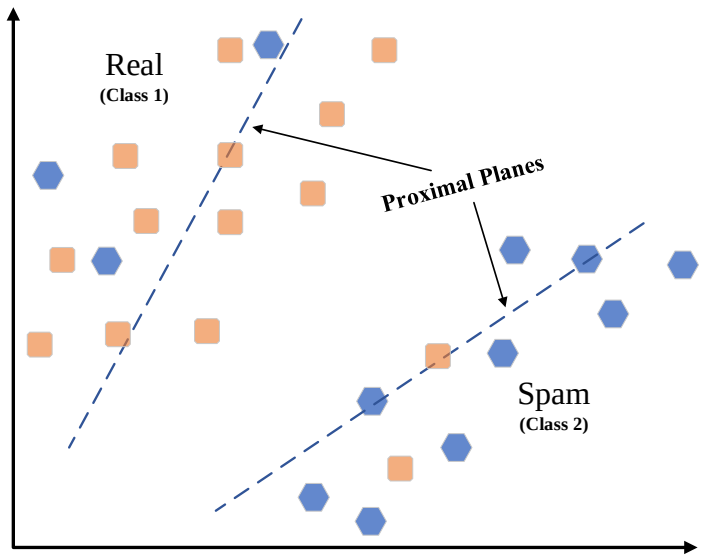


Fig. 2.2 Twin SVM.

$$\min \frac{1}{2}(Bw_2 + e_2b_2)^T (Bw_2 + e_2b_2) + c_2e_1^T\xi - (Aw_2 + e_1b_2) + \xi \geq e_1, \xi \geq 0 \qquad (2.5)$$

where the penalty parameters are c_1 and c_2, the column vectors are denoted by e_1 and e_2, T is the transposition, ξ denotes the slack variable and $w_1 \in R^n, w_2 \in R^n, b_1 \in R$ and $b_2 \in R$.

According to 2.4 and 2.5, the square distance is used to measure the distance from the hyperplane to the samples, then reduce the distance between the hyperplane and its own class samples as possible. As a result of the inequality constraint, the hyperplane distance from the samples ensures to be at least 1.

TwinSVM is used in many applications, including time series forecasting [306], computer vision [180], cancer detection [243], diagnosis of parkinson's disease [274], and gearbox anomaly detection [67]. It shows improved performance compared to the standard SVM and it is a promising algorithm for imbalanced data and kernel selection for the classification task.

The pseudocode of the TwinSVM algorithm is presented in Algorithm 2. The algorithm accepts the training data and then trains two SVM classifiers on these data. The learning classifier is defined so it separates the data with twin hyperplanes to distinguish the different classes of imbalanced data.

Algorithm 2 Twin Support Vector Machine (TwinSVM) Algorithm

Require: Training data $D = (x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$, where $x_i \in \mathbb{R}^d$ and $y_i \in -1, 1$
Ensure: Learned TwinSVM classifier $f(x)$
1: Train two SVM classifiers $f_1(x)$ and $f_2(x)$ using the training data D
2: The learned TwinSVM classifier is defined as:

$$f(x) = \begin{cases} 1 & f_1(x) = 1 \text{ and } f_2(x) = -1 \\ -1 & f_1(x) = -1 \text{ and } f_2(x) = 1 \\ \text{undefined} & \text{otherwise} \end{cases}$$

2.2.3 Other Machine Learning Models

Other machine learning algorithms are considered in this study, including XGBoost, MLP, and KNN, which are described as follows:

- **eXtreme Gradient Boosting (XGBoost)** It is a well-known machine learning algorithm that can work on classification and regression tasks. It is an efficient and scalable algorithm based on gradient-boosting decision trees. XGBoost has several advantages, including improved training speed, support for different types of input data, accepting sparse input, support for a customized objective function and evaluation function, and advanced performance [47].

  XGBoost is a well-known machine learning algorithm that has been applied to a number of tasks, including recommendation systems, computer vision, and NLP. It has been demonstrated to perform better than many other machine learning algorithms on a variety of benchmarks, which makes it an important tool for resolving challenging data science issues.

  The handling of sparse data sets, which are typical in many real-world applications, is one of XGBoost's important strengths. This is accomplished by using a tree-based learning method, which is capable of accurately modeling the interactions between various data features. Furthermore, XGBoost has the ability to automatically determine the right number of trees to include in the model, preventing overfitting and enhancing forecast accuracy.

- **Multilayer Perceptron (MLP)** A feedforward neural network called a multilayer perceptron (MLP) is frequently used for supervised learning tasks like classification and regression [99]. MLPs are made up of several interconnected layers of nodes, with each node standing for a computational unit. The input layer receives the input data and sends it to the hidden layers, where it undergoes a sequence of computations that

abstract and change the data. Based on the altered data, the output layer generates the final predictions.

Speech recognition [271], computer vision [39], and NLP [197] are areas where MLPs have found extensive use. They have been proven successful at resolving challenging issues and can be trained to produce accurate results. MLPs are a popular option for many data science practitioners since they are also reasonably simple to use and comprehend.

The necessity to carefully choose and tune the network architecture, by selecting the number and size of hidden layers, the kind of activation functions utilized, and the learning rate, are the main obstacles when utilizing MLPs. These hyperparameters are crucial to the network's performance and need to be properly selected to provide high performance.

- **k-Nearest Neighbors (k-NN)** is a class of machine learning algorithms used for classification and regression tasks. It uses a method called "K nearest neighbors" to produce predictions, hence the name "k-NN". In a k-NN model, data points are categorized based on the labels of the k" training dataset points that are closest to them. The k-NN algorithm employs the majority class among the K points that are closest to the new point to create a prediction for a new data point. k-NN is a straightforward and understandable algorithm that is frequently used as a benchmark for assessing the performance of more complex ones. Applications like recommendation systems and anomaly detection also make extensive use of k-NN [4, 247].

  The simplicity of k-NN is one of its main benefits. It is quite simple to comprehend and apply because it merely predicts using the labels of the k nearest locations. k-NN is a non-parametric approach, which means it does not make assumptions about the distribution of the underlying data. This makes it an adaptable and reliable algorithm that may be used in a variety of situations.

2.3 Metaheuristic Algorithms

Metaheuristic algorithms are a high-level strategy for finding the most optimal solution for different and complex problems. These algorithms are a group of approximate algorithms that are capable of solving a variety of optimization problems. Through using their dynamic ability to be customized according to the challenges faced, metaheuristic algorithms can achieve this in various ways. Metaheuristic algorithms can handle multiple objectives and constraints, and they can find effective solutions faster than other methods. They are also

robust, can handle incomplete or noisy information, and are easily parallelizable. They can be applied to a wide range of optimization problems, including classifiers, where they can optimize performance and find the best ensemble of classifiers. Additionally, they can handle imbalanced datasets, high-dimensional data, and non-linear problems, and are not sensitive to small variations in the data. Therefore, there would be more space to improve and fix the encountered problems, and improving SVM and TwinSVM performance is not an exception.

There are many different types of metaheuristic algorithms in the literature, such as Genetic algorithms [187], Simulated annealing [281], Ant colony optimization [73], PSO [142], Harris Hawks Optimizer (HHO) [109], Whale Optimization Algorithm (WOA) [184], Salp Swarm Algorithm (SSA) [183], Grasshopper Optimization Algorithm (GOA) [185], and Tabu search [102]. SVM and TwinSVM are some of the algorithms that can be improved through these metaheuristic algorithms, for example, by searching for appropriate values for its hyperparameters [85, 12]. In this thesis, we are going to use the HHO and SSA due to their effectiveness during the experiment process.

2.3.1 Harris Hawks Optimization (HHO)

HHO is a nature-inspired metaheuristic algorithm used for optimization. It is a population-based algorithm that mimics the behavior of chasing style of Harris' hawks in nature called surprise pounce [109]. The algorithm includes several processes, which are the exploration of a prey (rabbit), the surprising pounce, and the attacking strategies of Harris hawks. The main two phases of the algorithm are exploration and exploitation.

In the exploration phase, the hawks wait, observe, and monitor the desert site to detect prey. This phase has two cases, represented in Equation 2.6. The first one considers equal chance q for each perching strategy, which depends on the positions of other family members and the rabbit. The other case perches on random tall trees.

$$X(t+1) = \begin{cases} (X_{rabbit}(t) - X_m(t)) - r_3(LB + r_4(UB - LB)) & q < 0.5 \\ X_{rand}(t) - r_1|X_{rand}(t) - 2r_2X(t)| & q \geq 0.5 \end{cases} \tag{2.6}$$

where $X_(t)$, $X(t+1)$, $X_{rand}(t)$, $X_m(t)$, and $X_{rabbit}(t)$ are the position vector of hawks in the current iteration t, the position vector of hawks in the next iteration $t+1$, randomly selected hawk from the current population, the average position of the current population of hawks, and the position of rabbit, respectively. r_1, r_2, r_3, r_4, and q are random numbers in the range of [0,1] while LB and UB are the upper and lower bounds of variables. $X_m(t)$ is calculated by Equation 2.7 where $X_i(t)$ is the position of each hawk in iteration t and N is the total number of hawks.

$$X_m(t) = \frac{1}{N}\sum_{i-1}^{N} X_i(t) \tag{2.7}$$

The escaping energy of the prey decreases according to Equation 2.8 which affects the transfer behavior from exploration to exploitation and the difference in the exploitative behaviors. E is the energy value, T is the maximum number of iterations, and E_0 is the initial state of its energy in the range of [-1, 1]. The exploration behavior appears when $|E| \geq 1$ whereas the exploitation behavior appears when $|E| < 1$.

$$E = 2E_0(1 - \frac{t}{T}) \tag{2.8}$$

In the exploitation phase, the hawks perform a hard or soft besiege to catch the prey when $|E| \geq 0.5$ and $|E| < 0.5$, respectively. In the soft besiege the Harris' hawks make the rabbit more exhausted by applying the surprise pounce, represented in Equations 2.9 and 2.10. In the hard besiege, Equation 2.11 shows the change in the position.

$$X(t+1) = \Delta X(t) - E|JX_{rabbit}(t) - X(t)| \tag{2.9}$$

$$\Delta X(t) = X_{rabbit}(t) - X(t) \tag{2.10}$$

$$X(t+1) = X_{rabbit}(t) - E|\Delta X(t)| \tag{2.11}$$

where $\Delta X(t)$ is the difference between the position vector of the rabbit and the current position in iteration t, $J = 2(1 - r_5)$ is the random jump strength of the rabbit when escaping, and r_5 is a random number of the range [0, 1].

Hawks can progressively select the best possible dive toward the prey in competitive situations. In the soft besiege, Equations 2.12, 2.13, 2.14, and 2.15 are applied, where LF is the levy flight function (used for time series analysis and decomposition for forecasting purposes) [126], S is a random vector by size $1 \times D$, u and v are random values of the range [0,1], β is the value of 1.5. In contrast, in the soft besiege, hawks decrease the distance of their average location with the escaping prey, represented in Equations 2.16, 2.17, and 2.18

$$Y = X_{rabbit}(t) - E|JX_{rabbit}(t) - X(t)| \tag{2.12}$$

$$Z = Y + S \times LF(D) \tag{2.13}$$

$$LF(x) = 0.01 \times \frac{u \times \sigma}{|v|^{\frac{1}{\beta}}}, \sigma = \left(\frac{\Gamma(1+\beta) \times \sin\frac{\pi\beta}{2}}{\Gamma(\frac{1+\beta}{2}) \times 2^{\frac{\beta-1}{2}}} \right)^{\frac{1}{\beta}} \tag{2.14}$$

$$X(t+1) = \begin{cases} Y & if F(Y) < F(X(t)) \\ Z & if F(Z) < F(X(t)) \end{cases} \tag{2.15}$$

$$X(t+1) = \begin{cases} Y & if F(Y) < F(X(t)) \\ Z & if F(Z) < F(X(t)) \end{cases} \tag{2.16}$$

$$Y = X_{rabbit}(t) - E|JX_{rabbit}(t) - X_m(t)| \tag{2.17}$$

$$Z = Y + S \times LF(D) \tag{2.18}$$

2.3.2 Salp Swarm Algorithm (SSA)

SSA is a metaheuristic algorithm, inspired by the swarming behaviour of salps [223, 224]. Salps are Salpidae that have transparent barrel-shaped body. It is displayed in Figure 2.3. The salp chain includes the leader salp and follower ones. The leader salp is the one in the front of the salp chain which guides the swarm to the food position, whereas the remaining salps are the followers [26]. The positions of the leading and follower salps are changed according to two mathematical models, which are given by Equations 2.19 and 2.20, respectively [183].

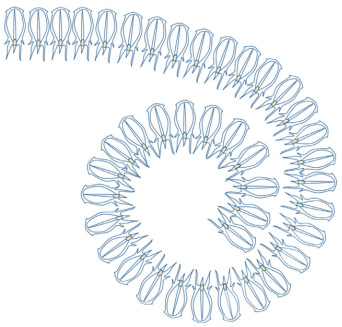


Fig. 2.3 Salp chain [183]

$$x_j^1 = \begin{cases} f_j + c_1((ub_j - lb_j)c_2 + lb_j)c_3 \geq 0 \\ f_j - c_1((ub_j - lb_j)c_2 + lb_j)c_3 < 0 \end{cases} \tag{2.19}$$

$$x_j^i = (x_j^i + x_j^{i-1})/2 \tag{2.20}$$

where x_j^1 is the position of the leader and f_j is the position of the food source in the j^{th} dimension. ub_j and lb_j are the upper and lower bound, respectively, for the j dimension. c_1, c_2, and c_3 are random numbers. x_j^i is the position of the follower i in the j^{th} dimension. c_1 is calculated using Equation 2.21.

$$c_1 = 2e^{-(4t/T)^2} \tag{2.21}$$

The algorithm works as follows:

1. Randomly create the salp chain based on the lower and upper limits.
2. Calculate the fitness of each salp in the chain.
3. Update the parameter c_1.
4. Change the position of the salps according to equation 2.19 and 2.20.
5. Record the best salp with the best fitness value.
6. Repeat Steps 2 - 5 according to the selected number of iterations.
7. Return the best salp of the last iteration.

SSA solves various optimization problems and can be found in many applications [2] including, feature selection [307, 13], training neural networks [3, 80], scheduling [269], control of power systems [76, 244], renewable energy system [239, 189], and other applications [22, 169].

SSA is a simple algorithm and it has only one parameter which makes it easy to control. SSA can also be hybridized with other optimization algorithms [2]. However, SSA has some problems such as slow convergence, single-objective optimization, and the low ability to control the difficulties of multimodal search strategies [2].

Chapter 3

Literature Review

3.1 Introduction

As part of this study, we outline the concepts and detection methods of spam reviews, sentiment analysis, and spam reviews utilizing sentiment analysis along with their implications in the environment of online reviews. The analysis revises the most remarkable spam reviews detection and sentiment studies. Furthermore, different detection and identification approaches have been discussed in order to investigate their specific advantages, limitations, and ways to improve their performance.

In order to identify relevant articles for this chapter. The selection criteria comprise the inclusion and exclusion phases. If a duplicate article showed up in the inclusion list more than once from different sources, it will still be included only once.

Due to the rapid alteration and increase of spam reviews or sentiment analysis recently. This chapter analyzes the mechanism used for each category alongside their limitation, and strengths, as well as providing solutions to enhance their performance.

Therefore, the objectives of this chapter are summarized in the following points:

- Outline the previous spam detection or sentiment analysis studies.
- Analyze all studies and identify their limitations, advantages, and how to improve them.
- Present a literature analysis and discussion of spam reviews detection and sentiment analysis.

This chapter consists of the following: Section 3.2 presents the literature studies on spam reviews detection. The studies of sentiment analysis are described in Section 3.3. Section

3.4 proposed the studies of spam reviews detection through the sentiment analysis method. Finally, Section 3.5 discusses previous works on TwinSVM in general.

3.2 Spam Reviews Detection

In this section, the spam reviews are addressed with a focus on their respective approaches. These approaches are categorized as Behavioral Features, Supervised Learning, Semi-Supervised Learning, Deep Learning, Multilingual Approaches, Ensemble Methods, Graph-Based Techniques, and Sentiment-Based Approaches.

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.

An example of such detection type is the study in [273] that explored spam reviews by employing a 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 is applied on the data for creating synthetic features (i.e. text features, rating, and metadata features). Furthermore, novel generators and discriminators are implemented for optimal training. By using this process, the GAN is able to create synthetic behavior features for new users. Using the Yelp dataset (a website for reviews and recommendations) as the basis for their experiments, the results indicate that the approach outperforms other approaches.

The work in [72] is another example of behavior detection. In the paper, a model was presented to distinguish genuine reviews from spam. A Random Forest is exploited in this model using end-to-end training (train all steps from the start to the end). Further, in order to identify the global parameter of a learning model, the decision tree technique is applied. When compared to other methods on the Amazon review dataset (a website for buying products), the experiments in this paper show that the proposed model produces better results.

The literature has also presented a novel method of detecting spam reviews, namely Supervised Learning-based detection. As part of the ML methods, this type relies on labeled samples to learn. A variety of studies have used supervised learning for detecting spam reviews [88]. The authors applied 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) [94] method to enhance the detection process.

Meanwhile, [143] has also adapted the supervised learning technique to address online review falsification. A neural approach was employed to classify the reviews using syntactic and lexical patterns in order to solve such a problem. A comparison is made between various types of supervised classification models and their proposal. In order to examine all approaches, they used Google's word embedding architecture (BERT). Also, a method for detecting fake reviews in the Tourism environment was presented by [193] (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 from tricking them.

Many researchers propose different approaches to solve spam review detection using deep learning thanks to its capability to learn from big data. As an example, consider the work presented in [18], 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, while, in terms of detecting spam reviews, the proposed CNN is more effective than other neural network architectures. Moreover, [83] also discusses 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, [63] 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 [295]. 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 of [250] raises the issue of labeled datasets being unavailable in spam reviews. In order to distinguish spam reviews, the authors utilized LSTM networks based on unsupervised learning. By training it with real reviews without labels, the model discovers patterns in the reviews. The experimental results indicate that the approach is capable of distinguishing real reviews from spam. In its study of spam reviews, [310] 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 (DBN) to specify the credibility of product reviews. To build a DBN model, standard features are combined with semantic features.

Similarly, in [35], the authors discuss the shortcomings of traditional ML approaches for detecting spam reviews. The 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 identifying spamming indications in the sentences and documents. After the sentence representation is analyzed, a CNN is constructed and discovers 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.

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 by [121], the majority of spam review detection studies have focused more on languages such as English, Chinese and Arabic. Thus, the authors 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 [120]. 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

In their paper [125], 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 variances. With regard to accuracy, Ensemble Bagged Classifier exceed all other supervised methods. Additionally, [163] presents an unsupervised method for detecting spam reviews on Chinese texts, images, videos, and other media. The following conclusions were found: 1. Video and text spam are less common than image spam; 2. Preferring to steal from reviews (ideas and suggestions) 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 [19], 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. By the results, the researcher 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 [81]. 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 used 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 the [209] study 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 a good rate in terms of accuracy.

Spam reviews based on semi-supervised methods are one of the approaches used for detection. A semi-supervised learning is a technique that operates on a dataset that combines labeled and unlabeled instances [312]. That is to say, the semi-supervised method is a merge between unsupervised learning (non-labeled instances) and supervised learning (labeled instances). Usually, the amount of unlabeled parts of sets exceeds the number of labeled sets by far.

In literature, this method obtained attention from researchers with its performance in handling different types of datasets. For example, [275] argues that there are many studies presented to solve the identification between real and spam reviews. However, the issue is still challenging to manage for supervised learning due to the lack of labeled data samples as well as imbalanced problems. Therefore, the authors see that using a semi-supervised learning approach is better for spam reviews problem. Their approach -Ramp One-Class

SVM- consists of a nonconvex semi-supervised method. Which operates on one-class data in order to deal with the lack of labeled datasets. Also, solving the outliers and non-review using the nonconvex properties of the proposed approach loss function. The experiments were executed on two datasets, namely Yelp and Ott. The outcomes show that the proposed method outperforms other techniques in terms of accuracy, precision, and recall.

Moreover, the work in [294] also studied the performance of the semi-supervised method on spam reviews detection. The hybrid semi-supervised learning approach that they presented relies on the user–product and users' characteristics relations. In their work, the hybrid PU-learning-based spammer detection (hPSD) starts its detection process by producing various positive samples. Then the semi-supervised learning classifiers are evaluated on the movie dataset. The hPSD achieves the best results when compared with other approaches. Furthermore, the approach used the real-life Amazon spam reviews detection dataset for more examinations.

Another recent framework proposed by [122] for spam reviews detection using semi-supervised learning. The work simply used the adversarial training mechanism that exploits the Generative Pre-Training 2 (GPT-2) abilities to detect spam reviews against unlabeled and labeled data. The experiments were performed on the TripAdvisor (a website for place recommendation) and YelpZip datasets. The presented model obtained the highest results with 7% more in terms of accuracy. Moreover, due to the lack of labeled data, the model generates synthetic samples (spam/non-spam reviews) to prove more labeled instances to learn better.

Ligthart et al. [168] stated that in real-world situations the datasets that are used usually lack the required labels to perform on supervised models. Therefore, applying the semi-supervised learning approaches can be more efficient. The purpose of this study is to explore the effectiveness of various types of semi-supervised learning as a classification method. The authors examined four different semi-supervised models in order to classify the spam online reviews for hotels. Additionally, the experiments comparison shows that the Naïve Bayes reach the best results in terms of accuracy. The study demonstrates the need to mitigate the labeling efforts while focusing more on model performance when the labels are limited. If Naïve Bayes gets very good results, it would be expected that more advanced techniques would get much better ones. Whereas, [205] introduced a dictionary based on online reviews and social network terms to find the hidden patterns and relationships of these terms. A number of language features were used, including, length of reviews, presence, and frequency of bigram (two-word sequence of words), and presence and frequency of unigram (single word sequence of words). The work employed both supervised and semi-supervised methods to detect spam reviews with the help of linguistic and behavioral features.

According to previously addressed studies, a semi-supervised approach is desired the most when the data are not fully labeled. Such scenarios can usually occur in real-world situations. Therefore, these kinds of approaches are important to overcome the problem of limited labeled data.

Furthermore, ensemble methods can also be utilized for spam reviews detection. These ensemble methods are known as machine learning techniques that consist of a set of classification models as shown in Figure 3.1. The point of aggregating them together is to take into account their weighted predictions as a vote. This type of machine learning method is performed in order to enhance the classification accuracy and depend more on several models rather than one model (best model). Fayaz et al. [86] presented an ensemble approach that combines different classifiers, which are, Random Forest (RF), Multilayer Perceptron (MLP), and K-Nearest Neighbour (k-NN) for detecting spam reviews. Their ensemble model was evaluated on the Yelp dataset alongside using different types of feature selection methods to select the best subset of features. The proposed approach outperforms the individual classifiers (RF, MLP, and k-NN), and other state-of-the-art methods.

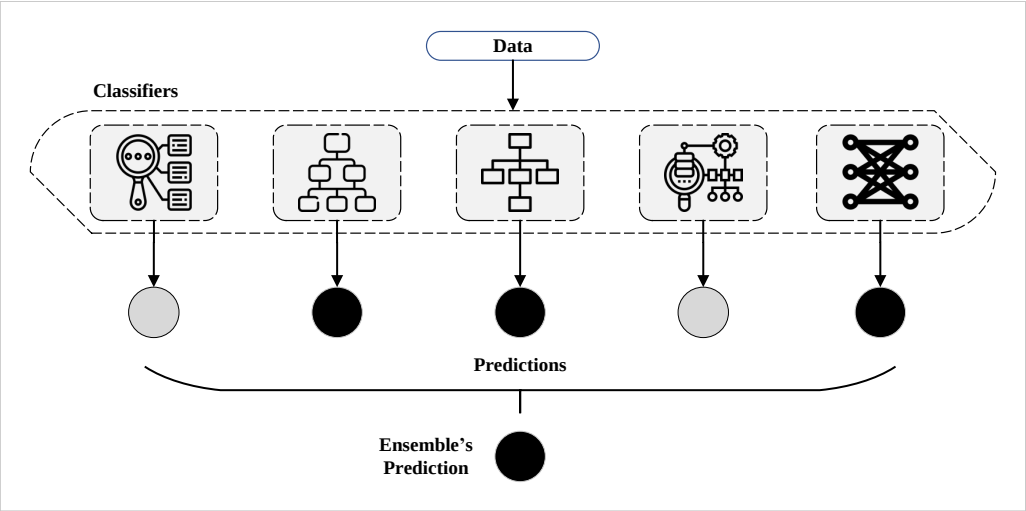


Fig. 3.1 Ensemble learning architecture.

Moreover, [191] proposed a novel study for spam reviews detection using ensemble machine learning methods. In order to improve the accuracy of classification, an additional study was performed to analyze features in combination with ensemble methods. Further, four ensembles are used, including, Extra Tree, Bagging, Random Forest, and Boosting. The ensemble method obtained the best results compared with other methods. Another recent work applied the ensemble model for the identification of spam reviews [300]. Four different steps are used in this study: Scilicet, Data resampling, Feature pruning, Parameter optimization, and classifier ensembling. Then a resampling technique is employed to solve

the imbalance problem. The results verified the performance dominance of the model against other methods in spam reviews detection.

Additionally, the authors of [124] introduced a spam reviews detection method based on the ensemble machine learning technique. Three models are given and trained on multi-view learning techniques to ensemble them together for majority prediction. Then they extract the text information of the reviews through parallel convolution neural networks (CNN) and bag-of-n-grams method. The CNN architecture employed the n-gram embeddings as input and extracted feature representations of reviews by using parallel convolutional blocks. The evaluation phase takes place on Yelp Filtered dataset with a good rate in terms of F1 scores. While the study of [43] built a detection framework for spam reviews using single and ensemble models. Also, the work proposed two random sampling methods for solving the class imbalance issue. The experimental results show that the sampling methods (random under and over-sampling) enhanced the classification accuracy of spam reviews. Besides, the Adaptive Boosting ensemble obtained the highest results for smaller datasets, while single classifiers achieved best when the dataset was larger.

The indication of previous studies suggested that using more than one classification model achieved better results in detecting spam reviews. In other words, taking the vote weighting of several classification models is better than relying on the best model, where the performance is more reliable due to the fact that various decisions are taken.

On the other hand, few studies used the graph-based for spam reviews detection. Graph-based spam reviews detection is a process of mathematically representing the network and their relationship through lines and points to describe the connection of the reviews entities. That is to say, the graph-based approach is a branch of data mining techniques that depend on the inter-dependencies between objects. The approach is used to study the patterns of the relations of a network in order to distinguish the hidden patterns. In this case, graph data of the spam reviews (fraudulent) rely on the rating of the reviewer for a product and the other reviewers' rate for that product. While the non-spam reviews (trustworthy) rely, somehow, on different products that they rated. Therefore, graph-based detection introduced a robust mechanism for capturing and identifying the relationships and correlations of data objects inter-dependent.

For instance, [36] provided a novel graph-based spam detection model to recognize the reliability of reviews. The paper compared various methods for the sake of proving the efficiency of the proposed model. Different attack scenarios are taken into account when examining the model. The results indicate that the proposed model identifies the spam reviews in different scenarios efficiently. Moreover, it can reduce the trust value of the spam review as well as prevent spammers from ruining the reputation of the product. Sundar et al. [270]

used a Deep Dynamic Clustering model to detect spam reviews by utilizing graph embedding structure to preserve the nonlinear information structure of the text. The authors also applied dynamic aspects of reviewers in order to determine the spam users from the normal ones. In the phase of the experiment, the results show an excellent performance in spam detection. While [206], stated that the current spam reviews detection techniques used one to two kinds of entities only alongside employing a few types of features, namely content, relation, and behavior features. However, such techniques suffer from not being able to be employed in context and tend more to synthetic criteria. Therefore, a new graph-based model is presented to detect spam reviews. The model (Multi-iterative Graph-based opinion Spam Detection (MGSD)) is utilized with all classes of entities as well as a unified structure. Afterward, both the implicit and explicit relation is used, then an evaluation of the Spamicity effects is applied. In order to improve the performance of the detection model number of weighted features were taken into account. The outcome of the experiments indicated the superiority of the proposed model for the synthetic dataset and Ott's dataset, respectively.

Further, [291] declared that the previous studies in the literature consider several reviews as a whole when performing feature extraction, at the same time, disregarding the internal differences and similarities. Thus, causes to not identify the most discriminative information due to semantics reviews disorder. To solve such a problem, they proposed a co-attention framework and orthogonal model for the decomposition process. Hence, learn the differences and similarities between the reviews. Also, to distinguish the strong social connection, they determined first the weak relation graph through the dynamic interactions method. Subsequently, graph representation is applied to learn the interaction of social connections. Their framework outperforms the state-of-the-art methods evaluated on two real-world datasets. Whereas, the authors of [44] designed a community detection technique (SC-Com) for spam reviews. SC-Com used the graph of reviewers for splitting the communities based on their reciprocal doubt. Then a temporal abnormality and community-based features extraction are employed for identifying the spam reviews from non-spam. Using a real-world dataset, an evaluation phase is done and the presented approach achieves the highest results utilizing ratings and time data.

The graph-based approach relies on the relationship between different entities of the review process, namely review, reviewer, rating, and so on. Hence, non-labeled data is needed and low complexity of the structure review compared to other approaches. This type of detection method facilitates the obtain of relation and interaction hidden-pattern between the network. Accordingly, making graph-based detection techniques different from other methods. As a result, it depends on other factors which aren't accessible in other cases.

In addition, most of the methods focus on solving spam reviews using different detection systems, such as Content-Based detection, Concept Drift benchmark detection, Epistemic Belief, and Unsupervised Learning.

For instance, [289] implemented their detection system through a collective technique to identify spammers and spam reviews (adversarial type). The process began by training two models Content-Based Module (CBM) and Behavior-Based Module (BBM). Afterward, both models were co-trained in order to receive each other feedback. According to their results, their method performs better in determining the adversarial reviews. Another example of this category for detecting spam reviews is proposed by [219]. The work developed a novel automated tool for identifying spam reviews using 1041 respondents. Then a comparison of the non and spam reviews with psycholinguistic deception cues. The findings show that the tool obtained a good rate in terms of accuracy.

Moreover, [298] presented a topic model and reviewer anomaly rate method for spam reviews detection. The authors split the spam reviews into deceptive and content-type. In the first part, the dataset is modeled through a Latent Dirichlet Allocation topic that observes the reviews' content type, while in the second part, the abnormality degree index is used to recognize the deceptive reviews. A score was assigned for each review and then merged with adaptive weight calculation according to the similarity and abnormality of the reviews. High-score reviews are considered spam, whilst low-score review is recognized as non-spam. Mohawesh et al. [192] investigated the drift problem in spam reviews using two techniques, namely content-based classification, and concept drift benchmark-detection techniques. Four real-world datasets were employed for the evaluation matter, the results show that there is a negative correlation between the performance of spam detection and concept drift.

The authors of [30] proposed a study about online reviewers' authenticity detection using algorithmic-based technique. The study takes into account the epistemic belief role, which is the person's ability to decide between facts and falsehood. More than 300 participants were getting involved in order to classify reviews, spam, and authentic. Using epistemic belief and some kind of justification to reduce the relationships between specificity and exaggeration. While the work in [233] declared that feature selection is one of the important methods to enhance the classification of spam detection and reduce the computation time of the training phase. Hence, they study the impact of various types of feature selection methods in detecting spam reviews. Several feature selection methods have been applied and trained on four classification models to improve the performance. Additionally, an examination is performed on three well-known datasets alongside the different types of feature selection methods, including, bigram, unigram, word embedding, and frequency count. Their experimental results prove the effects of using different factors on classification performance. Furthermore, [221]

proposed a detection approach for online reviews manipulation. The authors collected a number of reviews from 500 doctors, consisting of textual feedback and ratings (1 to 5). Their study explored the procedure to verify the negative reviews, then rank the doctors and reduce risks in the healthcare environment.

Additional recent research about pseudo-reviews detection presented by [61]. Two studies have been done for the purpose of identifying such reviews. The first study concentrates on ensuring the slight impact on product attitude when it's in an isolated environment. In the second study, both pseudo and authentic reviews are combined together, which shows that the impact was negative on purchase intentions. On the other hand, the work of [304] discussed the aspects of how to increase the confidence in reviews. Also, the study analyses two review sites and debates their trustworthiness, namely TripAdvisor and Booking. Three types of analysis were provided, verification analysis, SWOT analysis, and processual analysis. Their model provides verification of tourism services reviews in destination and reviews.

Besides, few studies applied the unsupervised learning technique for spam reviews detection. One example for this was proposed by [97], where the paper focused on an abandoned domain (movie reviews) and implemented a new unsupervised spam identification model based on an attention mechanism. Also, statistical features extraction was performed as well as introducing an attention mechanism for review embedding and applying conditional generative adversarial network for training the model. The results demonstrate the outperform of the model compared with other state-of-art approaches. Another example that employed the unsupervised learning technique was proposed in by [153]. The authors stated that unsupervised machine learning methods can use the clusters as features. This could help the classification model's performance during the feature reduction. In other words, the classification models improved when they were run after the clustering process. The outcomes of the experiments prove the enhancement of the SVM classifier when K-means clustering was applied earlier. Further, different feature selection methods are utilized to seek the optimum performance from the classification model.

Additionally, more advanced methods have been investigated during the development and maturation of the spam reviews detection analysis. One of these methods was using metaheuristic algorithms to enhance the ML classifiers for detecting spam reviews. For example, the authors have developed a spiral cuckoo search-based clustering approach [213]. In this method, 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 (spam tweets) was also used to prove the effectiveness of the proposed method. To validate the capability of the proposed clustering method, six metaheuristic clustering methods are compared with it. Both experimental results and

statistical analysis prove that the proposed method is faster and more accurate than existing approaches. An algorithm based on the military dogs' squad is presented in this paper [278]. Through this algorithm, the military dogs' searching abilities are mimicked. A performance test is performed 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 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 this paper to unravel the fake review detection in a big data environment [155]. 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 approach 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 [230]. The dataset for analysis was selected from Amazon's automotive products. In comparison to the 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 [228], an effective hybrid feature selection technique using Cuckoo Search with Harmony search is proposed and Naïve 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 this paper to extract features [229]. As an objective function, they use the accuracy of the Naïve Bayes classifier (NB). Comparing the proposed method to other competitive methods, the experimental results indicate that this method selected only the informative features and gave higher classification accuracy as compared to the others.

Another work for spam reviews detection is presented by [227], wherein the feature selection phase, a hybrid optimized Binary Particle Swarm Optimization (iBPSO) method is combined with Cuckoo search (CS). In order to classify reviews as spam or ham, Naïve Bayes and k Nearest Neighbors are used. According to the experimental results, the proposed algorithm consistently outperforms other Binary Particle Swarm Optimization (BPSO) algorithms.

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 [105] 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) word embeddings; (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. This paper proposes a novel content-based approach for review spam detection that takes into account both the bag-of-words and the word context [32]. To build a vector model, they exploit n-grams and a method called skip-gram word embedding. 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 curve (ROC).

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 ML algorithms [253]. The disadvantage of deep learning, in this case, is that only 1600 and 2000 reviews were used from Ott and Yelp Dataset, 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 word embeddings, 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, CNN, LSTM, MLP, and Recurrent Neural Network (RNN).

Unlike the previous studies, our first approach in Chapter 5 presents a different method for detecting spam reviews by utilizing metaheuristic algorithms for optimizing parameters and assigning weights to features. It also incorporates pre-trained word embeddings for multilingual analysis and examines reviews before and after the COVID-19 pandemic.

3.2.1 Literature Analysis and Discussion

In this subsection, detailed analyses and interpretations of the presented studies are proposed. These studies have their advantages and disadvantages. Therefore in this discussion, we will investigate their definition, strengths style, and weaknesses.

The first genre of studies focuses on studies that depend on behavioral features. These types of features are considered important and essential when trying to identify spam reviews.

Due to the fact that they rely on the attitude of the spammers themselves. As such, many researchers combined this method with their main approach thanks to its ability to detect some kinds of reviews. Examples of such behaviors could be the style of writing, the time of posting, the rating of each review, the name of their users, and so on. However, the behavior features technique could be tricked by using the correct methods (for example, changing the style of writing) without considering other combined schemes with this technique, such as machine learning methods.

As for the second genre, the supervised learning detection model is a well-known technique applied not only for spam reviews but also in different security and other domains. Supervised learning has a powerful ability to recognize the hidden pattern of reviews, including the relation of reviews, reviewers, products, features, and labels between each other. Such a pattern can help to improve the identification of spam reviews efficiently. Hence, it is considered one of the best methods to detect spam reviews recently in the literature. Nevertheless, this technique has an essential weakness when performing: the lacking of many labeled data and the preparation of the data. This is happened due to the fact that labeling and pre-processing of the data consume time, effort, and money.

Further, the semi-supervised learning method attempted to solve what supervised learning lacked. Handling unlabeled data can be resolved by utilizing the traits of semi-supervised learning without any problem. Semi-supervised can perform with a few numbers of labeled instances while the rest of the instances are unlabeled. That's why semi-supervised learning is considered as relatively inexpensive, where no need to spend money and time on labeling the whole dataset. In other words, it has the traits of both the supervised and unsupervised learning methods. On the other hand, the semi-supervised technique has two main disadvantages, which are the instability of iteration results and usually obtaining low accuracy.

Deep Learning is considered the future of machine learning, where it can detect spam reviews efficiently and dynamically. Particularly in spam reviews, where the construction of features is a necessity. Deep learning can handle different types of features, applications, and datasets. Thus, it can easily outperform other approaches in spam reviews detection. However, deep learning requires huge data (big data) to perform and outperform other methods. Also, it consumes a lot of time to train the model. Therefore, it needs more effort to deal with when optimal performance is the objective.

Ensemble machine learning-based achieved excellent performance for spam reviews detection. Either using the voting method or the stacking method (a new classifier learn from the previous one). Ensemble can handle complex and difficult problems that need multiple hypotheses as well as it is unlikely to have an overfitting issue. On the negative side, ensemble models are hard to interpret and understand the predictions (why these reviews are

spam) due to their complexity. Such complexity is also computationally expensive and needs time and memory to perform.

While graph-based spam reviews detection method consecrates on the relationships of the entities. In this scenario, these entities consist of reviews components and their relationship with each other. Thus, the detection phase of the graph-based depends on a low structure complexity alongside non-labeled data required. This method has also a number of advantages, such as summarizing huge datasets visually, the ability to compare several datasets, and can identify trends accurately. However, the graph-based method needs more analyses to perform than other approaches, where data misinterpretation could occur easily (ignoring important info, overlooking some prior knowledge, and focusing on irrelevant data) without taking the proper procedures. Moreover, multilingual approaches usually depend on the language characteristics and the context used in the reviews. Also, it addresses different languages to English alongside various regions to detect spam reviews. These languages and regions have different characteristics. Hence, it is important to have such a detection approach that can handle the linguistics, context, culture, regions, and multilingual spam reviews. But this method is hard to perform without the proper persons that can understand the used language. Also, it lacks the diversity of linguistic spam reviews detection methods.

Finally, the spam reviews detection based on sentiment analysis utilized natural language processing techniques to operate. This kind of detection attempts to understand and interpret the textual content in order to assist the process of spam reviews identification. Therefore, knowing different and unique aspects can improve the interpretation of some reviews content that is hard to understand on other methods. Nevertheless, combining sentiment analysis for spam reviews detection is not mature enough and needs more investigation. Hence, some wrong perceptions could take place. Table 3.1 illustrates the advantages and disadvantages of the aforementioned approaches.

In summary, there are many ways to detect spam reviews. Some of them are mature while others require more examination and study. Each approach has its own advantages and disadvantages. Hence, it is good to use different approaches depending on the problem we face. Some methods can handle small data and others can handle large ones, several requires labeled data and others don't, a few of them need more time to perform while others are needless, and so on. Thus, as mentioned previously the problem is what helps us to select the proper approach, not the other way around.

We also noticed that there are differences between the spam reviews studies before and after the COVID-19 situation. The rise of people staying indoors is responsible for the increase of reviews is an example. Between the two periods, different behavior of reviews is evident when using products and services or customers' target, context, and impact.

Table 3.1 Advantages and disadvantages of the reviewed approaches.

Approach	Advantages	Disadvantages
Behavioral features	Identify spam reviews by using the attitude of the spammers.	Modifying users' behavior (changing the style of writing).
Supervised learning	Powerful to recognize the hidden pattern of reviews, including the relation of reviews, reviewers, products, features, and labels between each other.	Lack of many labeled data and the preparation of the data.
Semi-supervised	Solve the supervised learning weakness, handling unlabeled data.	Instability of iteration results and obtain low accuracy.
Deep learning	Handling different types of features, applications, and datasets.	Requires huge data (big data) to perform.
Multilingual approaches	Handles different linguistics, contexts, cultures, regions, and multilingual spam reviews.	Hard to perform without the right persons that can 726 understand the used language.
Ensemble	Can handle complex and difficult problems that needs multiple hypotheses.	Hard to interpret and understand the predictions (why these reviews are spam).
Graph-based	Consecrate on the relationships of the entities. Depends on a low structure complexity alongside non-labeled data required.	Needs more analyses to perform than other approaches (data misinterpretation could occur easily)
Sentiment-based	Knowing different and unique aspects can improve the interpretation of some reviews content that is hard to understand on other methods	Not mature enough and needs more investigation

With more people reading and searching for reviews, spam reviews are also growing in number and, models of detection techniques improve and become more diverse.

3.3 Sentiment Analysis

This section delves into the exploration of sentiment analysis studies and the approaches employed to enhance the precision of sentiment identification in textual data.

In text analysis, sentiment is used to measure the text's emotional content. In accordance with Tubishat et al.[279], sentiment analysis, which is also known as opinion mining, emotions, attitudes and evaluations of services or products are studied in order to detect orientations. By analyzing text posted online, it detects feelings expressed through text analysis, linguistics or natural language processing. In this way, it can be beneficial for many individuals organizations, and governments. Each of which, can change, improve or cancel their decisions. As a result, they will be able to save time, effort, and money.

Sentiment for products and services reviews can help companies to upgrade or stop generating of one of the products and services. Such reviews can also be a form of a feedback of the products. Thus, interpret users opinions about the products in best possible way. There are two main approaches to do sentiment analysis, including, knowledge-based or ML-based. In contrast to the latter, the former uses NLP algorithms and lexicons techniques [201], meanwhile and based on training on data previously labeled by humans, the latter determines whether data is positive, negative, or neutral [100].

Businesses can use sentiment analysis to enhance business owners, service providers, and customers' decisions [111]. Also, the company's image and success can be improved by using Sentiment analysis [303]. The business can provide more praiseworthy services if they use customer reviews to improve the quality of their products and services. In turn, this increases customer satisfaction and income for the company [258]. The reviews can further be used by customers for making intelligent decisions based on previous reviews [111].

As explained earlier reviews recently have been changed and improved due to the covid-19 pandemic. More individuals nowadays depend on them as a result of isolation throughout the situation period. Thus, more attention and focus on reviews posted online. Also, the products and services that were used during and after the pandemic have changed.

The COVID-19 pandemic has made it increasingly important for customers to read online reviews in order to make safe dining decisions. Thus, to assist restaurants in understanding customer needs and maintaining their business under current conditions. The authors of [175] identifies restaurant features that customers care about in current circumstances. Furthermore, a deep learning algorithm is introduced to analyze customer opinions and identify reviews

with mismatched ratings. On Yelp.com, more than 112k restaurant reviews were analyzed between January and June 2020. The features that were most frequently cited by restaurant patrons (e.g., experience, food, service, and place) were identified along with the associated sentiment scores.

In addition to optimizing the performance of ML methods, evolutionary algorithms can also select features. Therefore, combing the evolutionary algorithms with classification models for sentiment problems (e.g., a large number of features) achieved excellent results according to the literature. For example, [77] provided a comprehensive comparison of different hyperparameter tuning methods for sentiment classification; including, Genetic Algorithm (GA), Bayesian Optimization, Grid Search, PSO, and Random Search. Six ML algorithms are optimized using them, NB, Ridge Classifier (RC), Decision Tree (DT), Logistic Regression (LR), SVM, and Random Forest (RF). With a score of 95.6 when using Bayesian Optimization, SVM had the highest accuracy before and after hyperparameter tuning. To address the imbalanced data problem, the authors proposed a hybrid approach that combines SVM with PSO and different oversampling techniques [208]. The dataset, which contains different reviews of several restaurants in Jordan, is optimized using SVM as a ML classification technique to predict sentiments. Weights are optimized using PSO and four oversampling techniques have been used in order to solve the imbalanced problem, including, SVM-SMOTE, borderline-SMOTE, Synthetic Minority Oversampling Technique (SMOTE), and Adaptive Synthetic Sampling (ADASYN). Based on the results of this study, they conclude that PSO-SVM produces the best classification results among a variety of classification methods. Furthermore, SVMs are combined with WOA for automatically tuning hyperparameters and weighting features to produce a decision support system based on sentiment analysis [207]. They used a hybrid evolutionary approach to study people's Facebook posts regarding the government's pandemic decisions, combining sentiment analysis with decision support systems. A total of five versions of each collected dataset were generated. In terms of accuracy, the WOA and SVM (WOA-SVM) obtained the best with 78.78% in comparison with other metaheuristic algorithms and standard classification models.

For the sentiment classification task, a novel metaheuristic optimization approach is proposed [305]. Using Ternary Bees Algorithms (BA-3+), this approach considers only three solutions per iteration to address large dataset classification problems. The proposed deep recurrent learning architecture is optimized by the collaborative search of three bees. In a sentiment classification task involving product reviews, movie reviews, and twitter, BA-3+ was tested to classify symmetric and asymmetric distributions. PSO and Differential Evolution (DE) algorithms have been compared in order to obtain comparable results for advanced deep language models. For the Turkish and English datasets, BA-3+ outperformed

the DE, PSO and Stochastic Gradient Descent (SGD) algorithms in terms of convergence to the global minimum. All classification datasets have seen improvements of 30–40% in accuracy value and F1 measure over standard SGD.

Aside from using evolutionary algorithms for improving ML methods, it is also essential to represent words correctly when processing text (reviews). As a result of word embeddings, is considered one of the best methods to represent the words in sentiment analysis.

Alharbi et al. [11] proposed a deep learning approach that will be evaluated in order to accurately predict customer opinion based on Amazon website mobile phone reviews. Analysis of these reviews is used to categorize them as positive, negative, or neutral. In this study, they implemented and evaluated a wide variety of deep learning algorithms, including four variants of simple RNN (Gated Recurrent Unit (GRNN), LSTM Networks (LRNN), Update Recurrent Unit (UGRNN), and Group LSTM Networks (GLRNN)). Several algorithms have been evaluated for sentiment analysis using word embedding as a feature extraction technique, such as FastText, Word2Vec, and Glove. For both balanced and unbalanced datasets, five different algorithms are evaluated based on F1 score, recall, accuracy, and precision. FastText feature extraction with GLRNN algorithms led to 93.75% accuracy for the unbalanced dataset. By using the LRNN algorithm, the highest accuracy was achieved with the balanced dataset at 88.39%. Another recent study [267] incorporated LSTM with word embedding, this paper proposes a new architecture for extracting semantic correlations between neighboring words. Further, key terms are extracted from the reviews using weighted self-attention. A word-embedding self-attention LSTM architecture based on the IMDB dataset was found to achieve an F1 score of 88.67% after an experimental analysis. An F1 score of 84.42% was achieved by the LSTM and 85.69% by the word embedding LSTM-based models, respectively.

A novel real-time sentiment (WRS) word embedding method is presented in [232]. The novelty of the WRS is its use of the Word2Vec approach to embed words into Word-to-Word Graphs (W2WGs). The WRS method uses the W2WG embedding method to achieve the word feature vector by assembling the different lexicon resources. In order to improve sentiment classification performance, robust neural networks integrate LSTMs and CNNs. CNN is used to extract the leading text features for sentiment classification by storing word sequence information in LSTM. IMDB and Twitter datasets are used for the experiments. Real-time sentiment classification results demonstrate the effectiveness of our proposed method. For sentiment word embedding learning, a new loss function is defined based on co-occurrence likelihoods and sentiment possibilities of words [272]. BESWE outperforms many state-of-the-art methods using sentiment lexicons and Movie Review datasets, namely, CBOW, DLJT1, SE-HyRank, C&W, and GloVe. Bayesian estimation is effective in capturing

the sentiment information of low-frequency words and integrating it into word embeddings through loss functions through sentiment analysis. BESWE can also significantly improve the performance of state-of-the-art sentiment analysis methods by replacing the embedding of low-frequency words with BESWE.

Occasionally the availability of labeled big data to perform word embedding is not possible, due to a lack of resources such as time, money, and effort. Therefore, the pre-trained word embedding can solve this dilemma. A pre-trained word embedding is a model that has already been trained on larger datasets with the same characteristics. By leveraging the trained embedding model, the model can be turned on the smaller dataset. This process is similar to transfer learning. Some researchers started to apply this method in the literature, for example, Kamyab et al. [132] proposed an attention-based model using CNNs and LSTMs is proposed here (ACL-SA). The first step is to enhance the data quality with a preprocessor and employ feature weighting provided by TF-IDF in combination with pre-trained Glove word embeddings. Additionally, it uses CNN's max-pooling to reduce feature dimensionality and extract contextual features. Furthermore, long-term dependencies are captured using a bidirectional LSTM integrated into the model. Moreover, it emphasizes the attention level of each word at the CNN's output layer. Four English standard datasets are used to evaluate the model's robustness, such as US-airline, SA4A, Sentiment140, and Sentiment140-MV. A variety of performance matrices and baseline models have been used to compare the efficiency of the model. Compared to the state-of-the-art models, the proposed method performs significantly better.

A novel sentiment analysis method called Improved Word Vectors (IWV) was proposed to improve pre-trained word embedding accuracy [238]. Word position algorithm, Part-of-Speech (POS) methods, lexicon, and Word2Vec/GloVe techniques are used in the approach. Different deep learning models and benchmark sentiment datasets were used to test the accuracy of our method. Using IWV for sentiment analysis is very effective, according to our results. Krouska et al. [154] presented research that explored big data analysis of tweets and used deep learning to classify their polarity by analyzing large numbers of tweets. In their paper, four well-known pre-trained word embedding methods have been used, including, Facebook's FastText, Stanford's Crawl GloVe, Google's Word2Vec, and Stanford's Twitter GloVe. Using deep learning, tweet classification outperforms baseline ML algorithms, according to one major conclusion. While FastText obtains better accuracy rates regardless of the dimensionality of the dataset, Twitter GloVe acquires great results notwithstanding its lower dimensionality.

An enhanced ensemble classifier framework is presented in [188]. Their approach is based on three methods, namely, bag-of-words, pre-trained word embeddings, and lexicon.

Afterward, each sentence is preprocessed using several techniques, including, lemmatization, stop-words removal, shortening, stemming, and POS tagging. Furthermore, 11 different classifiers are fed the previous feature vectors. Four datasets with five different lexicons are used to test and evaluate the proposed framework. The proposed approach achieved better results than their pervious model. Word embeddings were tested with fastText and GloVe [7]. Using pre-trained and self-trained word embedding vectors, we created eight test scenarios to assess the performance of the word embeddings. A stemmed dataset and an unstemmed dataset were prepared from Twitter data. The model that used self-trained GloVe and trained on an unstemmed dataset generated the highest accuracy from GloVe scenario group. A 92.5% accuracy was achieved. Alternatively, the most accurate fastText model generated by self-learning fastText and training on the stemmed dataset generated the highest accuracy from the fastText scenario group. Pre-trained embedding vectors resulted in lower accuracy than self-trained embedding vectors in other scenarios. Also, the dataset was sourced from Twitter, which contains non-standard sentences, rather than the Wikipedia corpus from which the embedding data was trained.

Using user reviews as a dataset, the study [144] proposed a multi-class Urdu language sentiment analysis. There are various domains represented in the dataset, including, software and apps, sports, movies and plays, politics, and food. This dataset consists of 9312 reviews that were manually annotated by experts and categorized into three categories: positive, negative, and neutral. Using rule-based, ML, and deep learning techniques, this study aims to develop a dataset for Urdu sentiment analysis. Moreover, Multilingual BERT(mBERT) has been fine-tuned to analyze Urdu sentiment. In order to train the classifiers, we used a combination of char n-grams, pre-trained BERT and fastText word embeddings, and n-grams. A total of two datasets were used for training and evaluating these models. According to the findings, a ranking of 81.49% was achieved by the proposed mBERT model with BERT pre-trained word embeddings.

Policy-making is one application domain in which public opinion analysis is a key component. A recently developed method for extracting subjective information from raw texts is sentiment analysis and Natural Language Processing. To analyze sentiments, it is imperative to exploit and correlate data from multiple languages in order to consider all available information holistically. Based on the translation of text for which neural sentiment analysis has already been applied in English. Using Neural Machine Translation for sentiment analysis is investigated in this paper [177]. The neural sentiment analysis is applied to translated German and Greek texts based on texts that have already been translated into English. A Neural Network model is used to translate the latter from the original language. A neural sentiment analysis was performed in order to analyze the difference between the

source language and the target language. There are two interpretations of the results of the proposed approach. The study concludes that current Neural Machine Translation tools can produce high-accuracy translations and only affect multilingual sentiment analysis in a minor way. Additionally, pre-trained Word embeddings are shown to be capable of providing multilingualism.

A new Sentiment analysis system, named EvoMSA, is presented in this work that unifies different Sentiment analysis systems [103]. Through the use of language-independent techniques, it becomes domain-independent and multilingual. A genetically programmed classifier called EvoMSA produces a predictive model by combining output from various text classifiers. In order to provide a global overview of EvoMSA's performance, we analyzed its performance across different Sentiment analysis competitions. As a result of these results, EvoMSA was ranked highest in several Sentiment analysis competitions. To facilitate a practitioner or newcomer to implement a competitive Sentiment analysis classifier, we also examined EvoMSA's components for their contribution to performance. Moreover, the authors of [265] demonstrate a method for classifying sentiments using deep learning in multilingual contexts. To train a classifier, they translate input text into English, embed the English words, and then train the classifier using embeddings. Multilingual sentiment analysis can be simplified and uniformly performed using this projection into English space and word embeddings. Adding polar words to these sentences with their polarities as well as their occurrences in the training data would be a novel approach. While most languages lack resources, this approach offers a 7-10% performance gain over traditional classifiers, regardless of text genre.

The purpose of this study is to investigate how sentiment embedding can be used to enhance the performance of sentiment prediction [203] First, they propose a new word embedding method that captures semantic sentiment details via a supervised method. In addition, Deep Learning models can be applied by combining different architectures, namely, CNN and RNN. Also, sentiment classification is improved through the development of attention mechanisms. The proposed method shows promising results when tested on benchmark datasets in different languages (English and Vietnamese).

Unlike these studies, our second approach in Chapter 6 for sentiment analysis is a swarm-based TwinSVM, which is a new and improved version of the SVM. Additionally, the work examines the multilingual context in reviews to understand the characteristics of different languages. To investigate the representation of words in multilingual environments, a variety of word embedding techniques have been applied (e.g. MUSE, BERT, and FastText). We employed pre-trained word embeddings as a solution to the big data requirement for word

embedding learning. As part of the study of review modifications, we analyzed the differences in review structure, context, style, and preference following the COVID-19 situation.

3.4 Spam Reviews Detection through Sentiment Analysis

This section focuses on the studies of spam reviews detection based on sentiment analysis. By leveraging sentiment analysis, researchers aim to enhance the accuracy of spam reviews detection and mitigate the impact of positives and negatives in identifying fake reviews.

Sentiment analysis analyzes the user's opinions and reviews about certain products or services, helping decision-makers make profitable decisions. The authors of [140] reviewed the various techniques and approaches for sentiment analysis and discussed the several applications of sentiment analysis.

A set of recent studies applied sentiment analysis as a mechanism to detect spam reviews. For example, [202] designed an opinion spam detection framework based on SA. The framework adopted the Persian language to identify the spam reviews and generate novel features related to that language characteristics. Using AdaBoost and Decision Tree the classification accuracy reached 98% and 98.6%, respectively. Besides, the newly created features were utilized and compared with other features set from the literature. Patil et al. [214] stated that opinion mining is generally performed for sentiment analysis recognition. However, not all data can be sure of its trustworthiness. As such, it's important to detect spam reviews. Therefore, this work presented a product-based sentiment analysis and a spam identification model for online reviews (ALOSI). Consequently, emphasize the dissimilarity of opinion sentiment with and without considering spam reviews.

The authors of [118] proposed a data-driven model for product recommendations based on preference profiling, reviewer credibility analysis, and fine-grained sentiment analysis. For avoiding spam reviews and reviewers, they used expertise, trust, and influence scores to determine the capabilities of reviewers and weight their reviews accordingly. Peng et al. [216] proposed an approach to detect spam reviews by computing the sentiment score by a shallow dependency parser and presented discriminative rules to detect spam reviews using a time series analysis method. Saumya S. and Singh1 JP. [249] considered the sentiments of the review and its comments, content-based factors, and rating deviation for suspicious reviews. Using the content of the comments in detecting spam reviews and solving the imbalanced problem of the spam minority class were the two main contributions of their work. They tested the approach with various classification techniques and achieved an F1-score value of 91%. [161] considered grouped spam instead of detecting individual frauds using aspect-oriented sentiment mining by grouping the reviews. Then, they used content duplication and

time bursts to detect cross-platform spam Chinese reviews using the K-Means clustering algorithm.

While, the study introduced by [255] examined several facets, which are language, rating sentiment, and content. Also, an extra investigation has been provided to explore the deception and behavior for detecting spam reviews. Above 20 features have been extracted to characterize reviews and multiple machine learning models evaluated on spam reviews detection. The outcome of this study suggests that there are discrepancies in reviews and have plosive impacts on the spam detection performance. Another recent framework devolved for the sake of detecting spam reviews using the opinion mining method [17]. Two machine learning models have been proposed, one for spam reviews detection and the other for knowing the rates of these reviews. The models were designed using random forest and Naïve Bayes techniques. Yelp dataset was considered in order to evaluate both models for the purpose of spam reviews detection and opinion mining.

Sentiment analysis is used in many areas, such as restaurants, movies, products, and hotels. [46] considered the fake reviews of hype about restaurants which causes misleading information to decision-makers. The authors analyzed the four dimensions that a review can include: taste, environment, service, and overall attitude, and accordingly, the review can be classified as the hype. Bayes classifier was applied by the classification task while generating a 74% accuracy. Another study in [181] considered a combination of emotions and sentiments, achieving better accuracy than each separately using three real-world datasets.

Online movie reviews are studied by [78] to detect spam reviews by comparing different classification algorithms, including NB, SVM, k-NN, KStar (K*), and DT. Based on the results, the SVM classification technique generated the best results compared to the other algorithms. Another study in [147] considered the detection of spam movie reviews. Ashwini MC. and Padma MC. [23] detected spam reviews for Amazon products using NB and RF classification algorithms. The work presented in [108] analyzed the sentiment distribution for spam reviews and the impact of probabilistic sentiment score using a sentiment model for hotel reviews. They used TF-IDF-based classification with an accuracy value of 92% for the classification task and a value of 89% for spam review detection. The authors of [240] proposed a system that can identify spam and ham tweets, and evaluate their emotional content. Following preprocessing of the tweets, the extracted features are classified using a variety of classifiers, including LR, SVM, RF, Multinomial and Bernoulli NB, and DT for spam detection. For sentiment analysis, SVM, RF, NB, LR, SGD, and deep learning methods. Analyses are conducted on the performance of each classifier. According to the results, the features extracted from the tweets can be successfully utilized to determine whether a

particular tweet is real or spam. In order to accomplish this, a learning model is developed that correlates tweets with particular sentiments.

For content-based applications, [264] proposed a sentiment-based spam detection mechanism. As a result of this technique, the sentiment score for each sentence is calculated. It is determined whether a review is fake or not based on its sentiment score. Additionally, the work presents a deep learning approach based on LSTM to detect fake reviews. In comparison with other existing models, the combination of these two approaches achieves a higher accuracy rate in the detection of opinion spam. Based on their model, they obtained an accuracy of 92.46%, with a false acceptance rate of 9.23% and a false rejection rate of 5.50%. The study of [263] applies sentiment analysis and NLP to identify fake reviews. Using preprocessing techniques, data is transformed into a format that can be analyzed and detected. They propose using gates, bidirectional LSTMs, and long short-term memories to analyze reviews and evaluate the results based on ReLu, TanH, and Sigmoid activation functions. By automatically analyzing reviews, the work in [139] generates quantitative scores based on negative and positive feedback. Amazon's online reviews were analyzed using sentiment analysis. By utilizing NLP, the Fake Review Detection Framework (FRDF) is able to detect and remove fake reviews. Reviews of high-tech products were conducted to test the FRDF. Consumer sentiment was used to rate brands. Combined with the 5-Star score, this tool would be useful for more comprehensive decision-making for brand managers and consumers. In the FRDF, for instance, price is ranked alongside sentiment value and the 5-star rating of the best products.

In certain scenarios of sentiment analysis applications, knowing the reviews' reliability is considered crucial for different parties. Various methods have been employed to combine the two methods as seen above. Such studies achieved another level or additional aspect of just knowing the sentiment of the reviews but also comprehending if ts spam or not. As a result, more understanding of the type of reviews and their properties in various cases. Nevertheless, more studies are needed on the detection of spam reviews using sentiment analysis techniques.

Our third approach in Chapter 7 is different from the aforementioned studies by including feature selection and parameter tuning using metaheuristic algorithms. In addition, using the recent HHO algorithm [109] for optimization is not found in other works, as far as we know. Additionally, we applied sentiment analysis in order to improve spam reviews detection using TwinSVM, as well as pre-trained word embedding in a multilingual environment was also employed for the first time. Finally, analyze the effects of COVID-19 on the reviews.

3.5 TwinSVM

In this section, studies on various topics employing TwinSVM are presented, aiming to demonstrate the effectiveness of TwinSVM within the literature.

TwinSVM works on two smaller quadratic programming problems than the standard SVM method [89] with two non-parallel hyperplanes instead of one [129]. Various variations to the TwinSVM can be found in the literature, including Least Squares TwinSVM (LSTwinSVM), structural risk minimization version of TwinSVM (TBSVM) [257], a robust version of TwinSVM (RTWSVM) [225], wavelet TwinSVM [69], multi-label TwinSVM (MLTwinSVM) [49], and others [50, 256, 48, 287, 226].

Many applications use TwinSVM in the literature. [306] considered the Hydrological time series forecasting as an application to applying TwinSVM using a cooperation search algorithm for parameter identification. Computer vision is also considered by [180] by applying Multi-task v-twin SVM. Moreover, the work of [243] used TwinSVM based on the kernel for the Pancreatic Cancer Early Detection problem. Another work by [274] considered the application of Diagnosis of Parkinson's Disease Based TwinSVM for Feature Selection. The Wind Turbine Gearbox Anomaly Detection using Adaptive Threshold and TwinSVM is proposed by [67]. The work of [129] considered various datasets in pathology, vehicle, engineering, biological information, finance, etc., of different sizes and imbalance ratios. Finally, [50] considered datasets from the UCI ML repository representing a wide range of applications, including pathology, bioinformatics, finance, and so on.

Many studies developed their ML approaches using TwinSVM with metaheuristic algorithms for parameter tuning of TwinSVM algorithm. [70] applied Fruit Fly Optimization (FOA) algorithm to the TwinSVM (FOA-TWSVM) to optimize the values of TwinSVM. They have achieved better Accuracy results compared to Proximal SVM (PSVM) and Generalized Eigenvalues PSVM (GEPSVM). Other studies exist in the literature which use the PSO with Gravitational Search Algorithm (PSOGSA) [114] and Chemical Reaction Optimization (CRO) algorithm [302] for parameters' optimization.

A variation of TwinSVM is named a Least Squares TwinSVM (LSTwinSVM), which depends on non-parallel twin hyperplanes. [248] applied simulated annealing to LSTwinSVM to tune the algorithm's parameters. They experimented with various datasets and obtained competitive Accuracy results compared to LSTwinSVM, TwinSVM, GEPSVM, PSVM, SVM, and C4.5 classification algorithms. Another variation of TwinSVM combines wavelet analysis techniques with the TwinSVM (WTWSVM). [69] combined glowworm swarm optimization with wavelet TwinSVM to optimize the algorithm's parameters using Accuracy as a fitness function. They compared their results with WTWSVM, FOA-TwinSVM, TwinSVM,

and SVM algorithms. They have achieved the best results for most of the selected datasets compared to the other algorithms.

Different applications were found in the literature combining the metaheuristic algorithms with the TwinSVM algorithm for parameter tuning. Wind Speed Forecasting is considered in the work of [113] by using PSO metaheuristic algorithm, while Kewen Li et al. [164] used Grey Wolf Optimizer (GWO) for the favorable reservoir of the oil field. In addition, the Artificial Fish Swarm Algorithm (AFSA) is used by [98] for flame recognition.

Chapter 4

Dataset

This chapter presents a detailed description of different topics related to the spam reviews dataset, such as datasets in the literature, multilingual motivations, word representation methods, and the proposed datasets.

The following points summarize the objectives of this chapter:

- Discuss and outline the spam review detection datasets in the literature.
- Examine the motivations for choosing English, Spanish, and Arabic as the primary languages in a multilingual context.
- Describe the characteristics of the proposed datasets and the preparation process.

4.1 Datasets in the Literature

A dataset in general is a collection of data that can be found in every research domain, for example, security, medical, business, geoscience, etc. These datasets have various forms based on their structure and properties. The structure of the dataset is usually what determines the research problem, namely, supervised, semi-supervised, and unsupervised learning. Data is considered a significant part of machine learning models that, normally, require several procedures to be generated. Such procedures range from collecting, cleaning, pre-processing, and formatting to labeling.

Regarding spam reviews datasets, there are three types used commonly in the literature, content, meta-data, and product information. Each of which has its own advantage and disadvantage traits [110]. The **content-type** data, for instance, consists of the review textual features. That is to say, it is a linguistic feature extracted from the review content. These

features can be Part-of-speech (POS) n-grams or words extracted for the purpose of identifying reviews' origin. Despite the fact that this type has a significant role in detecting spam reviews. The methods used are not sufficient to distinguish all kinds of spam reviews.

Further, the second type (**meta-data**) is more relevant for the review details besides its actual content, such as the identity of the reviewer, IP and MAC address (after being anonymized), rating, time of the review, geolocation place and review writing time duration. By examining such data, certain malicious behaviors can be identified. For example, when various user-ids post a number of negative and positive reviews of a certain product using the same device it shows suspicious behavior. Also, some reviewers write positive reviews for a particular brand and at the same time, negative reviews for other brands might consider doubtful reviews. Another example is where reviews for a specific hotel were found to be near its location; these reviews are obviously not trustworthy because reviewers should be in other places. This type is considered effective for spam reviews detection, however, the number of features is limited.

As for **product information** datasets, this type takes into account the product sales number and description to detect spam reviews. An example of such type could be: when a low-sales product has too many positive reviews demonstrating the reliability of the reviews. Besides, there are some datasets that combine two or three types together in order to increase the performance of the detection. More experts can distinguish spam reviews due to an increase in employed features.

Creating and generating spam review datasets is considered a complex task. Few researchers have adopted the new alternative method for labeling and collecting artificial reviews. By creating synthetic spam reviews datasets and taking existing real reviews to build their model. An example of the context of human labeling of spam and real reviews is available at [246]. Table 4.1 shows the sample of reviews and their labels.

Several numbers of the proposed spam reviews datasets in the literature are summarized in Table 4.2. Also, the distribution of the sources of the datasets can be found in Figure 4.1.

Table 4.1 A sample of the reviews and their labels from [246] dataset.

Text	Label
Alright for the price, but was a little disappointed with the quality of the product.	Fake
Excellent product and a much better quality than the one you get at Walmart for \$50.	Fake
Excellent quality product. Perfect for my ccozinha.	Real
Got these for the third time. I have a small dog and my son loves to throw them	Fake
Great cup. Will last forever. Keeps things cool / warm.	Real
Great quality and the fabric looks great. It is thick enough to hold a small amount of coffee	Fake
Great quality, beautiful color. We have had the same one for a few months now and love	Fake
Great stickers but a little small. The only reason I gave it 4 stars is because the one in	Fake
I put paper in one corner of the cabinet and put a piece of paper in the other corner.	Fake
It is nice bowl and have had a fast shipping!	Real
Like this little guy. Use it often. He is small.	Real
Love them, just thought they would be a bit bigger!	Real
Love this movie \& the time it took to finish. I will keep my review for the next time	Fake
Perfect. They do exactly what I need them to do. I will keep them for a long time	Fake
Save \$ on these instead of the original shipping box. I will keep my money!Very pretty.	Fake
Such a great purchase can't beat it for the price	Real
Super rough, not soft wash cloths, more like bar towels	Real
These are just perfect, exactly what I was looking for.	Real
These are so nice, sturdy, like the color choices too.	Real
These towels are real nice, and I love the feel of them. They have a nice feel	Fake
This fan is really pretty and I actually use it.	Real
What can you say— cheap and it works as intended.	Real
Will not seal properly. Atrocious little ones, they feel like they will break if you throw	Fake
Works great, easy to clean, and easy to clean. I will be purchasing more!Very pretty	Fake

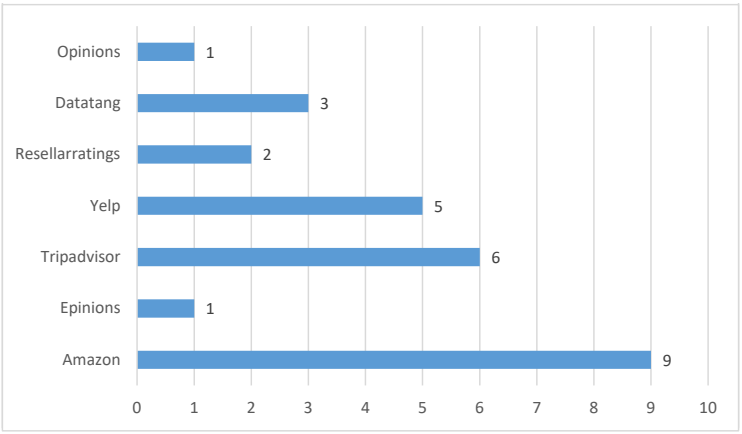


Fig. 4.1 Number of published datasets for each website (source).

Table 4.2 Descriptions of spam reviews datasets in literature.

#	Source	Volume	Reference
1	Amazon	5.8 million	[130]
2	Epinions	6000	[159]
3	Tripadvisor	3032	[162]
4	Yelp	-	[195]
5	Tripadvisor	1600	[210]
6	Tripadvisor	2848	[1]
7	Tripadvisor	27 952	[87]
8	Resellarratings	628 707	[215]
9	Resellarratings	408 470	[286]
10	Datatang	10 020	[128]
11	Datatang	493 982	[119]
12	Amazon	65 098	[288]
13	Amazon	109 518	[194]
14	Amazon	195 174	[173]
15	Amazon	3 million	[24]
16	Amazon	542 085	[53]
17	Amazon	6819	[93]
18	Tripadvisor	3000	[236]
19	Yelp	67395	[195]
20	Yelp	359052	[160]
21	Yelp	608598	[160]
22	Datatang	9765	[235]
23	Tripadvisor	800	[211]
24	Amazon	-	[38]
25	Amazon	142,8 million	[204]
26	Yelp	18912	[31]
27	Opinions	6000	[159]

4.2 Multilingual Motivations

With the evolution of technology, such as social networks. It started to attract all kinds of people from different regions and ethnics. Hence, with the advent of the Internet, which provides equal access to information for all individuals, our national boundaries are increasingly blurring. As such, it is imperative that our representations are not only free from biases related to gender or race but also geared toward biases related to language in a globalized society. Therefore, multilingual reviews begin to attract the attention of researchers. As it can be seen in this survey [5], the number of papers that tackle this problem has increased. One example of that is the work in [217] which addressed the novel task of clustering multilingual aspects from reviews written in different languages, with the goal of organizing semanti-

cally related aspects. By using word embeddings [156] (vector representation of words technique), they represent the contextual information of the aspects in their unsupervised method. Additionally, [297] used a single classification model in multiple languages, and they aimed to construct a universal sentiment classifier. They proposed a learning word embedding approach that accounts for sentiment across multiple languages simultaneously, based exclusively on labeled reviews in English and unlabeled data in other languages.

Handling reviews in different languages can benefit businesses and customers from other regions. For example, understanding and addressing reviews written in multiple languages can improve customer satisfaction and solve customer concerns. Further, reviews in multiple languages allow businesses to reach a wider audience and give them a better understanding of non-native speakers' experiences. Thus, businesses can increase their revenue and improve their localization efforts, which allows them to adapt their products and services to the needs of various regions and cultures.

The multilingual context should be taken into consideration when dealing with issues related to sentiment or spam reviews. As a result, the generated dataset will be more diverse and contain a wide range of writing styles, accents, and dialects. Also, training on multiple-language data can enhance the generalization performance of the model, since it will be exposed to various language structures and variations. The identification of spam reviews and sentiment analysis in a multilingual context could address the issue similarly to that found in real-world problems. Furthermore, ensure that the model can be used in a wide range of languages and contexts, and demonstrate its versatility and applicability. By doing so, you will be able to ensure that the designed model is robust as well as applicable to a wide range of customers.

In this thesis, we focused on extracting reviews in three languages: English, Spanish, and Arabic. These specific languages were selected for several reasons, including:

- Widely spoken languages: There are billions of people throughout the world who speak English, Spanish, and Arabic. According to Statista, the three languages are in the top 5 of the most spoken in the world [90]. By concentrating on these languages, a large audience can be reached and a substantial impact can be made.

- Rich history of written communication: Written communication has a long history in English, Spanish, and Arabic, with many texts and documents available in these languages. Thus, a wealth of data can be gathered for detecting spam reviews or identifying sentiment analysis.

- Widely used on the Internet: Internet users frequently use English, Spanish, and Arabic, with hundreds of social media platforms, websites, and other online resources. This can facilitate the collection of data for the detection of spam reviews.
- Linguistically diverse: The English, Spanish, and Arabic languages are linguistically diverse and feature a variety of accents, dialects, and writing styles (e,g, writing from right to left or vice versa). This can make identifying spam reviews and sentiment analysis in these languages more challenging, however, it also provides training data that is rich and diverse.

Overall, using multilingual context in spam reviews and sentiment analysis is a very challenging and complex task. For instance, challenges such as language diversity, translation issues, cultural sensitivity, code-switching, domain-specific complexities, and resource intensity may arise, requiring meticulous handling. However, doing so can help to maintain the credibility of reviews around the world.

4.3 Word Representation

Word representation is a fundamental aspect of NLP and machine learning. It refers to the method by which words are encoded and represented in a computational system. Common techniques for word representation include one-hot encoding, in which each word is represented by a binary vector with a single non-zero entry, or word embeddings, which represent words as dense, continuous-valued vectors in a high-dimensional space. The choice of a word representation method can have a significant impact on the performance of NLP and machine learning models. In the following subsection, different Word representation techniques will be explained.

4.3.1 Word Embeddings

In general, embeddings depict words as raw vectors composed of real values, including documents and chunks of text. It is possible to classify embedding models as predictive or frequency-based. Essentially, word embedding involves mapping semantic properties into implicit feature vectors, which are described by real-number vectors. Thus, each latent feature of a word is represented in a different dimension by its embedding. Textual data encoded in a string format is typically not-well processed by machine learning algorithms. For this reason, it is necessary to transform (encode) the data into a suitable format in order for the algorithm to be able to process it.

Various techniques have been proposed in the literature for representing text, including Frequency–inverse Document Frequency (e.g. frequency-based method) or Word2Vec, Glove, BERT, MUSE, and FastText (e.g. prediction-based methods). By applying shallow neural networks, embedding vectors can be created by using hidden layers within these networks. In texts, implicit meanings can be embedded (for example, questions) in order to produce smarter features. Consequently, neural networks should be able to tune the embedded vectors in such a way that they have similar meanings for similar words. Thus, a similar word in a related context with the same vector representation is likely to have a high similarity score.

Several techniques are available for embedding words. In this study, the following techniques will be applied, since they are able to deal with multilingual contexts:

- Bidirectional Encoder Representations from Transformers (BERT): In BERT, the attention mechanism was used in conjunction with the transformer methodology [66]. Utilizing the attention mechanism, one can ascertain the interrelation among words within a sentence. Consequently, it is crucial to acknowledge the diverse contexts in which a specific word might be employed. This leads to the understanding that certain words can possess multiple meanings, contingent upon the context of their usage [218].

  A further benefit of BERT is that it performs tokenization to create word pieces, such that the word 'run' can be broken down into two parts, run and ****ing. The tokenization process allows the word to be divided into parts if it is not included in the BERT vocabulary. Additionally, BERT word embedding can also be applied to multilingual data. Further, for example, in a sentence like "I love to eat fish, but I'm not a fan of salmon," BERT will capture the nuanced meaning by taking into account both left and right contexts, providing rich embeddings for each word.

- FastText: is considered a variation of Mikolov's approach. Utilizing computationally efficient algorithms, FastText embeds words, and the representation of distributed words is taken into consideration using different procedures [182]. It performs the same functions as the Continuous Bag Of Words (CBoW) model, but instead of a bag of words, a bag of n-grams and middle words are also taken into account. Thus, it will be able to capture some information regarding the order of words. The algorithm can be implemented in supervised or unsupervised modes. By computing the embedding mean of the related words, the documents in supervised mode are converted into vectors. After training the linear classifiers, the algorithm uses these vectors' values as input to train the hierarchical softmax function. However, FastText's unsupervised mode does not consider classes when creating word embeddings. Hence, using the n-grams technique, it is possible to handle many different languages and dialects.

Moreover, in sentences such as "embeddings capture semantic meanings", even with minimal training data, the FastText method can represent words like "semantic" based on its subword information.

- Multilingual Unsupervised and Supervised Embedding (MUSE): is one of the most widely used methods for embedding words in multiple languages [57]. For learning the embeddings of different languages, this method uses supervised machine-learning techniques that were developed by Facebook. In other words, it can generate embeddings in a variety of languages. Through the use of parallel sentences, MUSE is able to map words in multiple languages to a similar embedding space. It can produce embeddings that are cross-language comparable for tasks such as multilingual text classification and machine translation. Generally, the MUSE algorithm is capable of generating multiple language word embeddings and has shown impressive performance on many NLP tasks. So, a word like "apple" in English corresponding to "manzana" in Spanish can have similar representations in the vector space. This can facilitate seamless language transfer and understanding of different linguistic contexts.

Pre-trained word embedding

Pre-trained word embeddings are models of word representation that have already learned from large datasets. In the domain of natural language processing, tasks such as sentiment analysis and text classification benefit significantly from the utilization of pre-trained word embeddings. These pre-trained embeddings encompass words that have been previously converted into numerical vectors, facilitating the interpretation of their meanings. Consequently, these vectors serve as suitable input for machine learning models. Employing pre-trained word embeddings in natural language processing offers multiple advantages, some of which are outlined below as principal benefits.

- Time and resource savings: Models that are pre-trained on large datasets require significant computational resources and time to train. The use of pre-trained models can save researchers time and resources by avoiding the need to train a new word embedding model from scratch.

- Improved performance: pre-trained embedding models developed from trained datasets often capture rich syntactic and semantic information. Through this approach, subsequent tasks, such as linguistic translation and textual categorization, can be executed effectively due to the model's pre-existing familiarity with language structures and patterns.

- Access to specialized knowledge: It is common that pre-trained embedding models are developed based on specific domains or languages, such as medical texts or Twitter data. Researchers working in these areas may find this informative. The pre-trained model has already acquired language patterns and specific vocabulary relevant to that field.

In summary, utilizing pre-trained word embeddings offers numerous benefits compared to developing a word embedding model from scratch. Consequently, this approach serves as an efficacious instrument in the domain of natural language processing.

4.4 Proposed Datasets

In this subsection, the labeling criteria and the description, characteristics, and preparation of the proposed datasets are discussed.

4.4.1 Labeling Criteria

In this thesis, the labeling process was manual. Thus, this involves three experts who review and annotate the data for two distinct aspects: spam review detection and sentiment analysis. The experts are tasked with reviewing the data and assigning labels accordingly. The sentiment analysis annotation is relatively straightforward, however, the spam review annotation process requires the experts to consider certain guidelines before labeling the data. These guidelines are listed as follows:

- The review contains keywords or tags that are not relevant to the content and are often associated with spam, such as "free," "discount," or "sale."
- The review has been written by a user who has submitted an excessive amount of reviews within a short time frame, which could imply that the user is being paid to write spam reviews.
- The review includes personal information, like a phone number or email address, which can be used for spamming or other malicious activities.
- The review contains inappropriate or offensive content.
- The review is in a repetitive or senseless style, which may suggest that it was generated by a spam bot.

- The review contains multiple mentions of other products or services, which could imply that the user is attempting to advertise those products or services.
- The review has a significant number of emoticons or special characters, which may be used to conceal the fact that the review is spam.
- The review has many formatting mistakes or other visual peculiarities, which could be an indication that the review was generated by a spam bot.
- The review has a large number of hyperlinks or other external references that may be utilized to advertise spam websites or products.

Overall, these guidelines are employed as a strategy to assist experts in optimizing the effectiveness of the labeling process. They provide a valuable aid in recognizing and assessing distinct characteristics in reviews that may signify spam.

4.4.2 Dataset Description, Characteristics and Preparation

This subsection describes the data collection process, the characteristics of the data, and outlines the preparation steps. This thesis presents data on a variety of products and services that are currently available in a variety of languages. In order to collect the data, Twitter was utilized as it has the largest community of users who write reviews online. In comparison to other review websites, online social networks have the advantage of being easy to use and containing more users. Therefore, it can be used to influence a wide audience concerning a wide range of topics and issues, especially with regard to feedback related to products and services.

Using the API and Token information of the Twitter account allows extraction, the Rtweet package was applied to collect reviews in RStudio. Due to the fact that different languages were employed during the collection process, a particular language is specified for each search. Further, a term-based approach was used to collect reviews (see code below). During the collection, three languages were targeted: English, Spanish, and Arabic. As each language has specific requirements based on its characteristics, it was separated into different text files. Meanwhile, the original text was stored in CSV files for the labeling process.

```
tweets <- search_tweets("#term", lang = "es", n = 1000)
```

Taking language into account will aid in understanding product feedback from various regions. Consequently, different decisions will be made and different solutions will be offered depending on local conditions. Over 12,500 reviews have been collected from a variety of categories including entertainment, cosmetics, food, medical, and other products and services.

The collection of data took place from 2020 to 2023. Table 4.3 displays the details of the datasets.

Table 4.3 Dataset details

Data	Instances	Spam classes		Sentiment classes		
		Spam	Real	Postive	Neutral	Negative
Arabic	612	290	322	393	88	131
English	9900	6258	3642	2942	4393	2565
Spanish	2001	996	1005	1222	144	635
Multi	12513	7544	4969	4557	4625	3331

As soon as the data have been compiled into various CSV files, the labeling process will begin. Initially, each file will be preserved in its original context, but the text for labeling will be translated. Upon creation of the labels, the translated data will be archived, and the labels will be added to the original document later on. The reviews are labeled as Spam and Real for the spam reviews version and Positive, neutral, and negative for the sentiment version.

Moreover, the original file will be formatted, cleaned, and preprocessed in various ways. Thus, the data will be prepared for further analysis or modeling. The following steps are involved in the preparation of the dataset:

- Dataset Investigation: Begin by examining the dataset in order to gain a full understanding of its contents and structure. Basically, ensure that the data is free of noise, stop words, and duplicates.
- Removing irrelevant or noise: It may be necessary to remove special characters, HTML tags, numbers, punctuation, or other information that is irrelevant to the analysis or is likely to confuse the model.
- Normalizing the text: To make the text more consistent and easier to process, this may involve converting all text to lowercase, removing diacritics, or performing other normalization tasks.
- Remove duplicates: For the model to be accurate, duplicate rows should be removed from the dataset. A unique identifier can be used to identify and remove duplicate rows or by utilizing deduplication software.
- Remove stop words: It is usually recommended to exclude such words from text because they provide no meaningful information or may lead to inaccurate analyses.

Also, removing these words will help in decreasing the size of text data, prevent misleading results, and improve the interpretability of results. Due to the distinction between languages, the stop words for each language will differ. Table 4.4 provides examples of stop words for each language.

Table 4.4 Stop Words Examples for English, Spanish, and Arabic languages.

Language	Stop Words
English	a, an, and, the, in, of
Spanish	un, una, unos, unas, y, e
Arabic	و، في، من، إلى، كان، هو

- Stemming: Different methods have been used for stemming each language. This will result in a reduction in unnecessary features and an improvement in the selection phase. After a number of tests, the Snowball stemming [220] method was adopted for English and Spanish, while Arabic Light stemming [133] was selected for Arabic.

- Tokenize the text data: In tokenization, the text is broken up into smaller units referred to as tokens. An individual token may consist of a word, phrase, symbol, or other elements within the text.

Finally, the tokenized text data is converted into numerical vectors that can be fed as inputs to a machine-learning algorithm (vectorization). An effective method of accomplishing this is to use a pre-trained word embedding model, which was previously trained on a large text corpus and can be employed to map words or phrases into numerical representations. A pre-trained word embedding was performed due to the volume of the data and the fact that it contains multiple languages. The methods used in this study include MUSE, FastText, and BERT, all of which have multilingual embeddings. Further, each language has different pre-trained word embeddings generated by the three methods. It is important to note that all pre-trained word embeddings include a certain number of dimensions. Once the words in the dataset have been mapped to the pre-trained word embeddings, the embedded data can be used as input to our model. Consequently, eight versions of each dataset were generated for comparison, with 100 and 400 dimensions for each language. The dimensions were selected based on the predetermined dimensions of the pre-trained models. Utilizing these specific values aids in the examination of both higher-dimensional and lower-dimensional embeddings.

Chapter 5

Approach One: Spam Reviews Detection

5.1 Introduction

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. These approaches exhibit excellent performance compared to other techniques in the literature. Therefore, in this chapter, 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 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 [45]. 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 [106]. Incorporating this technique would significantly improve the performance of machine learning models by extracting useful features. One of these techniques is Word Embedding (WE) [167], 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 [167]. According to [131], 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.

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 [176, 285, 276], and [86]. In [176], 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 [285] they used Twitter's online social network to collect the reviews. Additionally, they applied a Support Vector Machine (SVM) in order to determine the spam reviews.

The previously mentioned approaches achieve good results in detecting spam reviews. However, they fail 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 by using metaheuristic algorithms. For instance, [213] presented clustering spiral cuckoo search methods for spam detection. [228] 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 [155], 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 [229].

However, all the aforementioned studies lack several aspects that this study aims to address. These aspects include the utilization of an advanced classifier to optimize its 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, this chapter presents a hybridization between Weighted SVM and Harris Hawk Optimization (HHO) algorithm for spam review detection. The HHO worked as an optimization algorithm for the hyperparameters and feature weighting. Three different languages 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 Words

(One-hot encoding)). Four experiments have been conducted, each of which focused on solving and demonstrating an issue (experiment) in terms of accuracy. The four experiments are a comparison of well-known classification algorithms, with various metaheuristics, a different version of SVM combined with HHO, and finally well-known classifiers combined with various metaheuristic algorithms. Further, a deep analysis has been conducted to investigate the context of reviews before and after the COVID-19 situation due to its effect on the users' writing style and preferences. Finally, generate a new dataset with statistical features and merge it with textual features to improve the detection performance.

In this regard, the proposed approach in this chapter can be summarized as follows:

- A WSVM combined with an HHO for hyperparameters optimization and feature weighting to enhance the overall performance.
- Spam reviews datasets have been generated, from the one mentioned in Chapter 4, taking into account two main aspects: the multilingual environment and the COVID-19 pandemic given the changing of review style and context. 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 to analyze different methods and seek the optimal solution to our problem.
- A deep analysis of the reviews before and after the COVID-19 pandemic has been conducted in order to study the new pattern of reviews. To put it another way, analyze how the reviews' structure, style, and content have changed.

5.2 Approach one Methodology

In this section, two stages will be described to design the methodology phase of our study. These stages are the Proposed approach and Evaluation measures. To enhance comprehension, numerical markers have been added to the method figure (Figure 5.1) to clarify specific descriptions within the points. These markers appear as ***{1}***, ***{2}***, ***{3}***, etc., and they do not need to adhere to a particular order; they serve merely as a mapping mechanism between text and the methodology figure.

5.2.1 Proposed approach (HHO-WSVM)

Improving the SVM performance has always been considered a complex problem in the literature. Accordingly, in this study, metaheuristic algorithms (e.g. HHO algorithm) have

been used to tune SVM parameters and feature weighting simultaneously. Therefore, 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.

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 γ (Gamma) parameters, while the second issue focuses on weighting the data features.

As a result, HHO produces elements for both the parameters (C and γ) 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 γ 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 γ, respectively. To perform this transformation, min-max normalization is applied, as shown in Eq.5.1. 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 ***{1}***. 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.

$$B = \frac{A - min_A}{max_A - min_A}(max_B - min_B) + min_B \tag{5.1}$$

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 ***{2}***. Therefore, it can be calculated as follows:

$$fitness(I_i^t) = \frac{1}{k}\sum_{k=1}^{k}\frac{1}{N}\sum_{j=1}^{N}\delta(c(x_j), y_j) \tag{5.2}$$

where I_i^t is the fitness, $c(x_j)$ represents the accuracy of the jth 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 δ, 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).

System architecture

Every dataset is divided into training and testing sets as part of our proposed approach ***{3}***. 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 use 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, γ, and the feature weights ***{4}***. 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 ***{2}***. In our case, we repeat the aforementioned steps until the HHO termination criterion is satisfied (number of iterations) ***{5}***. The best individual is generated by HHO during the testing phase when the maximum number of iterations is reached ***{6}***. 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 5.1.

5.2.2 Evaluation Metrics

As part of this stage, we will evaluate the predictive performance of the best-obtained model. 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 accuracy, which represents the machine learning model's performance in a tabular form as shown in Table 5.1. The confusion matrix will also be utilized in the upcoming chapters, specifically in chapters 6 and 7. Spam refers to the positive classes, while real refers to the negative classes.

Table 5.1 Confusion matrix

	Predicted Positive	Predicted Negative
Actual Positive	TP	FN
Actual Negative	FP	TN

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

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{5.3}$$

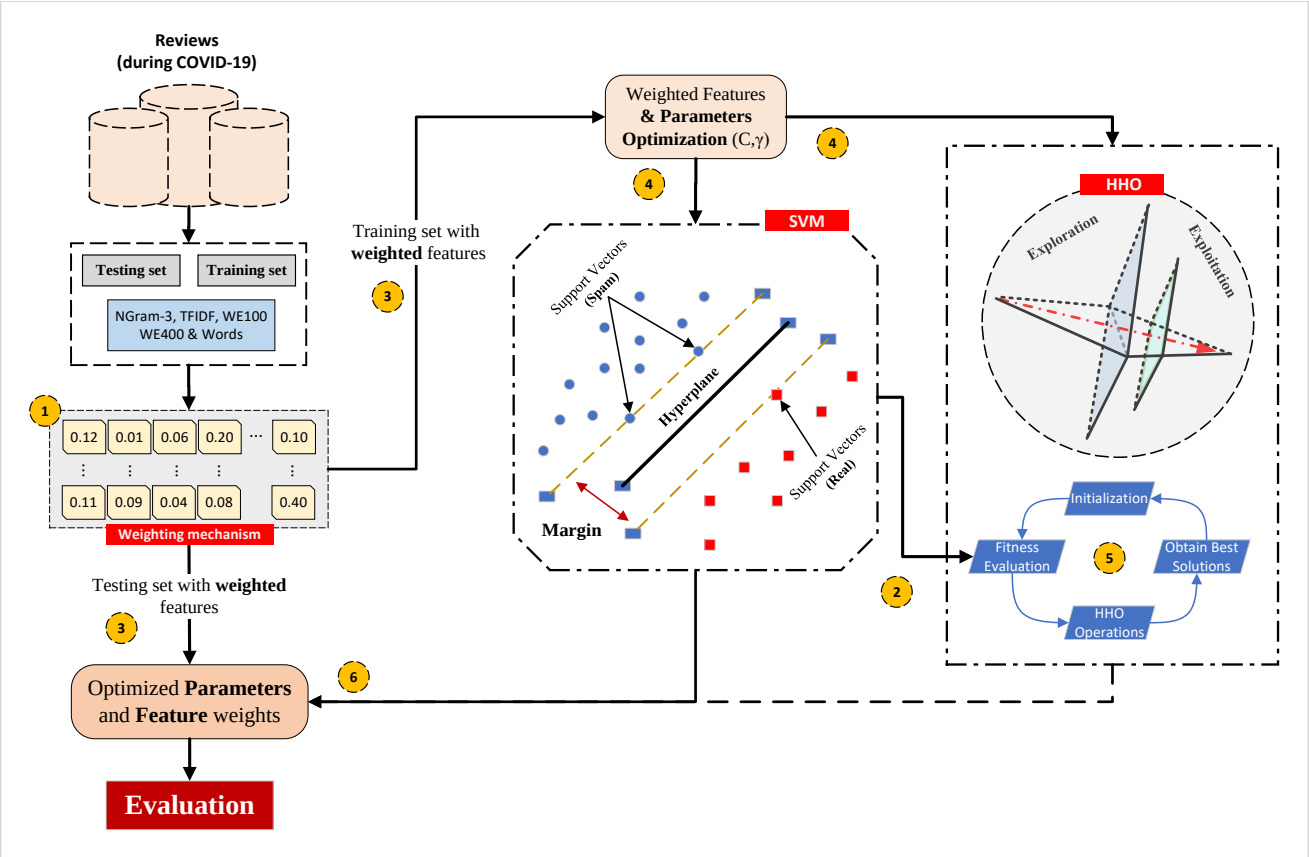


Fig. 5.1 A schematic diagram illustrates the flow of the proposed approach. The numerical annotations in the figure serve as a mapping mechanism between the text and the methodology figure; they are not arranged in any particular order.

5.3 Approach one Experiment 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 software was utilized to run all the algorithms. Additionally, 10-fold cross-validation was employed for both training and testing sets. The number of iterations used in this approach was 10 and was chosen after several experiments. The focus here was to determine the best overall method, without a primary focus on maximizing optimization due to the limited resources at first of the study.

In this first approach, the labels used for the data mentioned in Chapter 4 are 'spam' or 'real'. Then after the preparation phase, the pre-trained word embedding (BERT) was applied. 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). Furthermore, the following experiments were conducted with this approach:

- 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 new 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 HHO-WSVM 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 HHO-WSVM 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 HHO-WSVM 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 to improve performance.

The importance of selecting appropriate parameters has a significant impact on the efficiency of the model [296]. Further, the parameter settings of the used algorithms can be found in Table 5.2. Moreover, it is worth noting that the algorithm parameters were obtained from the original papers of the algorithms.

Table 5.2 Initial parameters of the optimization metaheuristic algorithms.

Algorithm	Parameter	Value
All	Iterations	10
MVO	Minimum wormhole existence ratio	0.2
	Maximum wormhole existence ratio	1
GA	Crossover probabilities	0.9
	Mutation probabilities	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]

5.3.1 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 5.3 presents the results of the first language data, which is the Arabic dataset, considering five versions of the dataset. As it 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, primarily because the J84 required a higher number of instances to perform well.

Table 5.3 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

In Table 5.4, 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 WE version (WE400) was used.

Table 5.4 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

Regarding the Spanish dataset, the results of the five versions are shown in Table 5.5. As illustrated, the highest accuracy was achieved by J48, followed by k-NN and NB, 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 5.6 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 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

Table 5.5 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

Table 5.6 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

the importance of employing different word representation methods for each dataset, rather than relying on a single technique. Consequently, the WE proved to be the most effective method.

5.3.2 Experiment II: Comparison of the proposed approach (HHO-WSVM) 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 5.2. All experiments were conducted 10 times to ensure a more robust model, and the average with standard deviation was taken into account. The *standard deviation* is employed in order to measure the spread or dispersion of data points, offering insights into the variability and distribution of values around the mean.

Table 5.7 illustrates the results of the Arabic data and its versions. The HHO-WSVM 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.

Table 5.7 Results of different metaheuristic algorithms and WSVM for all datasets

Versions	MVO-WSVM		GA-WSVM		PSO-WSVM		GOA-WSVM		SSA-WSVM		HHO-WSVM	
	Avg	Std Dev	Avg	Std Dev	Avg	Std Dev	Avg	Std Dev	Avg	Std Dev	Avg	Std Dev
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

For the English data, Table 5.7 presents the results for the data and its versions. The highest accuracy was achieved by HHO-WSVM 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 5.7. Furthermore, HHO-WSVM 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 5.7. The best result was achieved by HHO-WSVM for the WE400 version, followed by MVO-WSVM, PSO-WSVM, SSA-WSVM, GOA-WSVM, and GA-WSVM, respectively. The table also highlights that HHO-WSVM is the algorithm that consistently achieves the best results for most versions.

Figures 5.2, 5.3, 5.4 and 5.5 illustrate the convergence curve results for all algorithms. It's worth noting that in this comparison, utilizing only 10 iterations to identify the best

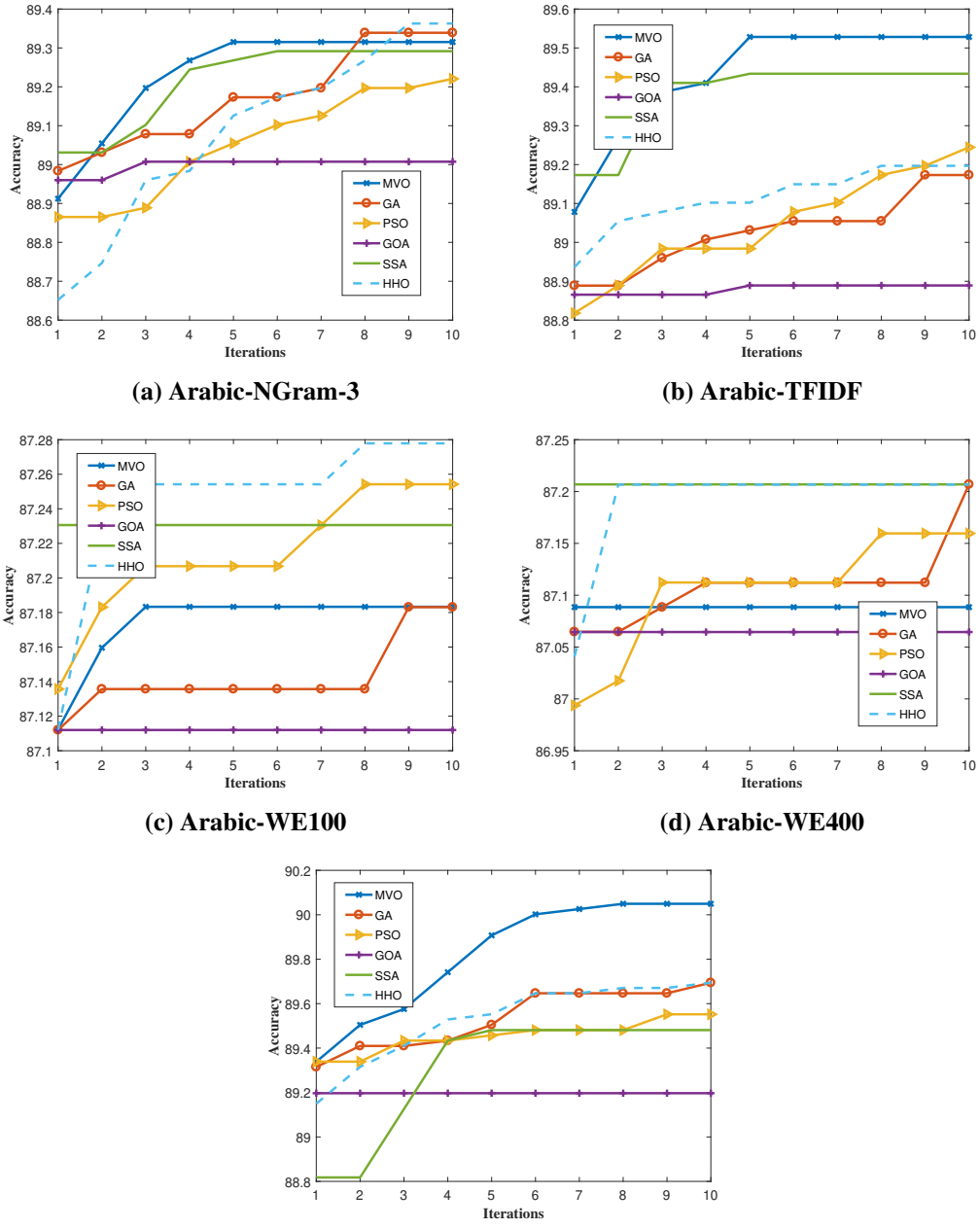
overall method, without an emphasis on maximizing optimization. This approach allows for a broader exploration of methods without prioritizing fine-tuned adjustments. In most datasets, HHO-WSVM exhibits the best convergence. More specifically, in the Arabic dataset, HHO-WSVM demonstrates robust convergence in Arabic-NGram-3, Arabic-WE100, and Arabic-WE400, while exhibiting competitive convergence in Arabic-TFIDF and Arabic-Words. In the English dataset, English-NGram-3, English-TFIDF, and English-WE100 exhibit outstanding convergence, with the exception of English-WE400 and English-Words, which achieve commendable competitive convergence. Regarding the Spanish dataset, convergence is notably good for Spanish-NGram, Spanish-TFIDF, and Spanish-WE400, with competitive values observed for Spanish-WE100 and Spanish-Words. Finally, the Multi dataset mirrors similar trends as the other datasets, achieving impressive convergence in three datasets—Multi-NGram-3, Multi-WE100, and Multi-WE400—and competitive values for two datasets, namely Multi-TFIDF, and Multi-Words.

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.

5.3.3 Experiment III: Comparison of the proposed HHO-WSVM approach against HHO-FsSVM

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

Table 5.8 illustrates the performance of the two approaches on the dataset versions. In the first data (Arabic), all results of the five versions show the superiority of the proposed approach to other methods. The HHO-WSVM 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 HHO-WSVM 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



(a) Arabic-NGram-3

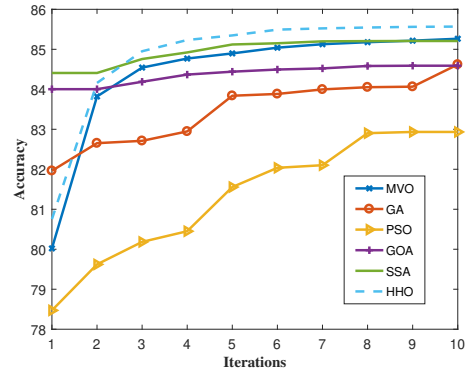
(b) Arabic-TFIDF

(c) Arabic-WE100

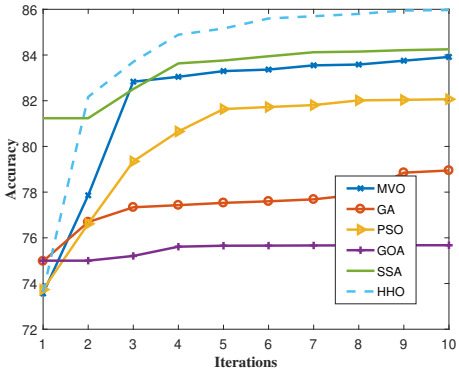
(d) Arabic-WE400

(e) Arabic-Words

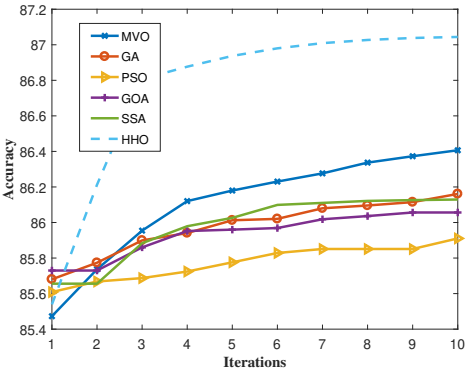
Fig. 5.2 Convergence curve charts for HHO and other algorithms based on 5 versions of the Arabic dataset



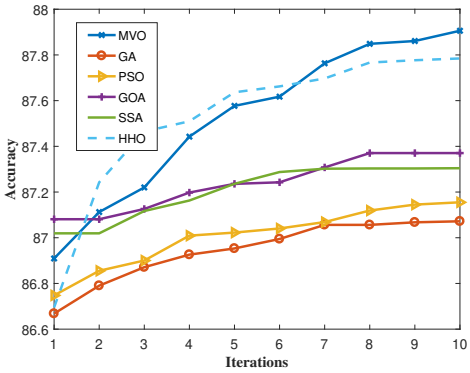
(a) English-NGram-3



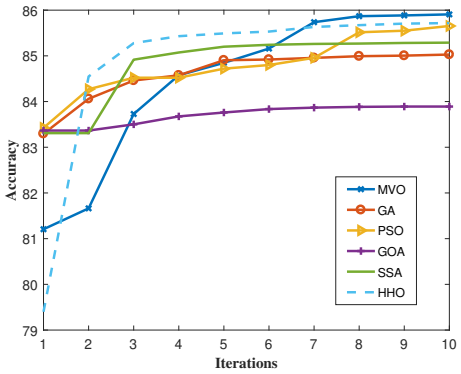
(b) English-TFIDF



(c) English-WE100

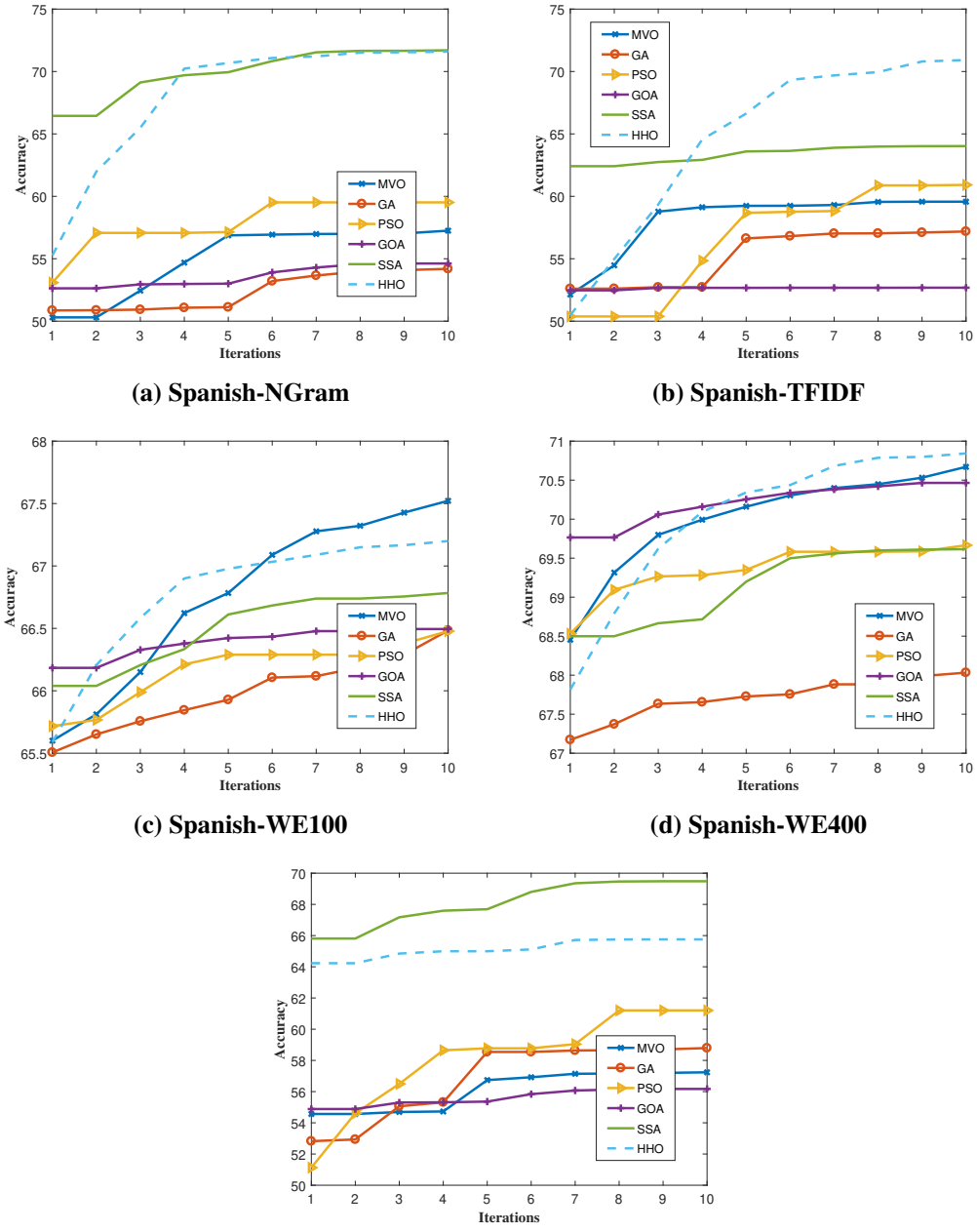


(d) English-WE400



(e) English-Words

Fig. 5.3 Convergence curve charts for HHO and other algorithms based on 5 versions of the English dataset



(a) Spanish-NGram

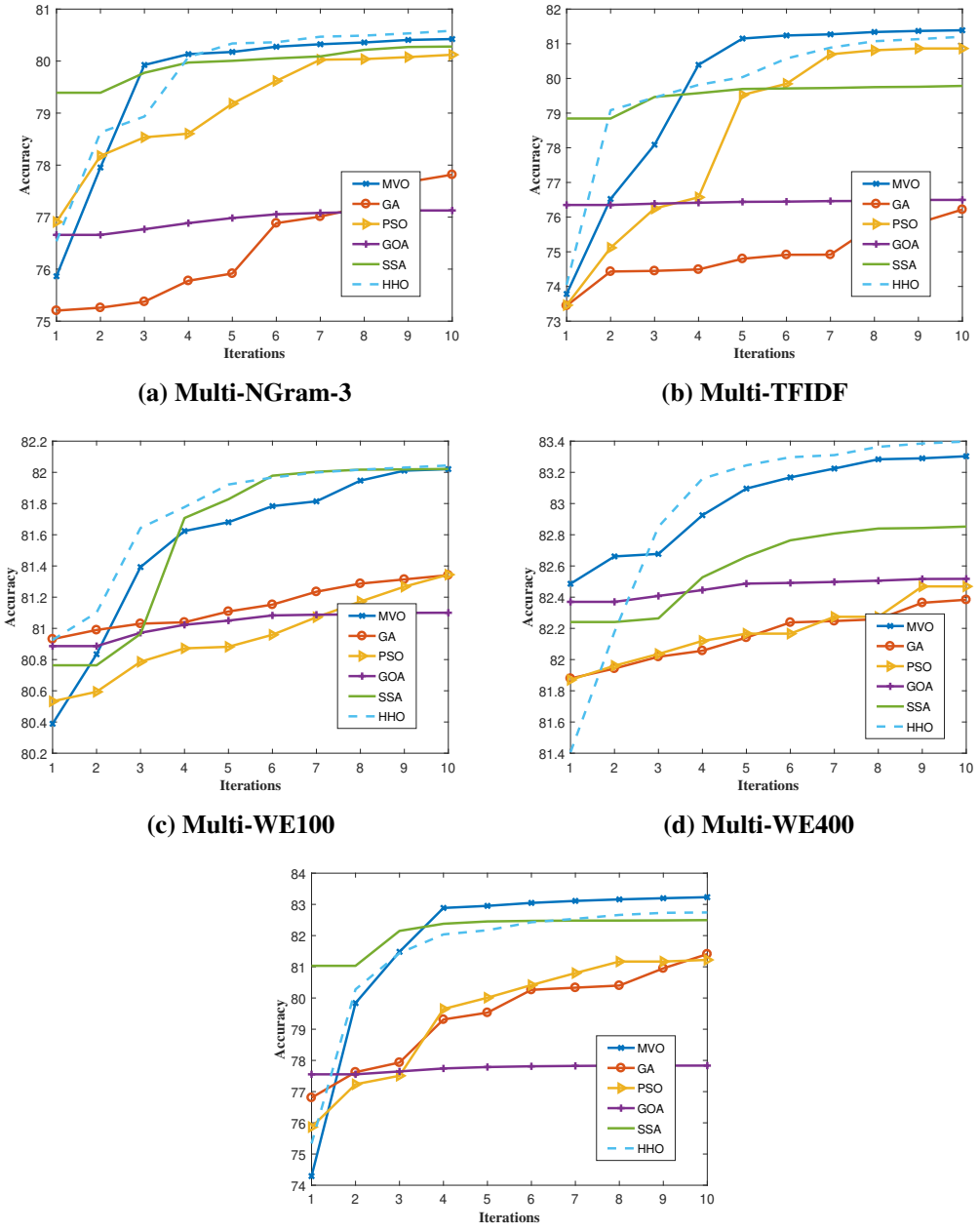
(b) Spanish-TFIDF

(c) Spanish-WE100

(d) Spanish-WE400

(e) Spanish-Words

Fig. 5.4 Convergence curve charts for HHO and other algorithms based on 5 versions of the Spanish dataset



(a) Multi-NGram-3

(b) Multi-TFIDF

(c) Multi-WE100

(d) Multi-WE400

(e) Multi-Words

Fig. 5.5 Convergence curve charts for HHO and other algorithms based on 5 versions of the Multilingual dataset

HHO-WSVM approach, whilst, NGram-3 and Words versions obtained the best results for HHO-FsSVM. On the other hand, the HHO-WSVM 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, HHO-WSVM 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 HHO-WSVM even when compared to HHO-FsSVM.

5.3.4 Experiment IV: Comparison of the proposed HHO-WSVM 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 5.6 presents the average accuracy of these models along with the proposed approach. As shown in the table, HHO-WSVM achieved the highest accuracy of 89.56% for the Arabic data, followed by EVO-RF, EVO-k-NN, and EVO-MLP. For the English data, HHO-WSVM also achieved the highest results, followed by EVO-RF with an accuracy of 86.547%. In the case of the Spanish data, HHO-WSVM attained the best results with an accuracy of 71.913%, while EVO-Bagging obtained the second-highest accuracy. Similarly, HHO-WSVM 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, HHO-WSVM demonstrated its superiority over other approaches by achieving the highest accuracy in these experiments.

5.3.5 Analysis and discussion

Four experimental phases were conducted in this first 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

Table 5.8 Results of the proposed approach (HHO-WSVM) and the feature selection approach (HHO-FsSVM)

Arabic				
Versions	HHO-WSVM		HHO-FsSVM	
	Avg	Std Dev	Avg	Std Dev
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	HHO-WSVM		HHO-FsSVM	
	Avg	Std Dev	Avg	Std Dev
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	HHO-WSVM		HHO-FsSVM	
	Avg	Std Dev	Avg	Std Dev
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	HHO-WSVM		HHO-FsSVM	
	Avg	Std Dev	Avg	Std Dev
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

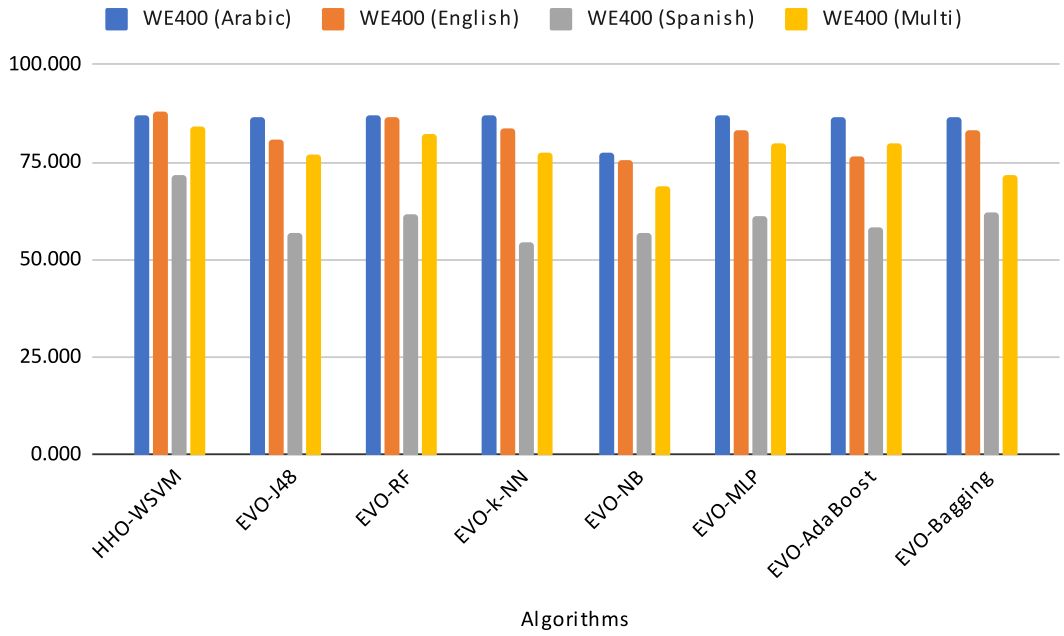


Fig. 5.6 Accurcy results of the proposed approach (HHO-WSVM) and various classifiers combined with evolutionary algorithms.

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 HHO-WSVM 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. This is attributed to various reasons, such as dialectal variations, and limitations in

data and resource availability. However, despite these challenges, they still exhibit promising competitiveness.

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.

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 5.9) 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 ver-

sion, 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 Dev.
- The new data (WE400StFE), the average accuracy is 90.992 with 0.026 Std Dev.

Table 5.9 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

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.

In this chapter, a hybrid Harris Hawks Optimization (HHO) and weighted Support Vector Machine have been used for spam detection. The HHO was responsible for optimizing the hyperparameter of the SVM and feature weighting. Moreover, multilingual datasets have been used (English, Spanish, and Arabic) and focused on the COVID-19 pandemic period. Several word representations have been compared. Besides, four experimental phases have been conducted, namely, well-known classic classification algorithms experiment, different evolutionary algorithms with SVM experiment, another version of the HHO-SVM experiment, and other classifiers combined with metaheuristic algorithms experiment. The results show that our proposed approach obtained the best results in every stage. Further,

a discussion of the reviews context alongside creating new data with statistical features is presented.

After developing an effective spam review detection approach, the following chapter will introduce the use of TwinSVM, a new variation of SVM, for identifying a different application known as Sentiment Analysis, then we will use additional pre-trained word embedding techniques in a multilingual context.

The use of sentiment analysis will provide us with more information than what we can uncover solely through spam review detection. This includes understanding the level of customer satisfaction or dissatisfaction, assisting businesses in pinpointing specific features or aspects of a product that customers appreciate or dislike, aiding companies in monitoring their brand perception in the market, facilitating competitor analysis, supporting customer service improvement, and providing decision support.

Moreover, the next approach will address several challenges associated with the current method. These challenges include reducing the execution time of the WSVM, simplifying the complexity in kernel selection, improving training efficiency, handling imbalanced data, and enabling multiclass classification.

Chapter 6

Approach Two: Sentiment Analysis Classification

6.1 Introduction

Currently, online social networks serve as the largest community for reviewing products and providing feedback, where everyone is free to post their thoughts whenever they want. As the most influential and effective platforms on the web, online social networks utilize user-friendly and convenient web-based platforms to facilitate the publication of user-generated content. Examining these reviews in this critical situation not only provides us with a deeper understanding of the current crisis but also offers valuable information for future similar situations. Recent years have witnessed a rapid increase in the popularity of social media websites [25]. In addition to their popularity, social media sites have become increasingly important in various areas of life, including politics, education, and business [241]. Nowadays, every business offers services and products online. These sites provide consumers with the ability to share their recommendations and experiences about these services, places, and products. Examples of such sites include Yelp, TripAdvisor, Facebook, and Twitter [303]. The widespread use of social media applications and websites has significantly increased the availability of product reviews [134]. Thus, there is a greater need than ever for automated methods to collect and analyze these reviews [165]. Therefore, these methods must be utilized to improve decision-making processes [212]. Governments and companies can both utilize these platforms to study and analyze consumer behavior, market their products, and mine reviews [15].

This issue can be addressed using sentiment analysis (SA). SA involves analyzing written text with the aim of identifying and categorizing the opinions expressed therein, which are

generally classified as negative, positive, or neutral [152]. SA was necessary due to the size of the data collected, extracted, and used in both structured and unstructured text obtained from the Internet [234]. SA applies NLP and text analysis techniques, either in combination or separately, to categorize reviews and opinions according to various criteria [152].

As SA is a large and evolving area of computational text analysis, there is no single best approach for its application. Researchers have conducted performance evaluations using various methods, including text classification using Naive Bayes, decision tree classifiers, and N-fold cross-validation [136]. There are two major approaches to sentiment analysis: knowledge-based SA and machine learning-based SA. The former relies on NLP algorithms or lexicon methods, while the latter analyzes data polarity based on pre-labeled texts, determining whether a text is neutral, positive, or negative [201, 100].

Using comments, reviews, and posts from customers, SA aims to categorize products and services into positive, negative, or neutral classes based on the sentiments expressed in the comments [107]. Both machine learning and lexicon-based approaches can be utilized for SA [96]. Various machine learning approaches, such as SVM, Random Forest, K-NN, and Neural Networks, are employed to evaluate sentiment-related results [301].

However, relying solely on machine learning algorithms without modifications and optimizations can sometimes yield unsatisfactory results. As a result, researchers have started combining evolutionary algorithms with machine learning methods to achieve optimal results. For example, [308] proposed a new algorithm based on the Local Search Improvised Bat Algorithm and Elman Neural Networks to enhance SA of online product reviews. Similarly, [208] investigated the analysis of customer reviews with imbalanced data distribution using a hybrid evolutionary SVM-based approach. Furthermore, Mehbodniya et al. proposed the use of deep belief networks and random evolutionary whale optimization algorithms to analyze the sentiment of online products [179]. They also introduced a hybrid feature selection algorithm to improve sentiment analysis by combining an evolutionary algorithm with an entropy-based metric for feature selection [64].

Using metaheuristic algorithms with SVM is not a novel approach. SVM has been utilized with metaheuristic algorithms in many applications, including, epileptic seizure detection [71], renewable energy [112], cancer classification [10], time series forecasting [199], intrusion detection [190], and credit scoring [252]. Most of the problems that have been applied to be solved in SVM are ranging from parameters optimization, feature selection, and fitness function determination [115].

In spite of this, SVM has several limitations that can impact its performance in certain cases. To overcome these limitations, we utilized Twin Support Vector Machines (Twin SVM). Twin SVM is an extension of the standard SVM algorithm used for classification

tasks. It incorporates two hyperplanes to create a "twin" structure that provides a more accurate representation of the data. The first hyperplane separates the majority class from the minority class, while the second hyperplane separates the minority class from the rest of the data. Compared to the standard SVM algorithm, Twin SVM offers several advantages. For instance, it is more effective in handling imbalanced data sets compared to standard SVMs. Additionally, Twin SVM is more flexible and efficient when using different kernel functions. While standard SVM is sensitive to the selection of kernel functions, Twin SVM can adapt to various kernel functions.

Therefore, this chapter proposed a TwinSVM and Salp Swarm Algorithm (SSA) for sentiment prediction. SSA is used here, after several experiments, as an optimization algorithm to optimize TwinSVM parameters and ultimately improve its performance. In addition, pre-trained word embeddings were applied as a word representation to deal with the big data issue. Furthermore, we once again investigated the multilingual context of reviews written in different languages; however, here, we take sentiment analysis into account. Taking the multilingual environment into account, three-word embedding techniques were employed, including MUSE, BERT, and FastText. Additionally, similar to the previous chapter, this study will particularly focus on the COVID-19 period to gain a more detailed understanding of the review targets, style, attitude, and context before and after the pandemic.

Several experiments have been proposed, including, a comparison of standard classification models that are well-known in the literature (e.g. J48, k-NN, NB, RF, and MLP), a comparison of regular SVM with other metaheuristic algorithms, and a comparison of the proposed SSA-TwinSVM with other metaheuristic algorithms. All comparisons were conducted based on the accuracy metric, with recall and precision added to experiment III. The datasets were categorized into three sentiment classes: positive, negative, or neutral. Finally, an analysis of how the multilingual environment affects the results using different word embedding techniques has been described, along with a comprehensive analysis of reviews.

The contributions of this chapter are summarized as follows:

- TwinSVM and SSA algorithm combined together for optimizing the parameters to improve its performance.
- The COVID-19 pandemic period and the multilingual environment were considered in generating new sentiment datasets of the original data that was explained in Chapter 4. As a result, four datasets are constructed, comprising, Arabic, English, Spanish, and multilingual (all three merged together).

- In order to improve word representation for limited datasets and multilingual contexts, three pre-trained word embedding techniques have been employed (MUSE, FastText, and BERT).
- Examining the reviews taken before and after the COVID-19 pandemic from various perspectives, including, style, context, and product direction.

6.2 Approach two Methodology

The proposed approach of this chapter is presented in this section. In this case, we describe the design of the approach in three phases, namely, representation, fitness, and architecture of the approach. Then we provide the Evaluation mechanisms used to evaluate the approach performance.

Further, similar to the previous chapter, numerical markers (***{1}, {2}, {3}***) have been added to the method figure (Fig. 6.1) and in the text for more understanding of the process.

6.2.1 Design of the approach

We will begin by defining the solution vector for our problem, then determine the best fitness function for evaluating our solution, and ultimately present the approach architecture. As it can be seen, these are the stages in explaining our approach design.

- **Representation of the solution:**

  The first step to optimize Twin-SVM parameters is to design the solution representation before using the SSA algorithm. In this version of the SVM (TwinSVM), the number of parameters is different from the original SVM. Therefore, we should consider generating a new solution vector that contains these parameters. In other words, to determine the optimal parameters of C_1, C_2, and γ. Thus, SSA algorithm produces elements for C_1, C_2 and γ. The initial values in the vector for the three elements consist of real numbers. Consequently, the search space of the parameters are scaled to [-2,2] for C_1 and C_2 while [-8,2] for γ. As shown in Eq. 5.1 in Chapter 5, min-max normalization is used to perform this transformation.

- **Fitness function:**

  In every iteration, the fitness function provides feedback for SSA based on an assessment of each individual. Based on Eq. 5.2 in Chapter 5, the classification accuracy of the Twin SVM classifier is chosen as the fitness function for evaluation.

- **Architecture of the approach:**

  With our proposed approach, we separate each dataset into a training set and a testing set ***{1}***. The splitting criterion is applied based on how many experiments are conducted. For each experiment (k), $k-(1/k)$ parts of the dataset are divided evenly and used as training parts, and $1/k$ parts as testing parts. Doing that will diversify the training and testing sets and produce the best possible models.

  At the beginning of each iteration, SSA generates a random vector of real numbers. The numbers are utilized for determining C_1, C_2 and γ ***{2}***. Using the generated solution, the training process of Twin SVM classifiers begins ***{3}***. Further, during training, inner cross-validation is used to produce a more robust model. Following the training process, SSA received the fitness value based on the accuracy of the Twin SVM classifier ***{4}***. To meet the SSA termination criterion, we repeat the prior processes until the highest number of iterations is reached ***{5}***. Then the best solution is determined ***{6}*** and used for the testing part for evaluation ***{7}***. In the end, each stage is repeated k times, and the average value is calculated. The mentioned processes are portrayed in Figure 6.1.

6.2.2 Evaluation Metrics

As for evaluation, three measures have been conducted in this study, including, accuracy, recall, and precision using the confusion matrix table utilized in Chapter 5 (Table 5.1). The accuracy equation is calculated by Eq. 5.3. As for the other metrics, they are explained as follows:

- *Recall*: This can be calculated by the proportion of all actual instances correctly identified across all classes.

$$Recall = \frac{TP}{TP+FN} \tag{6.1}$$

- *Precision*: Based on the percentage of all identified instances that are actually correct across all classes.

$$Precision = \frac{TP}{TP+FP} \tag{6.2}$$

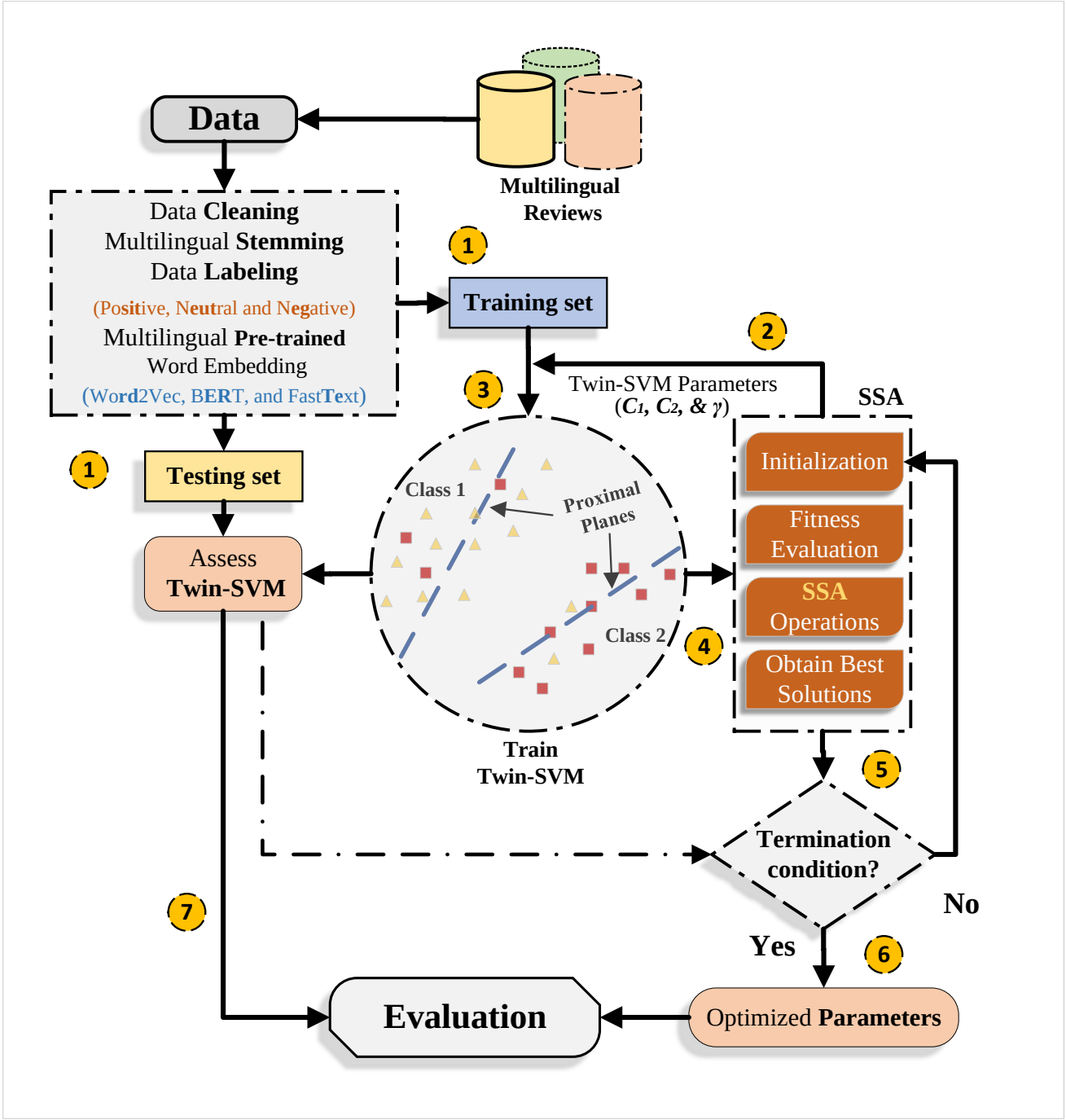


Fig. 6.1 The proposed SSA-TSVM process. The numerical annotations in the figure serve as a mapping mechanism between the text and the methodology figure.

6.3 Approach two Experiment and Results

This section presents three different experiments, including the well-known classification algorithms, the experiments of applying SVM with varying metaheuristic algorithms, and the proposed SSA-TwinSVM compared to the other metaheuristic algorithms. In addition, the analysis of the reviews and the importance of the features are discussed in this section. The experiments were conducted on a PC with an AMD Ryzen 5 5600X CPU and 16.0GB RAM. Windows 10 operating system and Python environment are used to implement the algorithms. The initial parameters used for the SSA are the same as the original paper, where c_1 ranges from [0-1] and c_2 ranges from [0-1], with 30 iterations for all algorithms. The number of iterations was chosen after several experiments to seek the best values and performance. Moreover, three metrics have been considered for comparison, namely, accuracy, recall, and precision. Recall and precision are added to evaluate the performance of classification in case there are imbalanced class distributions.

In this second approach, the datasets mentioned in Chapter 4 are labeled here as Positive, Neutral, and Negative, so it is kinda of different data (different classes) than the previous approach. Additionally, unlike the previous chapter (approach one), three pre-trained word embedding methods were utilized, including MUSE, BERT, and FastText.

6.3.1 Experiment I: Comparison of standard classification models

Experiment I includes a comparison between well-known induction classification algorithms commonly used in the literature (J48, k-NN, NB, RF, and MLP) to demonstrate the reliability of the data and establish a baseline for the results. The experiments are divided into three parts: the MUSE, BERT, and FastText for Arabic, English, Spanish, and Multilingual datasets for 100 and 400 dimensions. Tables 6.1, 6.2, and 6.3 show the results for applying MUSE, BERT, and FastText, respectively.

The accuracy results of the MUSE, represented in Table 6.1, show that the RF algorithm outperforms the other algorithms for the English, Spanish, and Multilingual datasets for 100 and 400 dimensions. In contrast, k-NN and MLP achieved better results than RF for the Arabic dataset for the 100 and 400 dimensions. It is also observed that NB has generated the worst results compared to the other algorithms for the Arabic and English datasets. It did not perform well for the Spanish and Multilingual datasets compared to the RF algorithm.

Different observations are concluded from Table 6.2, which represents the accuracy results of BERT word embedding. The RF algorithm shows the best results for the English dataset compared to the other algorithms, whereas MLP shows the best results for the Multilingual dataset. For the Arabic dataset, RF performed better than the different algorithms for the

Arabic and Spanish datasets, with a dimension value of 400. In contrast, the k-NN, NB, and MLP algorithms achieved the best results for the Arabic dataset with a dimension value of 100. Moreover, NB achieved the best results for the Spanish dataset with a dimension value of 100.

The RF algorithm again achieved the best accuracy results for the FastText, represented in Table 6.3, for almost all the datasets except for the Spanish dataset with the dimension value of 100, where the NB algorithm achieved the best results. The reason for this is that the dataset has unique features and patterns that align more with the probabilistic modeling such as NB.

Table 6.1 Accuracy results of different classic algorithms based on the MUSE word embedding in 100 and 400 dimensions for Arabic, English, Spanish, and Multilingual datasets

Algorithm	J48	k-NN	NB	RF	MLP
Arabic					
MUSE-100	69.296	**70.363**	28.998	69.723	70.149
MUSE-400	69.936	69.936	-	69.936	**70.149**
English					
MUSE-100	69.751	74.740	63.327	**78.285**	76.780
MUSE-400	70.902	74.811	-	**78.437**	78.123
Spanish					
MUSE-100	56.222	54.223	56.872	**63.019**	61.669
MUSE-400	55.122	55.572	-	**65.417**	63.268
Multi					
MUSE-100	46.868	45.631	49.487	**52.001**	48.937
MUSE-400	45.679	44.750	-	**52.308**	47.733

6.3.2 Experiment II: Comparison of regular SVM with other metaheuristic algorithms

In this section, the performance of SVM with various metaheuristic algorithms is explored. The enhancement of sentiment prediction involves feature weighting and parameter optimization, demonstrating the SVM's performance before experimenting with the proposed Twin-SVM. The PSO, GOA, SSA, WOA, and HHO metaheuristic algorithms are used for feature weighting and parameter tuning.

Table 6.2 Accuracy results of different classic algorithms based on the BERT word embedding in 100 and 400 dimensions for Arabic, English, Spanish, and Multilingual datasets

Algorithm	J48	k-NN	NB	RF	MLP
Arabic					
BERT-100	87.207	**88.699**	**88.699**	88.272	**88.699**
BERT-400	87.207	88.699	87.420	**88.913**	87.420
English					
BERT-100	81.527	80.396	67.064	**82.396**	72.154
BERT-400	82.749	80.184	68.912	**86.153**	78.558
Spanish					
BERT-100	69.815	64.368	**77.511**	76.012	71.4643
BERT-400	67.816	58.871	78.611	**78.911**	68.766
Multi					
BERT-100	48.953	47.902	49.762	50.732	**74.613**
BERT-400	49.285	47.805	48.226	51.087	**66.067**

Table 6.3 Accuracy results of different classic algorithms based on the FastText word embedding in 100 and 400 dimensions for Arabic, English, Spanish and Multilingual datasets

Algorithm	J48	k-NN	NB	RF	MLP
Arabic					
Fast-100	87.207	**88.699**	**88.699**	**88.699**	**88.699**
Fast-400	87.207	88.699	87.420	**89.126**	88.273
English					
Fast-100	81.325	80.416	67.236	**81.951**	76.184
Fast-400	83.032	80.366	68.922	**86.304**	78.467
Spanish					
Fast-100	70.415	64.968	**77.411**	77.261	75.112
Fast-400	66.617	58.971	78.361	**79.410**	69.165
Multi					
Fast-100	48.274	47.959	50.287	**50.643**	48.129
Fast-400	48.864	47.506	48.145	**50.901**	41.686

Tables 6.4, 6.5, and 6.6 show the average and standard deviation accuracy values, offering a comprehensive understanding of central tendency and variability, that are obtained from conducting the experiments using the SVM algorithm with the different metaheuristic algorithms for the dimension values of 100 and 400. The MUSE, BERT, and FastText word embedding are used in Tables 6.4, 6.5, and 6.6, respectively.

The MUSE results show that the HHO algorithm outperforms the other metaheuristic algorithms for the English, Spanish, and Multilingual datasets for the dimension value of 100. The WOA algorithm shows the best results for the Arabic dataset for the dimension value of 100. In contrast, the dimension value of 400 shows different results for the different datasets by having the best accuracy for the PSO, WOA, GOA, and WOA for the Arabic, English, Spanish, and Multilingual datasets, respectively. On the other hand, the BERT shows that HHO outperforms the different metaheuristic algorithms for the English and Spanish datasets for a dimension value of 100 and the Arabic and Multilingual dataset for a dimension value of 400. In contrast, the WOA algorithm obtains the best results for the English and Spanish datasets for the dimension value of 400. In addition, PSO and GOA achieved the best results for the Multilingual dataset for the dimension value of 100, but the GOA has a better standard deviation. The Arabic dataset with a dimension value of 100 has the best results when applying the SSA algorithm. Moreover, Table 6.6 shows that PSO has the best results for the Arabic, Spanish, and Multilingual datasets for the dimension value of 100, and it has the best results for the Arabic dataset for the dimension value of 400. WOA outperforms the other metaheuristic algorithms for the English dataset for both dimension values. In addition, HHO has the best results for the Spanish dataset, while GOA has the best accuracy values for the Multilingual dataset for the dimension value of 400.

In conclusion, MUSE and BERT analyses reveal HHO's consistent superiority in performance across English and Spanish datasets at a dimension value of 100, while WOA excels specifically for the Arabic dataset. At a dimension value of 400, various algorithms, including PSO, WOA, GOA, and HHO, exhibit optimal accuracy for different datasets, highlighting the importance of tailoring metaheuristic choices based on specific dimensional requirements.

6.3.3 Experiment III: Comparison of the proposed SSA-TwinSVM with other metaheuristic algorithms

This section compares the performance of the proposed SSA-TwinSVM with other TwinSVM metaheuristic algorithms, proving the superiority of TwinSVM compared to the standard SVM.

Table 6.4 Accuracy results of different metaheuristic algorithms combined with SVM based on the MUSE word embedding in 100 and 400 dimensions for Arabic, English, Spanish, and Multilingual datasets

Algorithm	MUSE-100		MUSE-400	
	Avg	Std Dev	Avg	Std Dev
Arabic				
PSO-SVM	70.365	6.820	**70.153**	**5.188**
GOA-SVM	70.569	8.700	69.746	6.084
SSA-SVM	70.352	4.730	69.898	8.647
WOA-SVM	**71.189**	**8.487**	70.139	5.562
HHO-SVM	70.153	5.188	69.935	9.434
English				
PSO-SVM	78.224	1.363	79.487	1.389
GOA-SVM	77.992	1.365	79.608	1.384
SSA-SVM	77.810	1.709	79.406	0.943
WOA-SVM	78.770	2.001	**80.174**	**0.827**
HHO-SVM	**79.002**	**1.719**	79.901	1.090
Spanish				
PSO-SVM	66.368	4.771	69.666	2.852
GOA-SVM	66.318	4.398	**70.714**	**3.780**
SSA-SVM	66.515	3.955	69.867	3.063
WOA-SVM	65.666	2.206	70.064	2.585
HHO-SVM	**66.615**	**3.228**	70.313	2.738
Multi				
PSO-SVM	50.489	1.484	51.580	0.992
GOA-SVM	50.594	2.145	51.516	2.385
SSA-SVM	51.677	1.180	51.597	1.717
WOA-SVM	51.807	1.133	**51.791**	**1.266**
HHO-SVM	**52.049**	**1.006**	51.572	1.578

Table 6.5 Accuracy results of different metaheuristic algorithms combined with SVM based on the BERT word embedding in 100 and 400 dimensions for Arabic, English, Spanish, and Multilingual datasets

Algorithm	BERT-100		BERT-400	
	Avg	Std Dev	Avg	Std Dev
Arabic				
PSO-SVM	71.637	5.612	73.335	6.273
GOA-SVM	72.234	9.415	69.732	7.914
SSA-SVM	**72.724**	**6.882**	72.484	6.198
WOA-SVM	72.044	7.193	72.919	6.357
HHO-SVM	71.013	8.247	**73.765**	**6.848**
English				
PSO-SVM	69.932	1.490	74.134	1.512
GOA-SVM	69.599	1.489	70.569	2.517
SSA-SVM	70.427	1.724	73.377	2.150
WOA-SVM	70.821	1.516	**74.538**	**2.038**
HHO-SVM	**71.154**	**1.621**	74.518	0.846
Spanish				
PSO-SVM	76.610	7.324	70.761	12.278
GOA-SVM	66.513	9.499	64.425	15.603
SSA-SVM	73.110	10.733	72.929	11.625
WOA-SVM	77.362	2.886	**78.510**	**2.560**
HHO-SVM	**77.860**	**2.740**	78.311	2.686
Multi				
PSO-SVM	**50.618**	**2.236**	50.626	1.028
GOA-SVM	**50.618**	**1.362**	50.319	1.182
SSA-SVM	50.538	1.406	50.635	1.187
WOA-SVM	50.586	0.996	51.039	1.006
HHO-SVM	50.546	1.683	**51.192**	**1.883**

Table 6.6 Accuracy results of different metaheuristic algorithms combined with SVM based on the FastText word embedding in 100 and 400 dimensions for Arabic, English, Spanish, and Multilingual datasets

Algorithm	Fast-100		Fast-400	
	Avg	Std Dev	Avg	Std Dev
Arabic				
PSO-SVM	**72.493**	**6.225**	**74.408**	**6.899**
GOA-SVM	71.841	7.693	71.637	5.960
SSA-SVM	71.647	4.456	73.529	5.674
WOA-SVM	69.713	9.164	72.896	8.697
HHO-SVM	72.280	6.656	71.457	7.588
English				
PSO-SVM	69.357	1.761	74.043	1.834
GOA-SVM	69.620	1.798	73.811	1.788
SSA-SVM	69.720	1.133	74.821	1.266
WOA-SVM	**70.064**	**1.620**	**74.922**	**1.374**
HHO-SVM	69.953	1.637	74.639	2.179
Spanish				
PSO-SVM	**77.860**	**1.668**	74.776	10.108
GOA-SVM	68.610	9.172	74.011	8.774
SSA-SVM	77.713	2.677	74.927	10.743
WOA-SVM	76.810	2.920	77.062	3.238
HHO-SVM	77.063	2.482	**77.260**	**2.759**
Multi				
PSO-SVM	**50.538**	**1.235**	50.384	1.298
GOA-SVM	50.311	1.584	**50.861**	**1.528**
SSA-SVM	50.368	1.068	50.643	0.825
WOA-SVM	50.400	1.176	50.505	1.101
HHO-SVM	50.392	1.042	50.335	1.375

Table 6.7 presents the accuracy results obtained from conducting the experiments for the TwinSVM with several metaheuristic algorithms for the MUSE word embedding in 100 and 400 dimensions for Arabic, English, Spanish and Multilingual datasets. It is observed that the SSA-TwinSVM outperforms the other algorithms for all of the datasets for the dimension value of 400, having the rank of 1. In contrast, it beats the different algorithms for the dimension value of 100 for the Arabic dataset. On the other hand, the HHO, MVO, and WOA show the best accuracy values for the English, Spanish, and Multilingual datasets for the dimension value of 100. The SSA-TwinSVM is ranked 3, 4, and 2 for the English, Spanish, and Multilingual datasets, respectively. Looking into the accuracy results of the BERT word embedding, represented in Table 6.8, the SSA-TwinSVM outperforms the other algorithms in almost all the datasets, having the rank of 1, except for the Arabic dataset with the dimension value of 100 and the Spanish dataset with the dimension value of 400, where the WOA-TwinSVM shows the best results for these datasets on the specified dimension values, and the SSA-TwinSVM shows the rank of 2 for these datasets. Finally, table 6.9 presents the accuracy results for the FastText word embedding with the best results of the SSA-TwinSVM for the Arabic and Multilingual datasets for the dimension value of 100 and the Arabic and English datasets for the dimension value of 400. The GOA and PSO show the best results for the English and Spanish datasets for the dimension value of 100, respectively. In contrast, the WOA shows the best results for the Spanish and Multilingual datasets for the dimension value of 400.

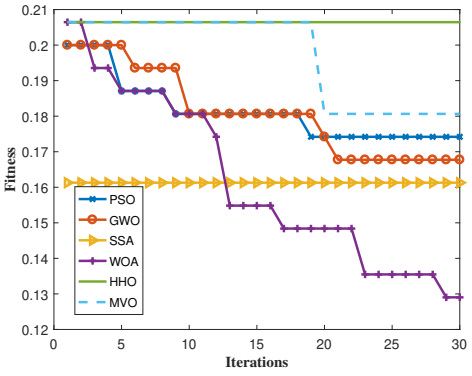
The convergence curves for the experiments are presented in Figures 6.2, 6.3, and 6.4. SSA-TwinSVM shows a competitive convergence behavior, compared to the other algorithms, for the Arabic-400, English-400, and Multi-400 datasets, of the MUSE word embedding experiments. However, it did not converge for the Arabic-100, English-100, and Multi-100 datasets across the course of iterations. Which may indicate either the attainment of an optimal solution or being trapped in a local minimum. Further, it decreased strikingly at the ninth iteration with the Arabic-400 dataset, the eighth and tenth iteration for the English-400 dataset, and the sixth iteration for the Spanish-400 dataset. In contrast, the BERT word embedding for the SSA-TwinSVM shows competitive values across the iterations for the Arabic-400, Spanish-400, and Multi-100 datasets. The WOA shows better values than the SSA for the Arabic-100, Spanish-400, and Multi-400 datasets. SSA descended steadily for the Arabic-100 dataset and has enduring values after the 15 iterations. On the other hand, by looking into the convergence of the FastText word embedding, the best values are achieved across the course of iterations by the SSA-TwinSVM for the Arabic-400 and English-400 datasets. In contrast, it shows advanced convergence for the English-100 and Multi-100 datasets at later iterations.

Table 6.7 Accuracy results of different metaheuristic algorithms combined with Twin-SVM based on the MUSE word embedding in 100 and 400 dimensions for Arabic, English, Spanish, and Multilingual datasets

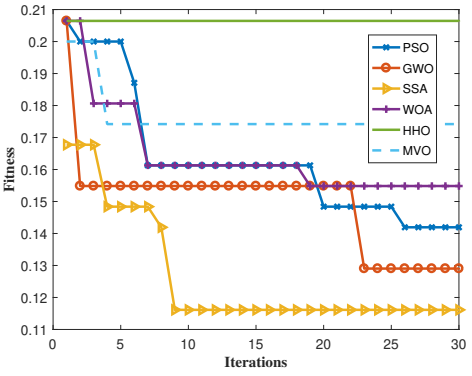
Datasets		Algorithm	PSO-TwinSVM	GOA-TwinSVM	SSA-TwinSVM	WOA-TwinSVM	HHO-TwinSVM	MVO-TwinSVM
Arabic	MUSE-100	Avg	82.000	81.900	**84.530**	83.770	79.000	80.100
		Std Dev	1.050	1.090	**2.920**	2.400	0.000	1.470
	MUSE-400	Avg	83.730	85.330	**87.600**	83.800	79.000	82.700
		Std Dev	2.050	1.490	**2.810**	1.630	0.000	0.920
English	MUSE-100	Avg	82.330	81.500	82.600	82.800	**83.370**	79.770
		Std Dev	1.320	0.680	3.290	1.100	**2.040**	0.430
	MUSE-400	Avg	79.030	82.500	**86.330**	82.800	82.630	83.500
		Std Dev	0.180	1.940	**3.110**	1.540	0.960	1.720
Spanish	MUSE-100	Avg	83.900	82.730	83.030	82.670	83.670	**84.200**
		Std Dev	1.450	1.010	3.360	1.090	1.580	**1.710**
	MUSE-400	Avg	82.070	82.300	**84.830**	83.070	83.370	81.870
		Std Dev	1.230	1.680	**3.170**	1.010	1.100	1.610
Multi	MUSE-100	Avg	83.000	80.100	84.530	**84.600**	80.800	82.130
		Std Dev	1.020	0.550	2.920	**0.970**	1.210	1.140
	MUSE-400	Avg	85.030	82.430	**87.100**	80.530	80.730	83.670
		Std Dev	1.710	1.590	**3.130**	0.680	0.640	1.090

Table 6.8 Accuracy results of different metaheuristic algorithms combined with Twin-SVM based on the BERT word embedding in 100 and 400 dimensions for Arabic, English, Spanish, and Multilingual datasets

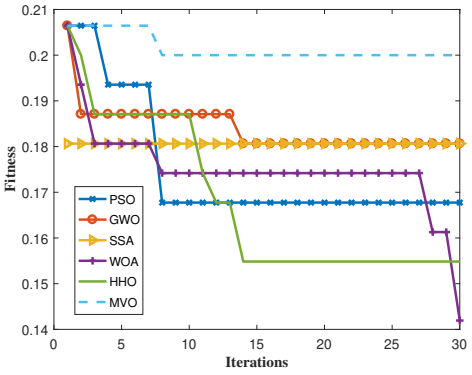
Datasets		Algorithm	PSO-TwinSVM	GOA-TwinSVM	SSA-TwinSVM	WOA-TwinSVM	HHO-TwinSVM	MVO-TwinSVM
Arabic	BERT-100	Avg	81.833	82.833	83.867	**84.133**	81.000	81.567
		Std Dev	0.999	1.578	3.266	**0.794**	1.829	1.563
	BERT-400	Avg	82.167	81.300	**83.933**	80.867	79.533	80.967
		Std Dev	0.999	1.578	**3.266**	0.794	1.829	1.563
English	BERT-100	Avg	82.533	80.900	**84.500**	83.633	83.567	82.467
		Std Dev	0.999	1.578	**3.266**	0.794	1.829	1.563
	BERT-400	Avg	81.767	83.267	**84.633**	83.633	83.500	81.333
		Std Dev	0.999	1.578	**3.266**	0.794	1.829	1.563
Spanish	BERT-100	Avg	83.730	79.370	**85.830**	81.470	81.870	83.000
		Std Dev	1.000	1.580	**3.270**	0.790	1.830	1.560
	BERT-400	Avg	84.170	83.100	84.870	**85.300**	81.870	84.870
		Std Dev	1.000	1.580	3.270	**0.790**	1.830	1.560
Multi	BERT-100	Avg	82.567	82.100	**86.367**	79.000	81.333	80.000
		Std Dev	0.999	1.578	**3.266**	0.794	1.829	1.563
	BERT-400	Avg	81.167	83.000	**85.567**	84.800	81.767	83.867
		Std Dev	0.999	1.578	**3.266**	0.794	1.829	1.563



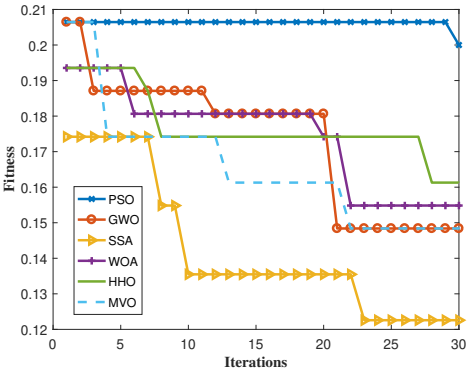
(a) Arabic-100



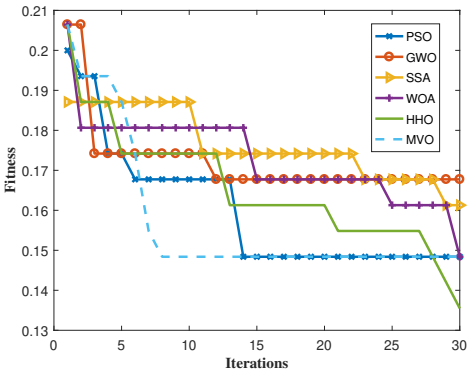
(b) Arabic-400



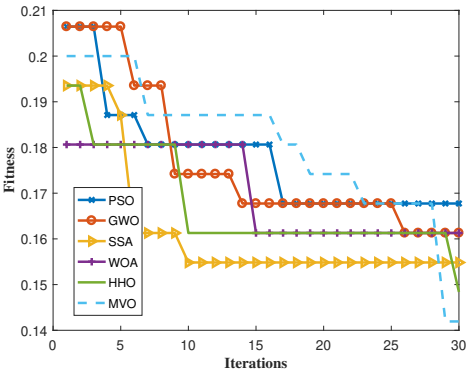
(c) English-100



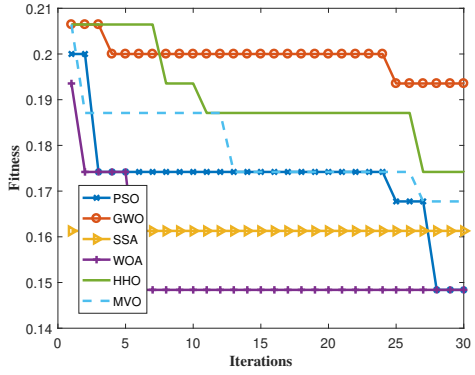
(d) English-400



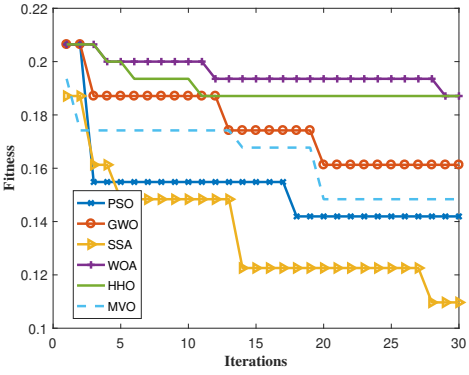
(e) Spanish-100



(f) Spanish-400

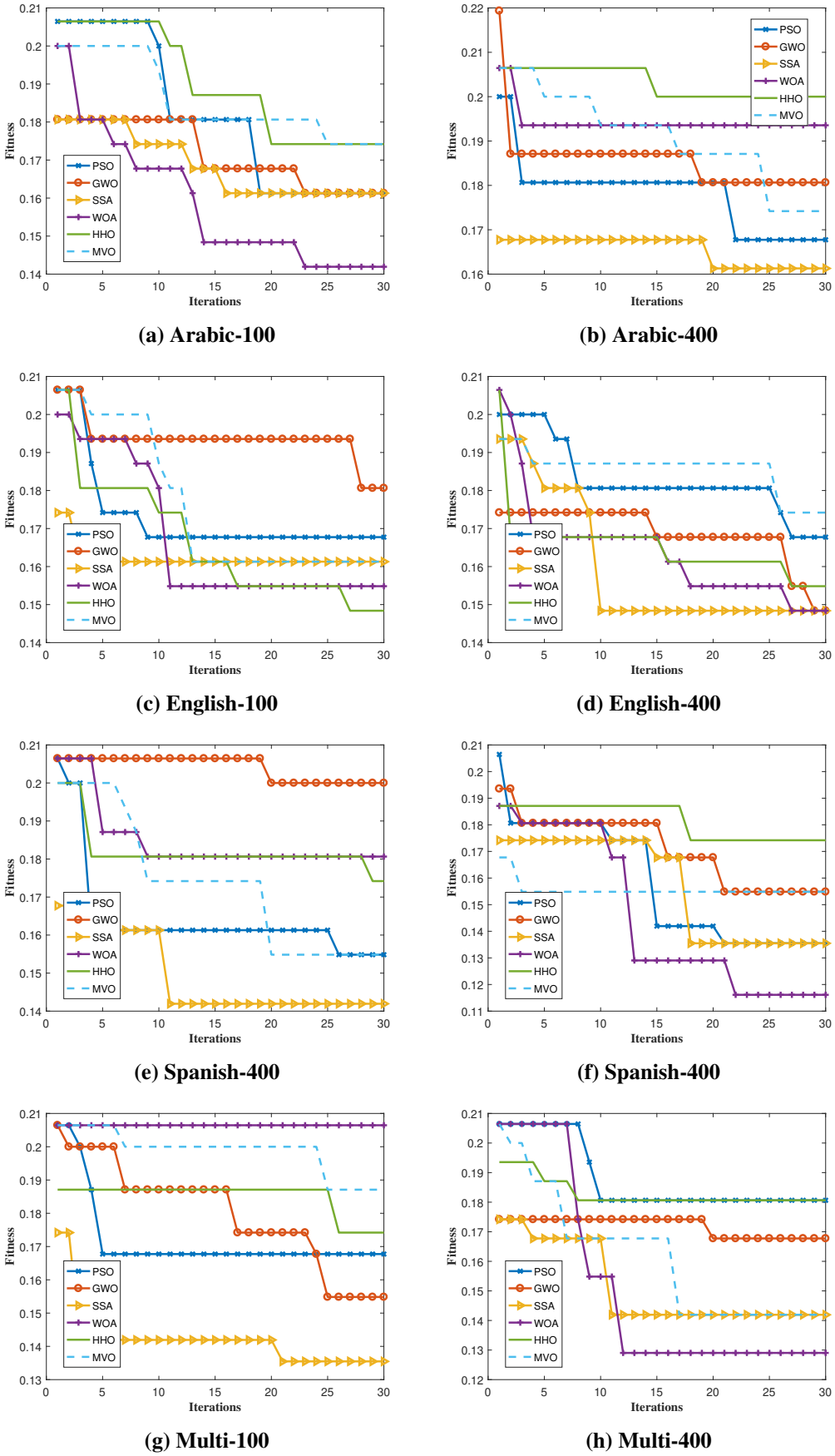


(g) Multi-100



(h) Multi-400

Fig. 6.2 Convergence curve charts based on the fitness for SSA and other algorithms based on MUSE 100 & 400 dimensions of each language



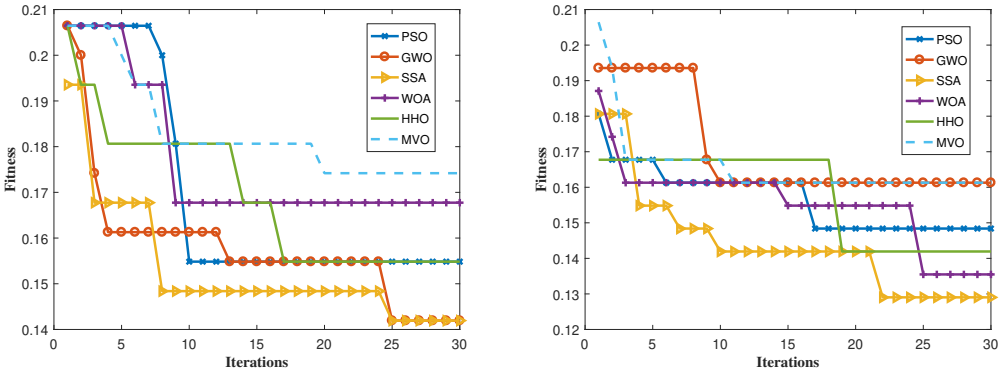
(a) Arabic-100 **(b) Arabic-400**

(c) English-100 **(d) English-400**

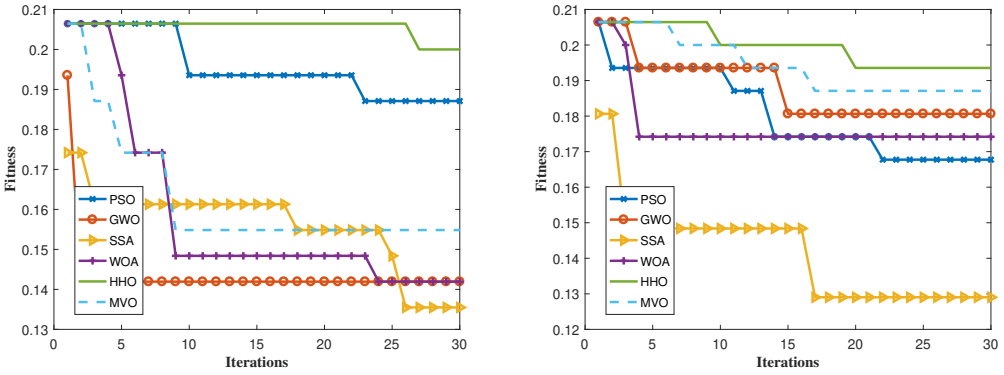
(e) Spanish-400 **(f) Spanish-400**

(g) Multi-100 **(h) Multi-400**

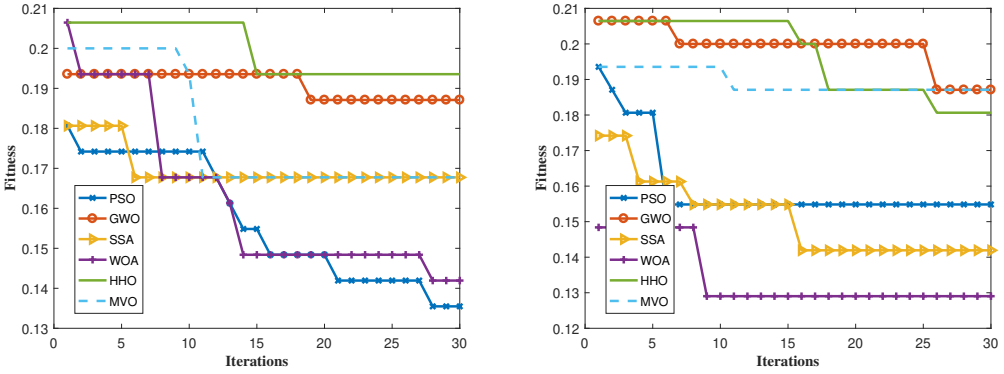
Fig. 6.3 Convergence curve charts based on the fitness for SSA and other algorithms based on BERT 100 & 400 dimensions of each language



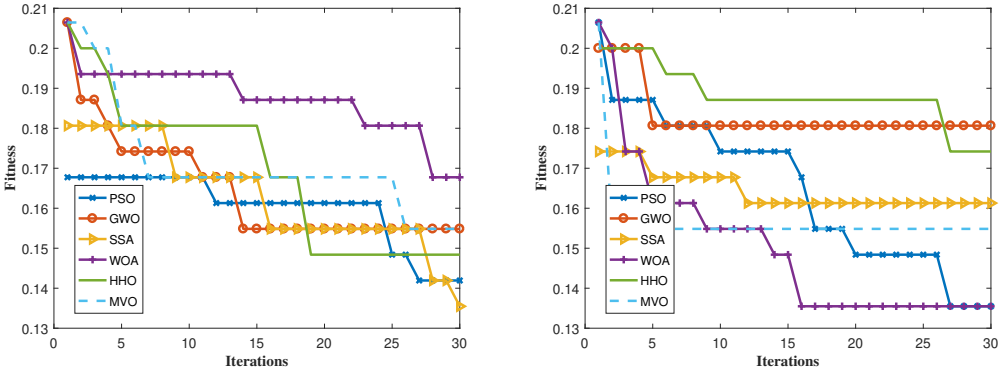
(a) Arabic-100 (b) Arabic-400



(c) English-100 (d) English-400



(e) Spanish-100 (f) Spanish-400



(g) Multi-100 (h) Multi-400

Fig. 6.4 Convergence curve charts based on the fitness for SSA and other algorithms based on FastText 100 & 400 dimensions of each language

Table 6.9 Accuracy results of different metaheuristic algorithms combined with Twin-SVM based on the FastText word embedding in 100 and 400 dimensions for Arabic, English, Spanish, and Multilingual datasets

Datasets		Algorithm	PSO-TwinSVM	GOA-TwinSVM	SSA-TwinSVM	WOA-TwinSVM	HHO-TwinSVM	MVO-TwinSVM
Arabic	Fast-100	Avg	83.333	84.467	**85.233**	82.133	83.333	81.833
		Std Dev	0.999	1.578	**3.266**	0.794	1.829	1.563
	Fast-400	Avg	84.267	83.167	**86.300**	84.600	84.200	83.467
		Std Dev	0.999	1.578	**3.266**	0.794	1.829	1.563
English	Fast-100	Avg	80.400	**85.833**	85.100	84.100	79.133	84.067
		Std Dev	0.999	**1.578**	3.266	0.794	1.829	1.563
	Fast-400	Avg	82.067	81.333	**86.333**	82.633	80.067	80.433
		Std Dev	0.999	1.578	**3.266**	0.794	1.829	1.563
Spanish	Fast-100	Avg	**84.467**	81.000	83.433	83.733	80.067	82.033
		Std Dev	**0.999**	1.578	3.266	0.794	1.829	1.563
	Fast-400	Avg	84.433	79.967	85.733	**86.467**	80.100	81.000
		Std Dev	0.999	1.578	3.266	**0.794**	1.829	1.563
Multi	Fast-100	Avg	83.967	83.833	**84.433**	81.300	83.033	82.800
		Std Dev	0.999	1.577	**3.266**	0.794	1.829	1.562
	Fast-400	Avg	83.533	81.733	84.200	**84.867**	81.100	84.800
		Std Dev	0.999	1.578	3.266	**0.794**	1.829	1.563

Further, recall and precision measures have been added for a more comprehensive comparison due to the presence of an imbalanced class distribution of some datasets. The comparison was conducted between all algorithms of the best datasets of each language. Figure 6.5 shows the recall results, where the SSA-TwinSVM obtained the highest values proving the algorithm's ability to capture a larger proportion of true positive instances. As for the precision results presented in Figure 6.6, the prospered SSA-TwinSVM achieved the best results in most datasets except the Multi-MUSE-400, demonstrating that the precision focuses on the accuracy of positive predictions.

6.3.4 Analysis of results, multilingual, and reviews

By reading reviews, consumers can find out more about a product or service before they buy it. In addition, organizations can improve their products by modifying their designs or redesigning their strategies. It is also pertinent to note that the context of reviews during and after the pandemic has shifted. The focus has been directed more toward health and technology products rather than, for example, clothing and beauty products. Changes in review structure and context resulted in changes in sentiment methods as well. Further, words (features) that are capable of predicting the sentiment of reviews have also been modified. During this study, we noticed that customers nowadays focus more on medical and entertainment products than any other products. Most reviews and features have changed across all languages. Thus, we must adapt our review analysis to reflect such changes in context and structure. To improve and predict the sentiment of reviews, we use pre-trained

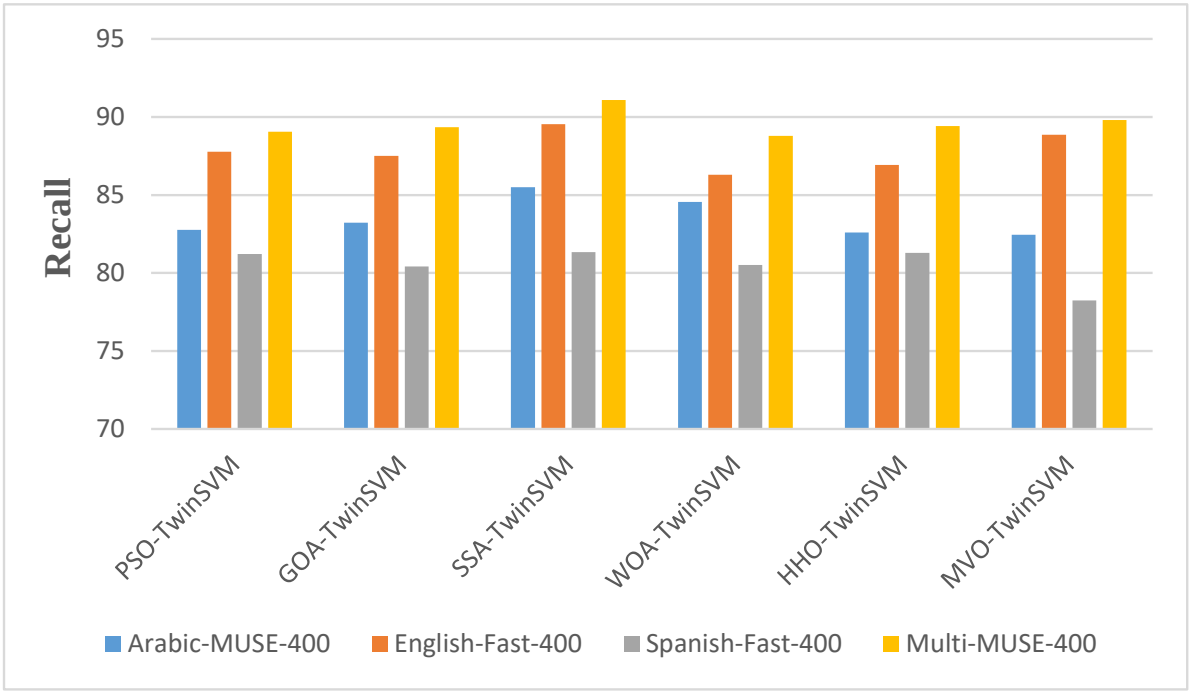


Fig. 6.5 Recall results of all algorithms for the best datasets of each language.

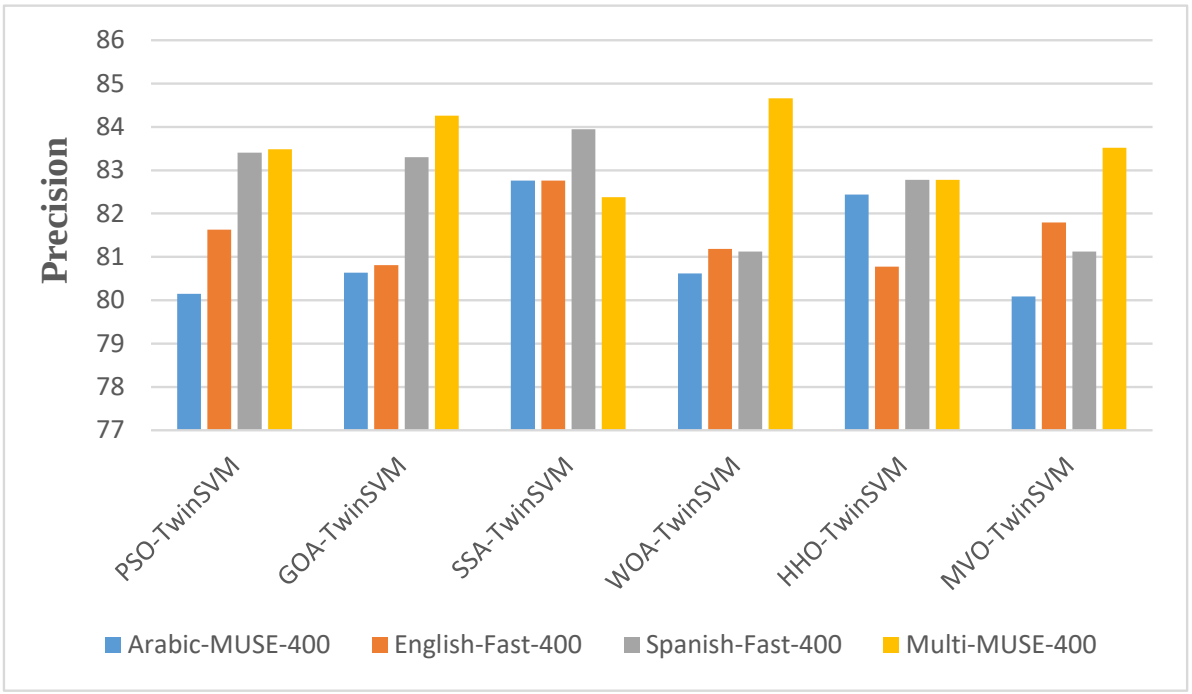


Fig. 6.6 Precision results of all algorithms for the best datasets of each language.

word embeddings that map target words to context words. The vector space of words is implicitly embedded with a sub-linear relationship due to a mapping between target words and context words. As a result, context plays a more critical role in understanding the relationship.

Therefore, we used three pre-trained word embedding techniques (MUSE, BERT, and FastText) to comprehend the multilingual context. Each of which uses a different method of representing words. Thus, it is noticed that the results have a slight variation in each method. For example, FastText achieved better results in the Arabic language in a 100-dimensional version than MUSE and BERT. In contrast, MUSE performs more efficiently with 400-dimensions of the same language. Additionally, BERT provided the best result for the 100-dimensions and a better result than FastText for the 400-dimensions in the multi-dataset. For the English data, FastText yielded the highest results in both dimensions. This type of variation shows that using different word representations is the right choice, due to differences in language characteristics and contexts.

The results show that SSA-TwinSVM outperforms all the other algorithms in the 400 dimensions data for the MUSE embedding technique and it beats most of the other algorithms in the 100 version. In addition, it gave the best results for almost all the other algorithms for both dimensions on different datasets using the BERT embedding technique. In addition, it outperforms the other algorithms for the Arabic dataset as well as the FastText embedding algorithm for both dimensions. In addition, it obtained the highest results from the English dataset of 400 version, as well as the Multilingual dataset of 100 version.

The current chapter examines the quality of TwinSVM and utilizes pre-trained word embeddings for sentiment analysis. The next chapter will propose a technique to enhance the detection of spam reviews using sentiment analysis. This involves employing a swarm-based algorithm to optimize TwinSVM hyperparameters and feature selection. By utilizing sentiment analysis to improve spam review detection, we aim to enhance review quality, achieve more accurate product evaluation, save time and resources, and build enhanced user trust. Thus, introducing an additional layer of credibility to assess the overall attitude conveyed in the reviews.

Furthermore, the next chapter's approach will address some drawbacks of the current method, such as dimensionality reduction, avoiding overfitting when dealing with high-dimensional data, computational efficiency, and handling noisy or redundant features. Additionally, we will incorporate the same pre-trained methods in the next study, as they have proven to be efficient in multilingual environments.

Chapter 7

Approach Three: Spam Reviews Detection through Sentiment Analysis

7.1 Introduction

Spam reviews can be divided into three categories:

- Misleading reviews: Falsely promoting, or demoting target objects by writing positive reviews.
- Brand-based reviews: An individual reviewing a product or service focuses on the brand, producer, or seller of the product, but does not address the product itself.
- Non-reviews: there are no opinions in the reviews, they are merely advertisements or questions related to the product.

It is possible for reviews to have a negative impact on concerned individuals, and they may lose their jobs or money if they decide to follow such reviews. Due to the fact that we cannot be certain that these reviews are accurate or trustworthy. It is therefore essential to address the issue of fake or spam reviews. In order to minimize the impact of this type of malicious activity that is reported to generate billions of dollars of benefits and damages, researchers proposed various types of detection approaches [82]. One of these techniques is by using sentiment analysis to improve spam reviews detection.

Sentiment analysis has been successfully applied in different detection approaches in the literature. For instance, as previously discussed in the first chapter, improving the ability to detect sarcasm, satire, and irony in informal and formal writing. Also, facilitates the detection of sexism, bullying, hate speech, harassment, and racism. Moreover, sentiment analysis is

also applied to enhance spam detection. Consequently, improving the detection of spam reviews through sentiment analysis can be an efficient and effective solution. Therefore, the combination of both methods can be defined through the utilization of sentiment analysis, adding an extra layer of authenticity to evaluate the overall attitude reflected in the reviews.

When it comes to one of the most effective machine learning techniques for text classification and particularly for spam reviews detection using sentiment analysis, SVM shows excellent performance. For instance, Eranpurwala et al. [78] proposed a study that used sentiment analysis techniques to detect spam reviews in online movie reviews based on SVM. They compare the SVM with other supervised machine learning algorithms including Naive Bayes, KStar (K*), K-Nearest Neighbors (KNN) and Decision Tree (DT). As a result of their experiments, it has been determined that the SVM algorithm achieves the highest accuracy, and it outperforms other algorithms. Moreover, Eranpurwala et al. conducted another study using SVM for spam reviews detection through sentiment analysis [79]. Unlike their previous paper, in this work, they used three different datasets for movie reviews. Also, as a performance measure, accuracy, precision, recall, and F-measure have been utilized. According to the results, SVM performed better than other algorithms in both text classification and the detection of spam reviews.

In addition, SVM and other machine learning techniques are combined with metaheuristic algorithms to improve their performance. There are many different types of metaheuristic algorithms in the literature, such as the WOA, SSA, and HHO. SVM can be improved through metaheuristic algorithms by searching for appropriate values for its hyperparameters.

Again as mentioned before SVM has several limitations that can adversely affect its performance if certain conditions are met. Accordingly, in order to resolve these limitations, we utilized TwinSVM. TwinSVMs are used for classification tasks as an extension of standard SVM algorithms. By utilizing two hyperplanes, it is possible to construct a "twin" structure that provides a more accurate representation of the data. As discussed before, TwinSVMs perform better than standard SVMs in handling imbalanced data sets. Aside from this, TwinSVM provides greater flexibility and efficiency when utilizing different kernel functions compared with standard SVM.

Thus, this chapter proposes a TwinSVM based on a swarm optimization algorithm and sentiment analysis to detect spam reviews. In order to optimize TwinSVM parameters and select features, the swarm algorithm HHO is used after several experiments.

As highlighted in the preceding chapters, reviews are not written in one language, three different languages are taken into consideration during the extraction of reviews, including English, Spanish, and Arabic. For word representation, we also used the pre-trained word embedding models (BERT, FastText, and MUSE) considering their efficiency and ability

to handle multilingual contexts as shown in the previous chapter. As a result, twenty-four datasets have been generated, each encompassing 100/400 dimensions in all languages, then they combined together to produce a multilingual dataset.

The experimental phase of this chapter consists of four parts. The first part applies standard SVM with various metaheuristic algorithms as an initial assessment. Then implementing the proposed HHO-TwinSVM-Fs as the second phase to compare two aspects; various metaheuristic algorithms and standard SVM. In the third experiment, the highest accuracy datasets of each language were used to investigate the performance of HHO-TwinSVM-Fs with other well-regarded classification models combined with HHO, including XGBoost (XGB), MLP, and k-NN. Further, sentiment analysis-based features were employed to improve the detection of spam reviews in the fourth phase. Eventually, a detailed analysis is presented of the results, reviews characteristics modifications, and sentiment impacts.

In order to gain a better understanding, the following points summarize the contribution of this chapter:

- A swarm-based Twin SVM approach is proposed for the detection of spam reviews.
- In order to improve the performance of the Twin SVM, HHO is used for parameter optimization and feature selection.
- The sentiment-based features as well as the sentiment dataset labels (from approach two) added as features and used to enhance the spam review detection phase.
- As before, three languages, English, Spanish, and Arabic, have been examined to address the multilingual problem.
- Again, a set of three pre-trained word embedding models (BERT, FastText, and MUSE) was applied, which provides better word representation and language handling than traditional methods.
- A total of 24 datasets have been created based on the three pre-trained models for the purpose of examining spam reviews from every perspective using sentiment features.
- As before, a deep analysis of reviews structure, style, and context (before and after COVID-19), results, and sentiment effects is proposed.

7.2 Approach three Methodology

In this section, we discuss three key points regarding how HHO is proposed to be implemented for the selection of features (attributes) and optimization of TwinSVM parameters. These

are the encoding scheme, fitness function, and system architectures used to represent HHO candidates' solutions. Following that, we discuss how the performance of the approach is evaluated. In the following, we describe the overall system architecture of the proposed model regarding each of these key design issues.

7.2.1 Solution representation: The encoding scheme

In order to select the features and optimize the Twin-SVM parameters, it is imperative to design the solution representation prior to using the HHO algorithm. In this study, it has been demonstrated that the architecture of individuals is encoded in the form of a vector of real numbers. In the case of each vector, the number of elements is equal to the number of features in the dataset, along with three additional elements to represent the parameters of the TwinSVM: the Cost1 (C_1), Cost2 (C_2) and the Gamma (γ). An example of how the encoding scheme was implemented is shown in Figure 7.1. The elements of the vector are randomly generated numbers in the interval [0,1]. As a result, element values that represent features are rounded to 1 if the element is more than or equal to 0.5, and are rounded to 0 if it is smaller or equal to 0.5. The element equals 1 means that the feature is selected, otherwise, it means that the feature is not selected. Since C_1, C_2, and γ have different search spaces, their parameters must be scaled differently. In this case, the elements corresponding to C_1 and C_2 have a value that is mapped to the interval [-2,2], whereas γ has a value that is mapped to the interval [-8,2]. Based on the linear transformation, previously employed in preceding chapters, Eq. 5.1, we obtain the values of C_1, C_2, and γ. To put it in another way, HHO generates elements that encompass the three parameters plus the D number of features for each dataset. These values have been combined into a one-dimensional vector of D + 3 real numbers.

7.2.2 Fitness evaluation

A predetermined measurement is used to evaluate the quality of each generated individual. In order to accomplish this, a fitness function must be defined. Throughout each iteration, HHO individuals are assessed using this fitness function. For the purpose of evaluating the fitness function, the classification accuracy of the TwinSVM classifier is used as the previous chapter, which can be calculated with Eq. 5.2 that has already been utilized in preceding chapters:

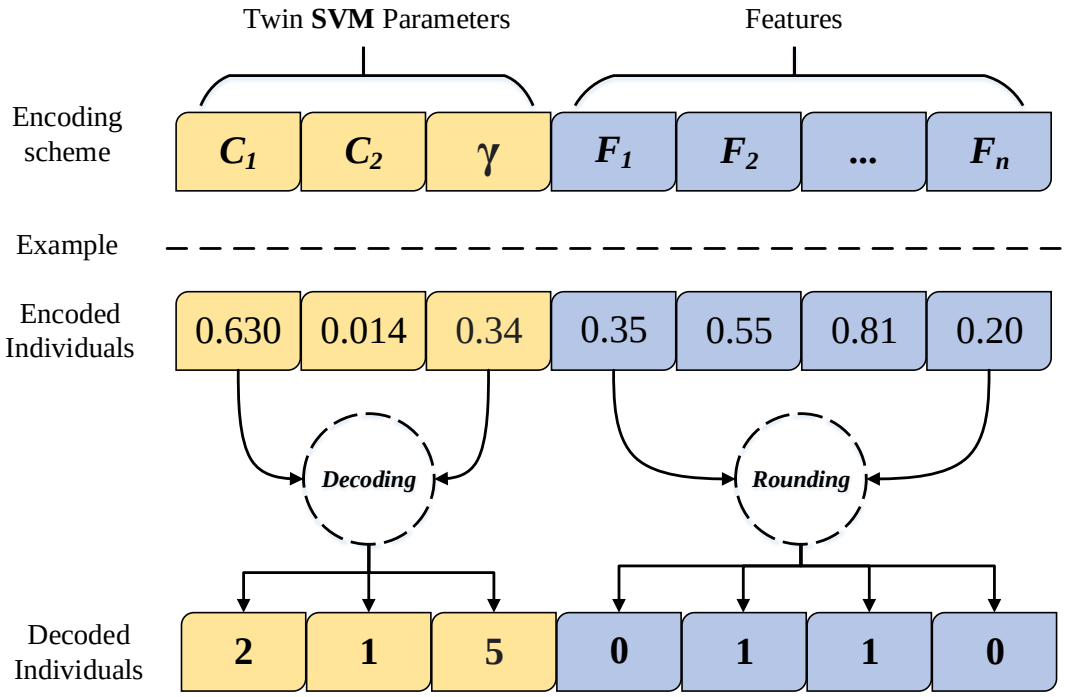


Fig. 7.1 Twin SVM individuals representation.

7.2.3 An overview of system architectures

In this subsection, we outline the primary systems architectures that are employed to conduct parameter optimization for TwinSVMs and feature selection simultaneously (for efficient performance) by utilizing the HHO algorithm. In previous studies, the term "system architecture" was used to describe the sequence of processes followed to complete this task.

Following are the bullet points and figures that describe the workflow of this architecture. Akin to the previous chapters, numerical markers (***{1}, {2}, {3}***) been included to the method figure (Fig. 7.2) and in the text.

- Data normalization: In this step, all datasets' features are preprocessed. To prevent the learning process of the algorithm from being affected by some features that have different range values, all values are mapped to the same scale. Then, we use a special form of Eq. 5.1, Eq. 7.1, which equalizes all features' weights and normalizes them to fall in the interval [0,1].

$$B = \frac{A - min_A}{max_A - min_A} \tag{7.1}$$

- Splitting criteria: We begin by dividing every dataset into training and testing sets in order to implement our proposed approach ***{1}***. Depending on the number of experiments, the splitting criterion is determined. Therefore, the dataset for k experiments is partitioned into k parts, where $k-(1/k)$ portions are allocated for training, and the remainder $(1/k)$ is assigned for the testing phase. Thus, to ensure that the model produced is as reliable as possible, it is essential to guarantee maximum diversity both in the training and testing sets.

- Decoding of individuals: HHO generates vectors individuals in two parts: the first three elements of the vector (TwinSVM parameters) are converted using Equation 5.1. A binary vector is formed from the remaining elements, which correspond to the features that were selected ***{2}***.

- Feature selection: The training dataset is used to select the corresponding features after decoding the individuals into a binary vector as described previously ***{2}***.

- Fitness evaluation: Based on the fitness function, HHO assesses each generated solution that is based on the parameters of TwinSVM and a subset of selected features ***{3}***.

- Termination criteria: Until an appropriate termination condition is met, the evolutionary cycle of HHO continues. The termination condition in this case is the maximum number of iterations ***{4}***.

- Reproduction operators: The following is a sequence of operators that are applied by HHO to develop a better quality solution for the generated individuals ***{5}***.

As soon as the maximum number of iterations has been reached, the best individuals produced by HHO are used for testing ***{6}***. At the end, all previous processes are repeated k times, and the average values are taken into account. Figure 7.2 illustrates all the previous steps.

7.2.4 Evaluation measures

At this stage, we will evaluate TwinSVM's performance as well as the performance of the other approaches. In order to assess the problem of review detection, several measurements will be taken, such as accuracy, recall, precision, and f-measure. The confusion matrix is also used to calculate all measures (described in Chapter 5 (Table 5.1)). Based on this assumption, each evaluation measure is described as follows:

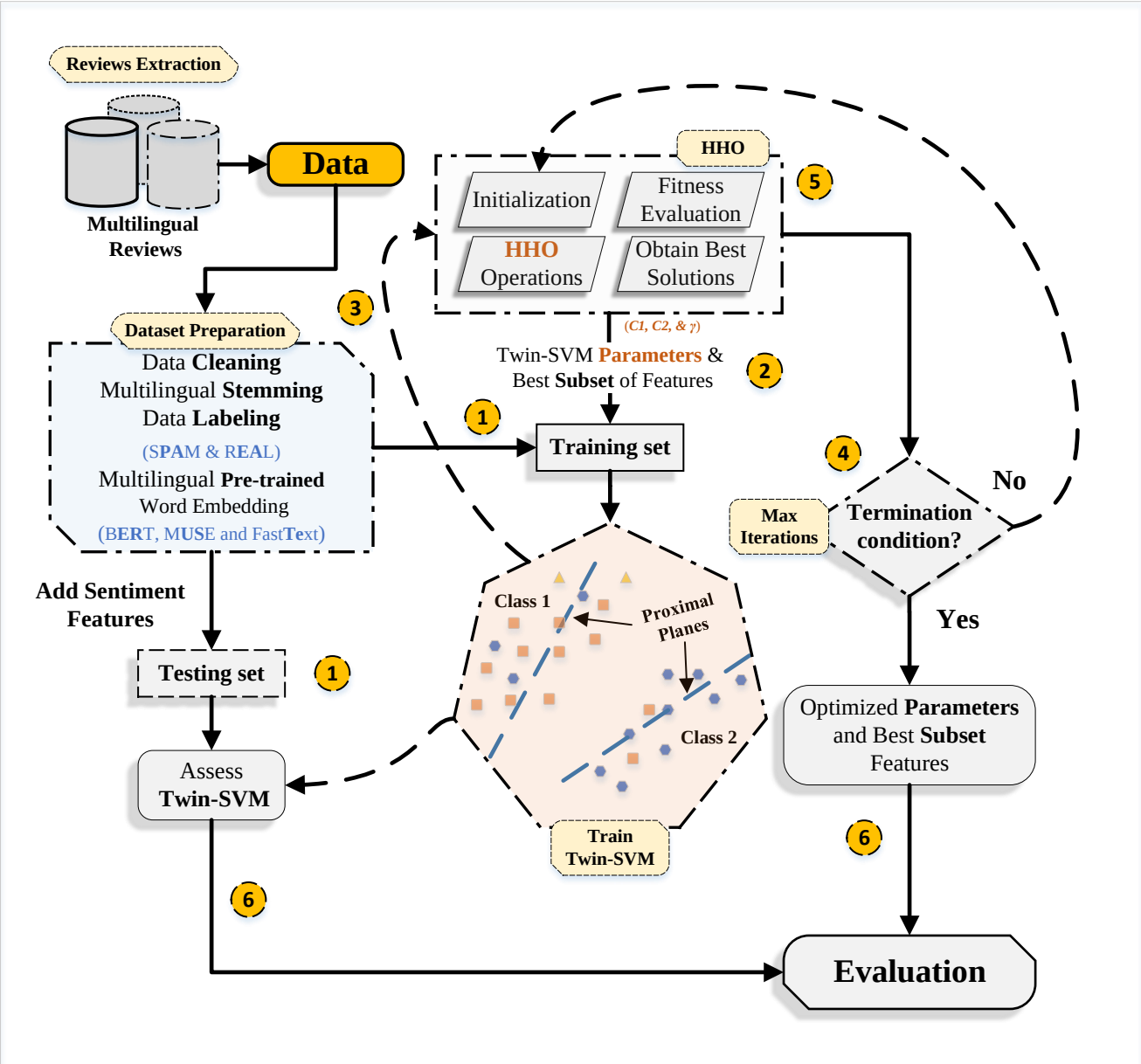


Fig. 7.2 The proposed HHO-TwinSVM process. The numerical annotations in the figure serve as a mapping mechanism between the text and the methodology figure.

- Accuracy: The number of spam and real reviews that were correctly classified is divided by the total number of reviews (Equation5.3).
- *Precision*: Based on the percentage of spam reviews that have been classified correctly overall anticipated spam reviews (Equation 6.2).
- *Recall*: can be measured by taking the number of correctly anticipated spam reviews divided by the total number of spam reviews (Equation 6.1).
- F-measure: This measure is defined as the harmonic mean of precision and recall, which can be calculated using the following equation:

$$F - measure = \frac{2 * (precision * recall)}{precision + recall} \tag{7.2}$$

7.3 Approach three Experiment and Results

Throughout this section, all results and experiments are presented, in detail, in order to demonstrate the performance of different algorithms on a variety of datasets. Four different experiments are described. Firstly, standard SVM experiments were conducted using various metaheuristic algorithms for feature selection. Secondly, the proposed HHO-TwinSVM-Fs approach integrated with various metaheuristic algorithms is presented. Thirdly, HHO-TwinSVM-Fs is compared with well-known classification models (XGBoost, MLP, and k-NN) combined with different metaheuristic algorithms. Finally, new features based on sentiment analysis are added to the original data. Furthermore, a discussion of the review analysis and the significance of the features is provided.

7.3.1 Experimental Setup

To ensure a fair comparison, all tests in this study were conducted under the same conditions. Table 7.1 provides the specifications of the personal computer we used. Both SVM and TwinSVM classifiers are implemented using the LIBSVM library. The training and testing sets are further validated by 10-fold cross-validation. The overall experiments were conducted on the 30 iterations and have been chosen after various runs to find the optimal solution. Then the average and standard deviation were taken.

In this third approach, the labels are spam and real, and then the labels from the second approach datasets are employed as features to improve the detection performance. So, in other words, both the sentiment-based features alongside the sentiment dataset labels are used as features for spam reviews detection.

Additionally, Table 7.2 shows the initial parameters of the metaheuristic algorithms applied in this study.

Table 7.1 The detailed settings of the used system.

Name	Settings
Hardware	
CPU	AMD Ryzen 5 5600X processor
Frequency	3.7GHz
# of Cores	6
RAM	16GB
Hard drive	1 TB
Software	
Operating system	Windows 10 (64-bit)
Language	Python 3
Application	Visual Studio

Table 7.2 A description of the initial parameters used for the optimization metaheuristic algorithms.

Algorithm	Parameter	Value
All	Iterations	30
MVO	Minimum wormhole existence ratio	0.2
	Maximum wormhole existence ratio	1
PSO	Acceleration constants	$[2.1, 2.1]$
	Inertia w	$[0.9, 0.6]$
GWO	α	Min=0, and Max=2
WOA	a	2 to 0
	r	[0, 1]
SSA	c_1	[0-1]
	c_2	[0-1]

The following points provide a more detailed description of the experiments and analyses that will be conducted:

- Experiment I: In this experiment, as part of our analysis of preliminary results and the performance of classification algorithms on the data, we applied different metaheuristic algorithms accompanied by original SVM with feature selection (SVM-Fs).
- Experiment II: This experiment examines the performance of the proposed HHO-TwinSVM-Fs approach and compares it with various applied metaheuristics. Thus,

demonstrating the superiority of the TwinSVM against other metaheuristic algorithms as well as the original SVM-Fs.

- Experiment III: A performance comparison of the proposed HHO-TwinSVM-Fs approach with other well-known classifiers (XGBoost, k-NN, and MLP) combined with metaheuristic algorithms is presented in this experiment.
- Experiment IV: In the final experiment, we study the improvement of spam review detection by integrating novel features based on sentiment analysis.

7.3.2 Experiment I: Comparison of standard SVM-Fs with different metaheuristic algorithms

In this experiment, we applied standard SVM combined with different metaheuristic algorithms on 24 datasets. Further, we performed parameter optimization and feature selection with the help of wrapper-based methods (e.g. metaheuristic algorithms). Tables 7.3, 7.4 and 7.5 present the accuracy results of the standard SVM along with the other metaheuristic algorithms (PSO, GOA, SSA, WOA, HHO and MVO) on the three versions of the datasets (BERT, FastText, and MUSE).

According to the average accuracy values shown in Table 7.3, HHO-SVM has outperformed the other metaheuristics and obtained the best performance on five out of the eight datasets. As with the Spanish BERT-100 and Multi BERT-100/400 datasets, HHO-SVM yields very competitive results in terms of accuracy. For example, HHO-SVM performed better than the five other algorithms on the Arabic BERT-100/400 datasets. Also, HHO-SVM obtained the highest results for the two English datasets (BERT-100/400) with 78.2077% and 78.7967%, respectively. As for the Spanish datasets, HHO-SVM achieved the best accuracy for the BERT-400, while the BERT-100 acquired the highest results by SSA-SVM. On the Multi datasets, MVO-SVM and PSO-SVM obtained the best accuracies for BERT-100 and BERT-400, respectively.

Regarding the FastText version, Table 7.4 illustrates the accuracy results of the Arabic, English, Spanish, and Multi datasets. HHO-SVM performed better than other algorithms in four out of eight datasets. While the remaining four datasets (English Fast-400, Spanish Fast-100, and both versions of the Multi dataset) achieved the best results by different algorithms. For example, GWO-SVM has the highest accuracy for the English Fast-400, PSO-SVM outperformed other algorithms for the Spanish Fast-100 dataset, and WOA-SVM and SSA-SVM have the best accuracy for the Multi datasets of Fast-100 and Fast-400, respectively.

Table 7.3 Accuracy results of standard SVM using various metaheuristic algorithms for Arabic, English, Spanish, and Multilingual datasets based on the BERT method.

Datasets			PSO-SVM	GWO-SVM	SSA-SVM	WOA-SVM	HHO-SVM	MVO-SVM
Arabic	BERT-100	Avg	87.2203	87.4262	92.5418	89.3480	**93.6056**	87.2272
		Std	4.0713	3.6934	3.6166	2.4943	**1.3367**	4.9309
	BERT-400	Avg	87.2180	87.2203	87.2203	87.2249	**87.2318**	87.2226
		Std	3.4718	4.6912	3.7228	4.4962	**5.8995**	4.2884
English	BERT-100	Avg	78.1072	78.1087	78.1036	78.1097	**78.2077**	78.1082
		Std	2.4821	2.9000	2.1009	3.8825	**2.7925**	3.5962
	BERT-400	Avg	76.2760	75.3136	75.0155	73.8507	**78.7967**	78.6478
		Std	3.0809	3.0742	2.9664	1.8226	**2.1701**	1.3868
Spanish	BERT-100	Avg	64.8172	64.7160	**65.1668**	61.9177	63.9686	61.9192
		Std	3.8223	3.5484	**1.9756**	1.6758	1.2715	2.1682
	BERT-400	Avg	55.0238	56.5736	58.1718	58.0703	**58.8214**	53.6731
		Std	2.3551	2.7126	1.6306	2.0822	**1.3265**	2.0668
Multi	BERT-100	Avg	58.9719	66.3961	62.9293	66.3177	67.7730	**70.1126**
		Std	4.1496	3.2682	3.3046	1.0647	1.7714	**2.0945**
	BERT-400	Avg	**70.2857**	64.8681	65.6664	66.4754	68.1755	65.5841
		Std	**5.7421**	5.5052	2.4288	4.4457	2.5422	3.0596

Table 7.4 Accuracy results of standard SVM using various metaheuristic algorithms for Arabic, English, Spanish, and Multilingual datasets based on the FastText method.

Datasets			PSO-SVM	GWO-SVM	SSA-SVM	WOA-SVM	HHO-SVM	MVO-SVM
Arabic	Fast-100	Avg	89.7621	87.2226	88.2727	87.2226	**90.1990**	87.2226
		Std	2.2082	4.3927	1.1665	5.3668	**6.0999**	4.1270
	Fast-400	Avg	87.2226	87.2249	87.2203	88.9156	**88.9202**	88.7028
		Std	4.8809	5.8908	3.7228	1.0571	**2.2664**	3.2581
English	Fast-100	Avg	74.3049	74.2059	73.9025	72.2854	**76.6805**	72.6403
		Std	2.3104	1.7732	0.7259	2.8521	**3.2025**	1.7066
	Fast-400	Avg	74.9648	**78.7979**	73.9506	76.4272	78.0407	75.9718
		Std	3.4148	**1.6827**	2.4472	1.4994	2.2870	1.8709
Spanish	Fast-100	Avg	**66.0172**	65.5667	63.1209	64.7691	65.3667	63.7647
		Std	**2.3257**	2.1373	4.3964	1.6769	2.3086	6.8374
	Fast-400	Avg	64.0675	65.3677	60.5713	56.9234	**65.9658**	55.1227
		Std	1.1914	3.7863	2.1439	2.3578	**2.5117**	2.0347
Multi	Fast-100	Avg	63.4086	63.4847	61.1480	**65.9106**	63.0061	63.9725
		Std	2.2659	3.0635	1.8773	**2.6116**	1.4247	2.4499
	Fast-400	Avg	68.4214	68.9017	**70.3562**	68.3391	69.5449	68.5830
		Std	3.3829	1.4630	**2.4835**	2.5500	3.2044	3.1384

Further, Table 7.5 shows the average accuracy results of SVM combined with the metaheuristic algorithms for the MUSE version. HHO-SVM achieved better performance than the other algorithms in six out of eight datasets. For instance, regarding the Arabic language, HHO-SVM has the best results with 87.6389% and 87.2295% for MUSE-100 and MUSE-400 datasets, respectively, while in the English language, it achieved the highest accuracy for the MUSE-100 dataset. In addition, for the Spanish MUSE-100 dataset, HHO-SVM obtained the most accurate results with 68.1171%. With regard to the Multi, both datasets (MUSE-100/400) also yielded the highest results by SSA-SVM. On the other hand, WOA-SVM outperformed the other algorithms for the English MUSE-400 dataset and Spanish MUSE-400.

Table 7.5 Accuracy results of standard SVM using various metaheuristic algorithms for Arabic, English, Spanish, and Multilingual datasets based on the MUSE method.

Datasets			PSO-SVM	GWO-SVM	SSA-SVM	WOA-SVM	HHO-SVM	MVO-SVM
Arabic	MUSE-100	Avg	87.2203	87.2226	87.2226	87.2203	**87.6389**	87.2249
		Std	3.9585	4.7872	5.0181	3.2696	**1.6958**	7.5421
	MUSE-400	Avg	87.2249	87.2226	87.2249	87.2157	**87.2295**	87.2203
		Std	5.8135	4.4946	5.2401	3.8451	**5.3313**	4.6912
English	MUSE-100	Avg	83.1892	82.0285	72.1876	83.3416	**83.3927**	82.7863
		Std	1.4799	2.7218	2.0993	1.5825	**0.9672**	1.9795
	MUSE-400	Avg	78.0420	76.4265	84.4021	**85.4621**	84.8051	81.8277
		Std	1.0816	1.1264	1.3078	**1.3209**	0.9175	1.3594
Spanish	MUSE-100	Avg	58.5741	66.5180	65.8170	66.3691	**68.1171**	60.8707
		Std	4.1364	2.2992	1.7587	3.0807	**1.6312**	1.2327
	MUSE-400	Avg	70.9632	68.6667	69.7173	**71.0171**	70.6163	69.7653
		Std	1.9502	1.7304	2.4611	**3.5912**	2.1111	0.9543
Multi	MUSE-100	Avg	76.8170	72.5349	71.0840	78.5105	**79.4038**	71.4843
		Std	3.9213	3.2888	2.5989	2.7712	**2.5380**	1.9494
	MUSE-400	Avg	79.8083	78.2712	72.0533	79.6441	**80.2929**	73.1817
		Std	1.3435	1.3140	1.8228	1.3969	**1.9305**	2.9708

Overall, HHO-SVM achieved the highest accuracy in 15 out of 24 datasets, while for the remaining datasets, the algorithm showed competitive results. Among the three-word embedding techniques, BERT and FastText achieved the best results in the Arabic 100 and 400 datasets. On English, Spanish, and Multi datasets, the MUSE technique has the best results compared to the other word embedding methods.

7.3.3 Experiment II: Comparison of the proposed HHO-TwinSVM-Fs and other metaheuristic algorithms

As part of this experiment, HHO-TwinSVM-Fs will be compared to PSO, GOA, SSA, WOA and MVO in order to perform feature selection and optimize Twin-SVM parameters. Based on the 24 datasets described previously, six algorithms are evaluated. Tables 7.6, 7.7, 7.8 and 7.9 present the average accuracy rate as well as the best number of selected features.

As shown in Table 7.6, the proposed HHO-TwinSVM-Fs is compared against other similar approaches in terms of accuracy. In general, it can be observed that the HHO-TwinSVM-Fs approach has better performance than other algorithms in five out of eight datasets. For example, the best algorithm in terms of accuracy for the Arabic and English BERT-100/400 datasets was HHO-TwinSVM-Fs with 92.9741%, 89.1166%, 81.6787% and 85.3954%, respectively. Regarding the Spanish language, only the BERT-100 dataset achieved the best accuracy with HHO-TwinSVM-Fs, whereas the highest accuracy was with SSA-TwinSVM-Fs for the BERT-400 dataset. In addition, WOA-TwinSVM-Fs has reached to the finest accuracy rates for the Multi BERT-100 and BERT-400 datasets.

Moreover, in comparison with standard SVM, the TwinSVM-Fs show better performance on average accuracy on all datasets except the Arabic BERT-100 dataset. In more detail, the differences for English BERT-100/400 were 3.4710% and 6.5992%, respectively. In the other languages, the values were greatly increased by 6.9050%, 18.0038%, 9.3954%, and 10.4235% for Spanish BERT-100/400 and Multi BERT-100/400 datasets, respectively. On the other hand, in the Arabic BERT-100 dataset, the difference was in favor of standard SVM, while the BERT-400 was in favor of TwinSVM-Fs.

Table 7.6 Accuracy results of HHO-TwinSVM-Fs and other metaheuristic algorithms for Arabic, English, Spanish and Multilingual datasets based on the BERT method.

Datasets			PSO-TwinSVM	GWO-TwinSVM	SSA-TwinSVM	WOA-TwinSVM	HHO-TwinSVM	MVO-TwinSVM
Arabic	BERT-100	Avg	88.5060	88.4829	89.1119	90.6059	**92.9741**	91.0407
		Std Dev	6.7361	6.0418	3.6119	5.0754	**4.1355**	3.1546
	BERT-400	Avg	88.2794	88.6957	88.7142	88.2794	**89.1166**	88.7003
		Std Dev	3.9004	5.6869	5.1051	4.7176	**5.6515**	2.0117
English	BERT-100	Avg	80.9112	81.0928	81.3959	81.0827	**81.6787**	76.4899
		Std Dev	1.0741	1.9483	1.5157	1.4036	**1.6740**	4.3290
	BERT-400	Avg	83.1026	81.1229	85.0419	85.1631	**85.3954**	77.2862
		Std Dev	1.7913	2.3854	0.7109	0.8105	**1.0268**	5.3709
Spanish	BERT-100	Avg	78.5602	77.4097	79.3577	79.2102	**79.5080**	79.1592
		Std Dev	3.0263	7.9966	3.5249	1.2767	**2.4882**	2.6133
	BERT-400	Avg	53.7746	48.0774	**80.7092**	77.4575	74.6102	61.5580
		Std Dev	11.5642	2.6378	**2.3275**	12.2695	14.4718	17.7555
Multi	BERT-100	Avg	70.6167	70.0993	71.2475	**72.0718**	71.8535	70.1967
		Std Dev	1.3838	1.1117	1.5470	**1.5177**	1.4413	4.9054
	BERT-400	Avg	74.1332	74.3432	76.8166	**76.8252**	76.7844	71.5683
		Std Dev	1.4020	2.1918	0.9954	**1.6908**	0.9291	4.1725

Based on the results in Table 7.7, it is revealed that the proposed HHO-TwinSVM-Fs also attained the best average accuracy compared to the other algorithms in five out of eight of the FastText datasets. For instance, HHO-TwinSVM-Fs has the maximum classification accuracy on Arabic Fast-100, English Fast-100/400, Spanish Fast-100, and Multi Fast-100 with 91.2581%, 81.4665%, 85.3751%, 79.6604% and 69.2668%, respectively. However, according to accuracy results for Fast-400 datasets for Arabic, Spanish, and Multi languages, the highest accuracy rates were obtained by different algorithms, which are GWO-TwinSVM, HHO-TwinSVM, and SSA-TwinSVM, respectively.

Further, we notice that TwinSVM-Fs boosts the results for all datasets compared to standard SVM-Fs. As an example, the values increased by 1.0591% and 0.2056% for both datasets of the Arabic language, while for the English datasets (Fast-100 and Fast-400) the accuracy level has boosted up by 4.7860% and 6.5772%, respectively. Additionally, the proposed TwinSVM-Fs significantly improved the classification accuracy for the Spanish Fast-100/400 datasets with 13.6432% and 15.2954, respectively, while the difference was 3.3562% and 5.6847% for the Multi Fast-100/400 datasets, respectively.

Table 7.7 Accuracy results of HHO-TwinSVM-Fs and other metaheuristic algorithms for Arabic, English, Spanish, and Multilingual datasets based on the FastText method.

Datasets			PSO-TwinSVM	GWO-TwinSVM	SSA-TwinSVM	WOA-TwinSVM	HHO-TwinSVM	MVO-TwinSVM
Arabic	Fast-100	Avg	89.9722	88.4736	89.7549	90.4117	**91.2581**	89.7641
		Std Dev	6.0369	4.0946	4.3886	3.3498	**4.6447**	2.6207
	Fast-400	Avg	89.1119	**89.1258**	88.2794	89.0981	87.8446	89.1212
		Std Dev	4.4819	**3.6776**	4.7176	6.2545	4.0241	2.7605
English	Fast-100	Avg	81.0221	81.2644	81.3351	81.4263	**81.4665**	80.8100
		Std Dev	1.8276	1.6634	1.4841	1.6376	**0.6870**	1.2252
	Fast-400	Avg	83.6587	82.7489	85.3247	84.9913	**85.3751**	84.1934
		Std Dev	3.2775	3.6573	0.7155	1.3580	**0.8350**	2.6792
Spanish	Fast-100	Avg	78.3095	70.3124	78.4612	78.4587	**79.6604**	78.5587
		Std Dev	2.8308	14.0348	1.7328	3.7555	**2.2460**	3.9429
	Fast-400	Avg	73.0607	59.8597	80.0077	**81.2612**	80.3580	66.9244
		Std Dev	13.4986	16.8788	4.9688	**3.3960**	2.4633	16.8727
Multi	Fast-100	Avg	68.3049	68.2891	69.1214	69.2183	**69.2668**	69.0891
		Std Dev	1.6423	1.8305	1.4860	1.4957	**1.6377**	1.4769
	Fast-400	Avg	74.7392	73.7693	**76.0409**	75.8387	75.8385	75.3859
		Std Dev	1.8161	1.5910	**1.3763**	1.2211	1.7285	2.1707

According to Table 7.8, HHO-TwinSVM-Fs has again achieved the best average accuracy among the other competitors in five out of eight datasets, followed by SSA-TwinSVM-Fs, and WOA-TwinSVM-Fs. For example, HHO-TwinSVM-Fs have the maximum accuracy in Arabic MUSE-100/400, English MUSE-400, Spanish MUSE-400 and Multi MUSE-400 with 87.2155%, 78.2109%, 89.0314%, 72.5639%, and 85.0859%, respectively, whereas MUSE-100 for English, Spanish and Multi ranked 1st by WOA-TwinSVM-Fs and SSA-TwinSVM-Fs, respectively.

In comparison with the standard SVM-Fs, the TwinSVM-Fs have the highest accuracy in five datasets, namely, English MUSE-100/400, Spanish MUSE-400, and Multi MUSE-100/400 with a difference of 5.0140%, 3.5693%, 1.5468%, 4.5828% and 4.79300%, respectively. As for the reaming datasets (Arabic MUSE-100/400 and Spanish MUSE-400), the standard SVM-Fs outperformed the TwinSVM-Fs with differences of 0.4234%, 0.0186, and 0.0542%, respectively. With respect to the datasets favoring standard SVM, the difference is very small over when TwinSVM-Fs had the best results.

Table 7.8 Accuracy results of HHO-TwinSVM-Fs and other metaheuristic algorithms for Arabic, English, Spanish, and Multilingual datasets based on the MUSE method.

Datasets			PSO-TwinSVM	GWO-TwinSVM	SSA-TwinSVM	WOA-TwinSVM	HHO-TwinSVM	MVO-TwinSVM
Arabic	MUSE-100	Avg	86.7715	86.9889	86.9843	87.1970	**87.2155**	86.9981
		Std Dev	3.4927	4.9744	4.2309	3.5264	**4.2351**	5.3390
	MUSE-400	Avg	86.7715	86.9796	87.0028	86.9935	**87.2109**	86.9843
		Std Dev	3.6339	5.0184	5.6087	3.0824	**2.4356**	5.4747
English	MUSE-100	Avg	87.7286	87.4961	88.0718	**88.2435**	87.5872	87.9305
		Std Dev	0.8113	1.2155	1.3427	**0.9351**	0.5690	0.7761
	MUSE-400	Avg	88.7083	88.6577	86.6228	88.8597	**89.0314**	88.9809
		Std Dev	1.2104	1.1027	2.5007	0.9762	**1.1229**	0.8283
Spanish	MUSE-100	Avg	66.1172	65.9159	**68.0629**	68.0159	67.4172	66.8182
		Std Dev	3.4192	3.1286	**3.4686**	4.5034	1.9083	4.0951
	MUSE-400	Avg	69.3637	72.1102	72.0647	69.8142	**72.5639**	70.5664
		Std Dev	4.1218	3.9640	2.5230	4.1446	**3.2505**	2.9266
Multi	MUSE-100	Avg	83.2106	82.4992	**83.9866**	83.4370	83.5744	83.5099
		Std Dev	1.0048	1.2179	**1.3924**	2.0008	0.9200	1.0737
	MUSE-400	Avg	84.4878	84.0432	84.6010	84.1806	**85.0859**	84.9243
		Std Dev	1.0601	1.2294	1.7565	1.3168	**0.8379**	1.3537

As for the best-selected features, which the least the better, HHO-TwinSVM-Fs outperformed the other algorithms in most datasets as shown in Table 7.9. HHO-TwinSVM-Fs obtained the best features in twelve out of 24 datasets, followed closely by the GWO-TwinSVM-Fs and MVO-TwinSVM-Fs algorithms, respectively. Based on the results of the BERT method, we can see that HHO-TwinSVM-Fs attained the best-selected features in all languages except the Arabic language. For the FastText version, we can observe that the results were distributed across several algorithms: HHO-TwinSVM-Fs achieved the best results in Spanish Fast-100 and Multi Fast-100/400, GWO-TwinSVM-Fs in Arabic Fast-100/400, MVO-TwinSVM-Fs in English Fast-100/400, and the remaining Spanish Fast-400 was obtained by WOA-TwinSVM-Fs. As per the results for the MUSE version, the best algorithm was GWO-TwinSVM-Fs, followed by HHO-TwinSVM-Fs, MVO-TwinSVM-Fs, and PSO-TwinSVM-Fs, respectively.

Thus, and according to the results shown in Table 7.9, the proposed HHO-TwinSVM-Fs is capable of selecting the best subset of features while providing excellent performance in terms of accuracy as well.

Table 7.9 Comparison of obtained number of features using HHO-TwinSVM-Fs with different selection algorithms for all datasets.

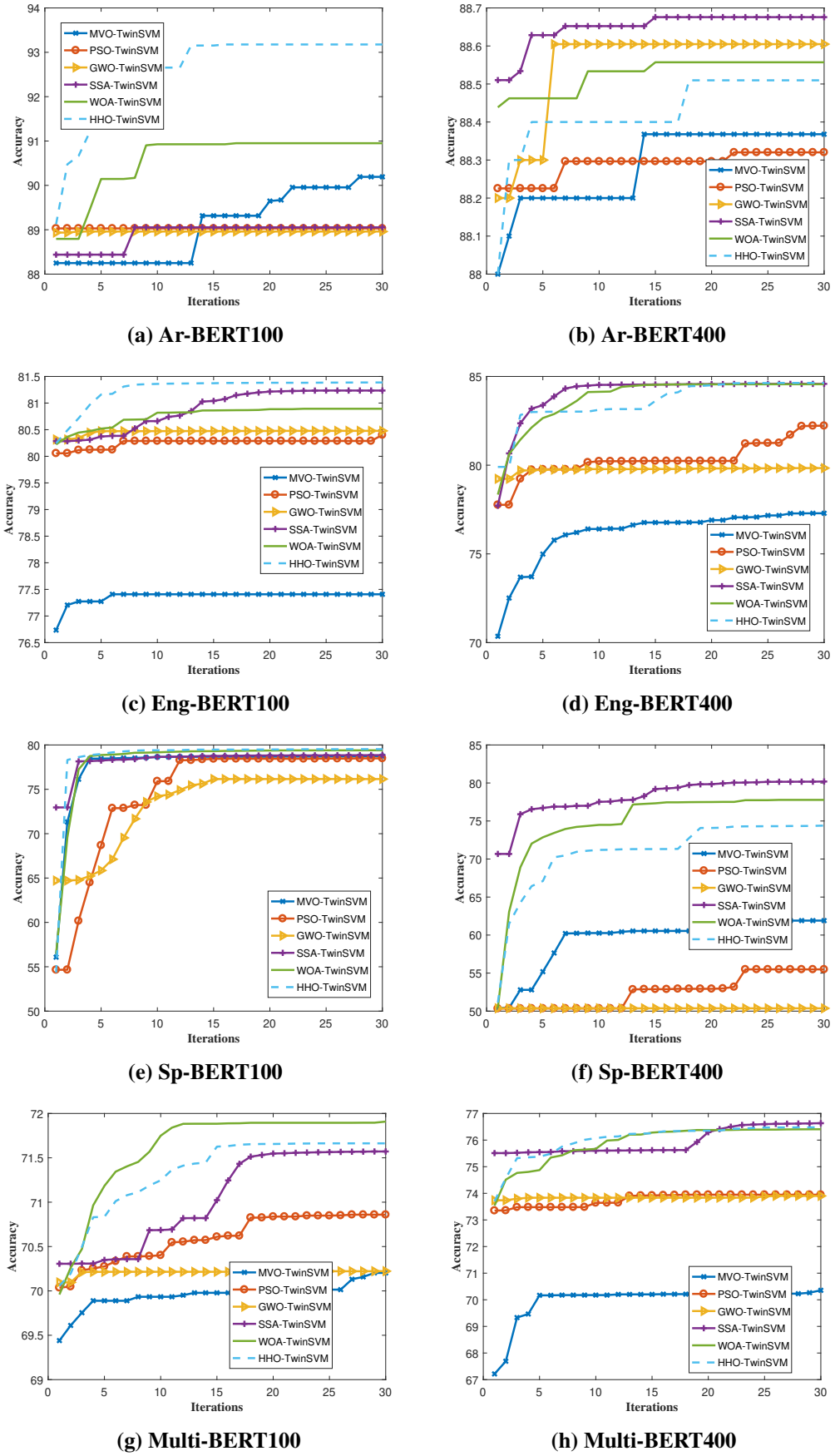
Datasets		PSO-TwinSVM	GWO-TwinSVM	SSA-TwinSVM	WOA-TwinSVM	HHO-TwinSVM	MVO-TwinSVM
BERT							
Arabic	BERT-100	51	**47**	48	68	49	58
	BERT-400	192	**179**	192	288	244	192
English	BERT-100	46	51	50	59	**20**	55
	BERT-400	188	181	210	169	**145**	198
Spanish	BERT-100	47	48	53	54	**39**	52
	BERT-400	213	162	210	61	**49**	203
Multi	BERT-100	48	46	40	68	**20**	49
	BERT-400	211	159	189	196	**168**	206
FastText							
Arabic	Fast-100	60	**39**	54	42	52	40
	Fast-400	182	**168**	193	215	173	187
English	Fast-100	53	50	55	51	56	**45**
	Fast-400	201	171	193	219	268	**200**
Spanish	Fast-100	47	78	56	49	**37**	55
	Fast-400	176	149	211	**72**	97	204
Multi	Fast-400	48	58	55	81	**43**	47
	Fast-100	201	268	193	219	**171**	200
MUSE							
Arabic	MUSE-100	51	46	46	59	85	**42**
	MUSE-400	199	**174**	216	285	184	209
English	MUSE-100	62	39	47	95	**13**	54
	MUSE-400	213	**177**	203	355	**177**	197
Spanish	MUSE-100	51	**42**	54	72	51	51
	MUSE-400	191	**159**	204	239	248	191
Multi	MUSE-100	**44**	50	51	69	48	50
	MUSE-400	216	140	195	311	**105**	199

Figures 7.3, 7.4 and 7.5 illustrate the convergence curve of HHO-TwinSVM-Fs and other algorithms. HHO-TwinSVM-Fs shows a competitive convergence behavior, compared to the other algorithms for all datasets. In the BERT dataset, for example, the HHO-TwinSVM-Fs have excellent convergence in 5 out of 8 datasets, especially in Ar-BERT100, Eng-BERT100, Eng-BERT400, Sp-BERT100, and Multi-BERT400. In the other datasets (Ar-BERT400, Sp-BERT400, and Multi-BERT100) we can see that it shows a competitive convergence. As for the FastText version, the HHO-TwinSVM-Fs has good convergence in all datasets except Ar-Fast400 and Multi-Fast400, however, it displays competitive convergence in about of them. With respect to the final version, MUSE, most datasets show satisfactory convergence for the HHO-TwinSVM-Fs excluding, Eng-MUSE100 and Multi-MUSE100.

Based on the results shown in Tables 7.6, 7.7 and 7.8, the proposed approach (HHO-TwinSVM-Fs) outperforms the other algorithms in 15 out of 24 datasets in terms of accuracy. In comparison with standard SVM-Fs, TwinSVM-Fs had the highest accuracy in 19 out of 24 datasets. It is noteworthy that the datasets for which standard SVM-Fs performed better have a very small difference in comparison to the datasets for which TwinSVM-Fs performed better, where the differences were more significant.

Moreover, according to the best version for each language, BERT-100 for Arabic, Fast-400 for Spanish, and MUSE-400 for English and Multi. These datasets have been used for further analysis and examination using other measures, namely recall, precision, and f-measure. Given the critical significance of this comparison, alternative metrics such as recall, precision, and f-measure are incorporated to ensure a comprehensive evaluation of the algorithm's overall efficacy. The goal of this critical comparison is to identify the most effective algorithms among several when combined with TwinSVM for detecting spam reviews.

Figures 7.6, 7.7 and 7.8 show the comparison results of HHO-TwinSVM-Fs and the other algorithms. As it can be seen in the figures, HHO-TwinSVM-Fs outperforms the other algorithms in terms of recall and f-measures in most datasets (Arabic BERT-100, English, and Multi MUSE-400), whereas for the precision measure, it has competitive results. Thus, it proves that HHO-TwinSVM-Fs effectively identifies positive instances while minimizing false positives, and achieving an overall balance between precision and recall. This happened due to the algorithm's sensitivity in capturing true positive instances and minimizing false negatives, resulting in higher recall and overall F-measures. However, in Spanish-Fast-400, the best results were obtained by WOA-TwinSVM-Fs in terms of recall, precision, and F-measures. This can be interpreted given the specific characteristics of this dataset, such as its distribution of positive and negative instances or the nature of the data, which favor the strengths of WOA-TwinSVM-Fs. On the other hand, the precision measure shows more



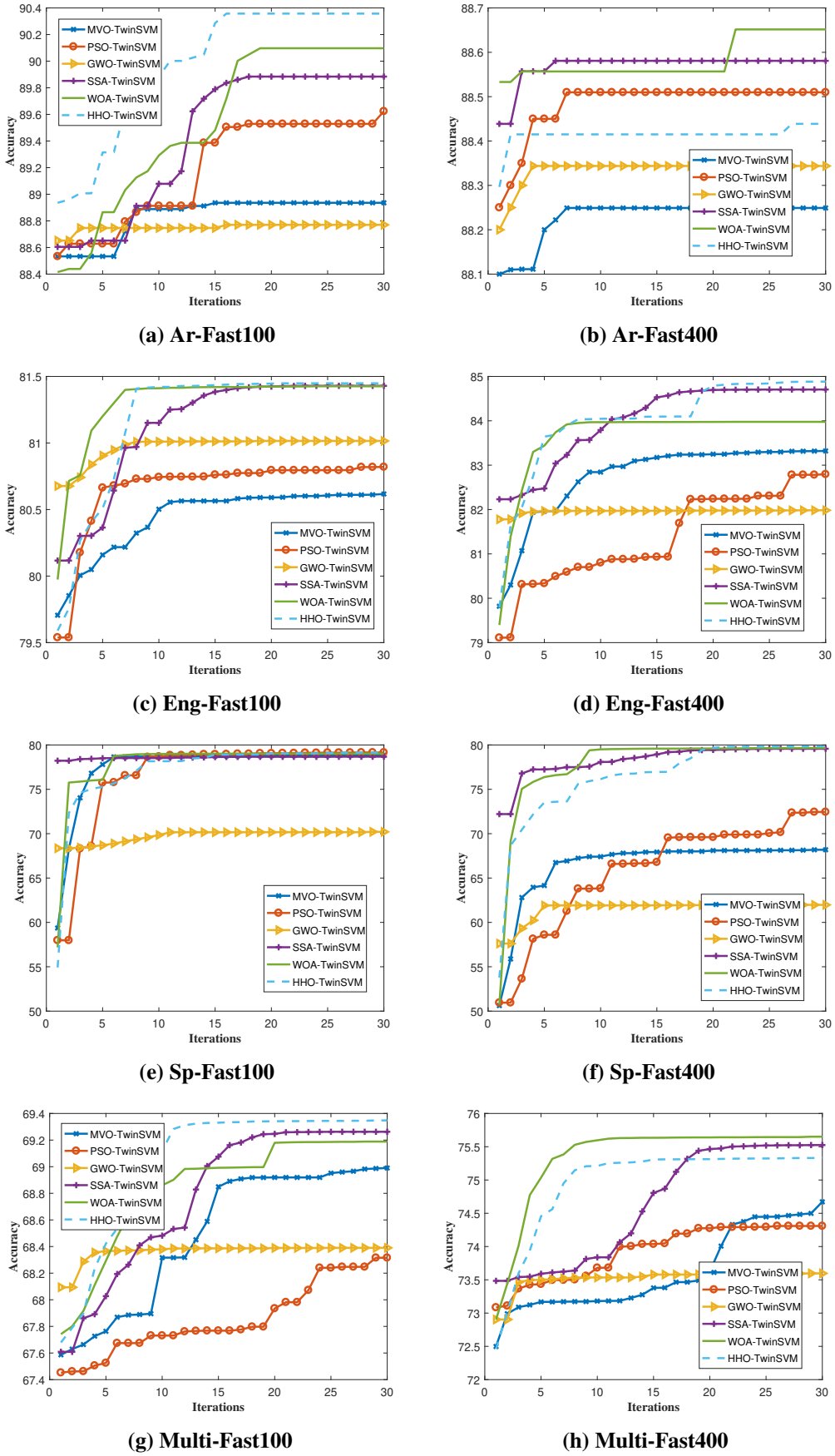
(a) Ar-BERT100 (b) Ar-BERT400

(c) Eng-BERT100 (d) Eng-BERT400

(e) Sp-BERT100 (f) Sp-BERT400

(g) Multi-BERT100 (h) Multi-BERT400

Fig. 7.3 Convergence curve charts for HHO-TwinSVM-Fs and other algorithms based on BERT 100/400 dimensions of each language



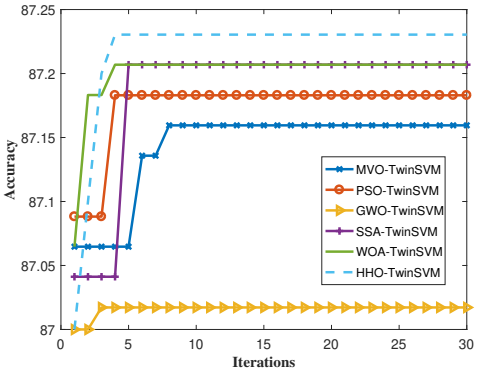
(a) Ar-Fast100 (b) Ar-Fast400

(c) Eng-Fast100 (d) Eng-Fast400

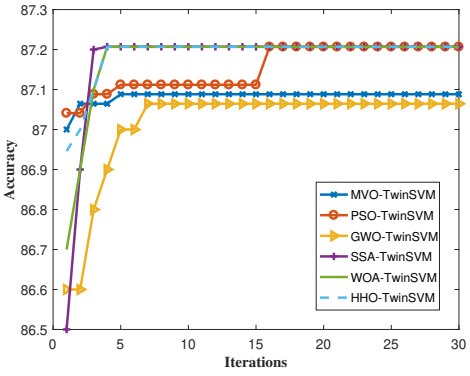
(e) Sp-Fast100 (f) Sp-Fast400

(g) Multi-Fast100 (h) Multi-Fast400

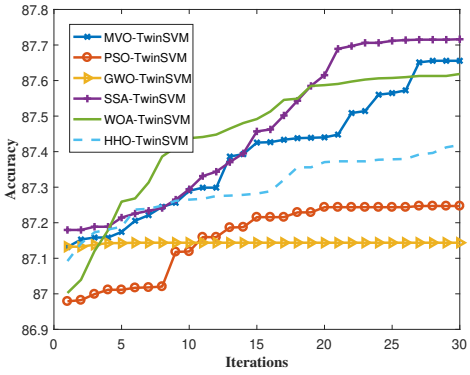
Fig. 7.4 Convergence curve charts for HHO-TwinSVM-Fs and other algorithms based on Fast 100/400 dimensions of each language



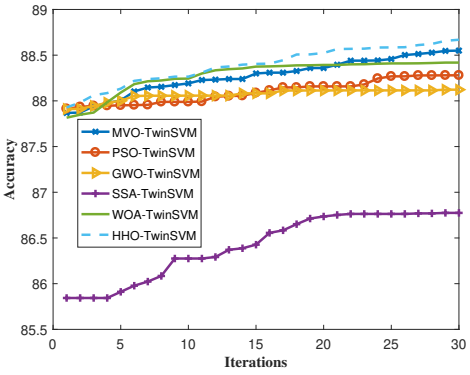
(a) Ar-MUSE100



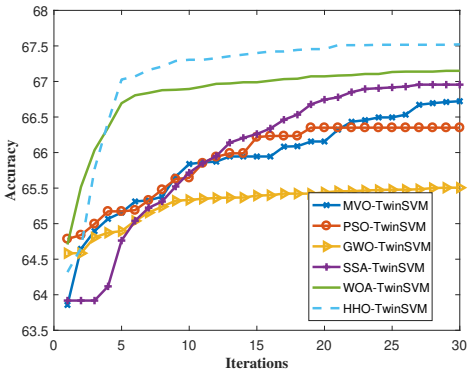
(b) Ar-MUSE400



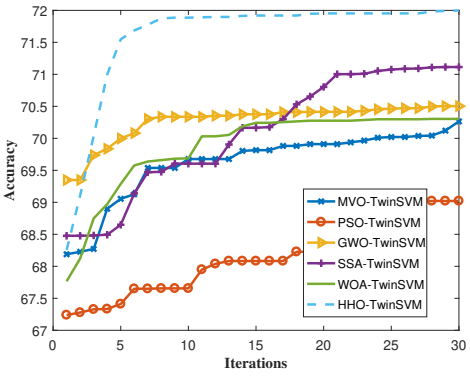
(c) Eng-MUSE100



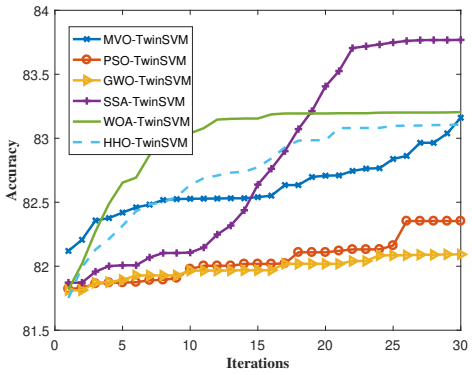
(d) Eng-MUSE400



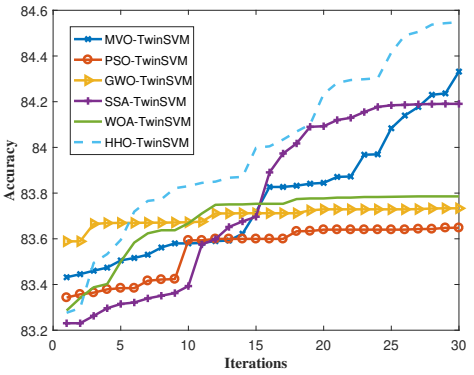
(e) Sp-MUSE100



(f) Sp-MUSE400



(g) Multi-MUSE100



(h) Multi-MUSE400

Fig. 7.5 Convergence curve charts for HHO-TwinSVM-Fs and other algorithms based on MUSE 100/400 dimensions of each language

distribution in results where in Arabic-BERT-100, English-MUSE-400, and Multi-MUSE-400 achieved by different algorithms, which are PSO-TwinSVM-Fs, HHO-TwinSVM-Fs, and SSA-TwinSVM-Fs. The distribution in precision results across these datasets highlights the adaptability of different algorithms to diverse linguistic and multilingual contexts. Also, each algorithm's unique optimization strategy or learning mechanism led to varying degrees of success in precision.

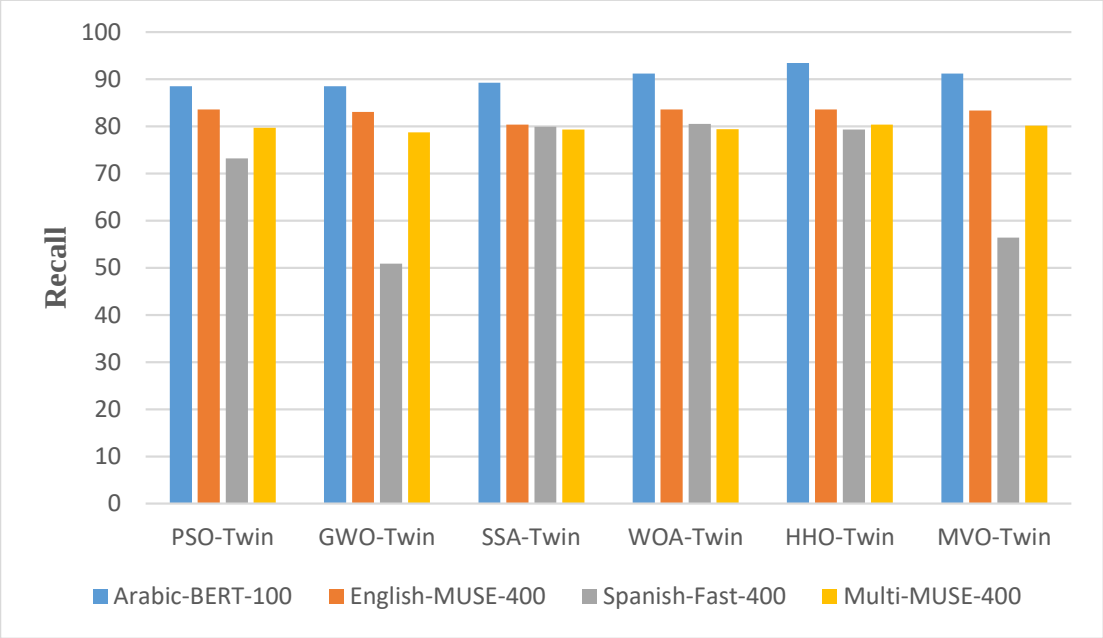


Fig. 7.6 Recall results of HHO-TwinSVM-Fs and other algorithms for the best datasets.

Also, in the next experiment, the same best datasets will be used as a representative of the other data in each language. As a result, we will conduct more examinations and analyses using only these datasets.

7.3.4 Experiment III: Comparison of well-known classification methods against the proposed approach

In this experiment, the proposed approach HHO-TwinSVM-Fs will be compared with well-known classification methods integrated with HHO in terms of accuracy, including XGboost, MLP, and k-NN. Here, our focus was on overall classification correctness rather than fine-tuning the algorithm for specific performance measures, which is the main goal of the model to enhance detection performance. According to the literature, these classifiers have high performance when paired with metaheuristic algorithms for feature selection

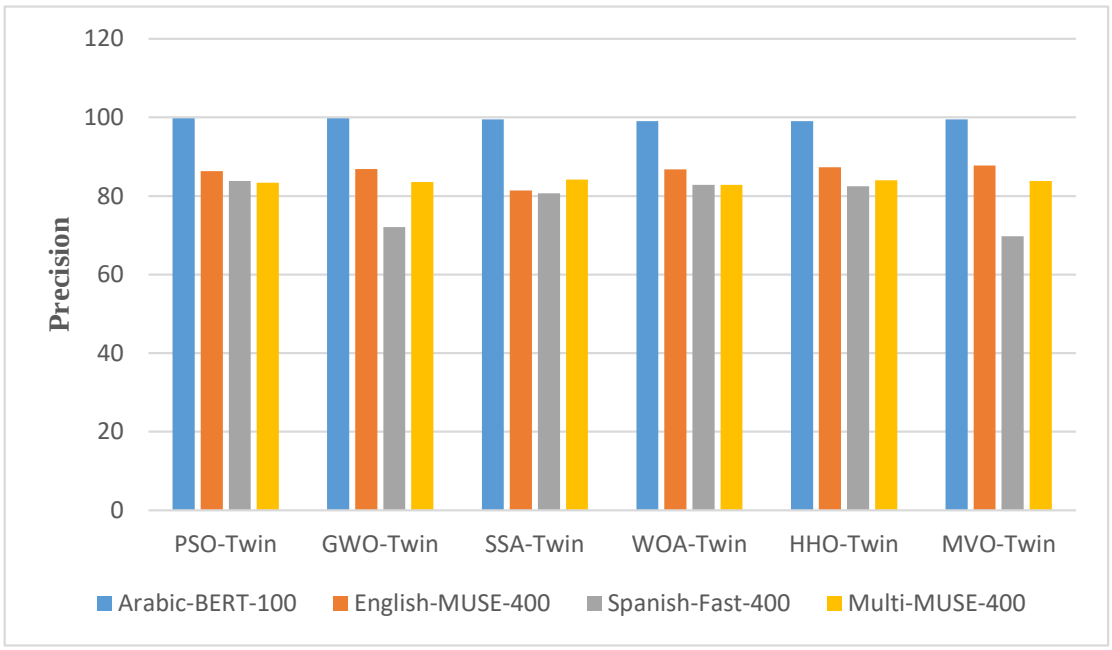


Fig. 7.7 Precision results of HHO-TwinSVM-Fs and other algorithms for the best datasets.

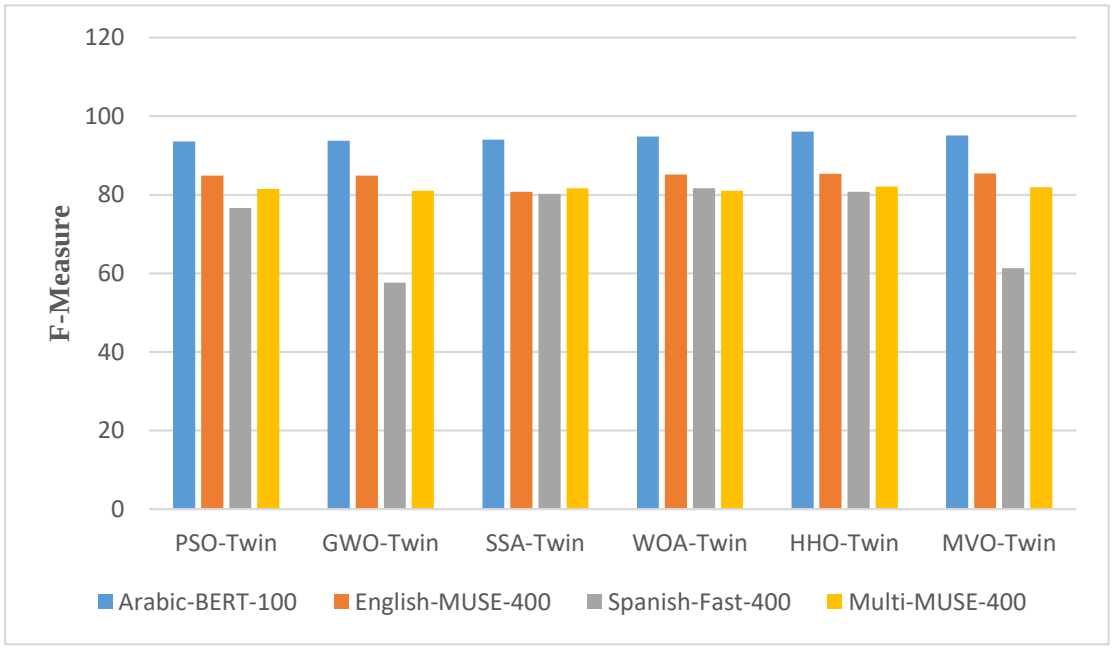


Fig. 7.8 F-measures results of HHO-TwinSVM-Fs and other algorithms for the best datasets.

and parameter optimization. In this study, HHO alongside feature selection will serve as parameter optimization for XGboost, weights & biases for MLP, and k value for k-NN. HHO is used as the main metaheuristic algorithm in this experiment due to its performance with TwinSVM previously. Additionally, the comparison will be focused on the datasets that obtained the highest accuracy in the prior experiment between all languages, namely, Arabic BERT-100, English MUSE-400, Spanish Fast-400, and Multi MUSE-400.

Table 7.10 presents the results of HHO-TwinSVM-Fs as well as the other approaches (HHO-XGboost-Fs, HHO-MLP-Fs, and HHO-kNN-Fs) for Arabic BERT-100, English MUSE-400, Spanish Fast-400, and Multi MUSE-400 datasets. Based on the results, HHO-TwinSVM-Fs outperformed the other algorithms in terms of accuracy on all datasets, followed by HHO-kNN-Fs, HHO-MLP-Fs, and HHO-XGboost-Fs, respectively. To be more specific, the differences between HHO-TwinSVM-Fs and the best second algorithm (HHO-kNN-Fs) were 2.7822%, 3.5047, 11.9422% and 8.7461% for Arabic BERT-100, English MUSE-400, Spanish Fast-400, and Multi MUSE-400 datasets, respectively.

Table 7.10 Results of HHO-TwinSVM-Fs and other well-known classifiers with HHO in terms of accuracy

Datasets	HHO-TwinSVM-Fs	HHO-XGboost-Fs	HHO-MLP-Fs	HHO-kNN-Fs
Arabic-BERT-100	**92.9741**	0.8681	0.8686	90.1919
English-MUSE-400	**89.0314**	0.6600	0.6681	85.5267
Spanish-Fast-400	**80.3580**	0.5015	0.5329	68.4158
Multi-MUSE-400	**85.0859**	0.6897	0.6317	76.3398

7.3.5 Experiment IV: Detection improvement using sentiment features

In this experiment, we generated new features based on sentiment analysis as well as the sentiment dataset labels as features and integrated them with original datasets in order to improve the detection of spam reviews. In this phase, we used the best datasets that we utilized in the previous experiments, including Arabic BERT-100, English MUSE-400, Spanish Fast-400, and Multi MUSE-400. As proven to be the best approach in the previous experiments, HHO-TwinSVM-Fs will only be used for this experiment. A list of some of the sentiment features that will be employed is provided in Table 7.11. It is also worth mentioning that sentiment labels (Positive, Neutral, and Negative) are also used as a feature in the new version of the datasets.

Based on the results of Table 7.12, we can notice the accuracy improvement in all datasets. For example, the Arabic BERT-100 sentiment version achieved better results than the normal version with 94.0735%, while the English MUSE-400 results were also in favor

Table 7.11 Sample of the sentiment features and their description.

#	Features	Description
1	mpqa-posCount	These features refer to a count of positive and negative words in a text, which are based on a specific list of words.
2	mpqa-negCount	
3	BingLiu-posCount	
4	BingLiu-negCount	
5	AFINN-posScore	Based on a specific list of words, these features indicate the positive or negative sentiment scores of a text.
6	AFINN-negScore	
7	S140-posScore	
8	S140-negScore	
9	NRC-Hash-Sent-posScore	
10	NRC-Hash-Sent-negScore	
11	NRC-10-anger	Based on a specific list of words, these features indicate whether particular emotions (such as anger, anticipation, disgust, etc.) are present in a text.
12	NRC-10-anticipation	
13	NRC-10-disgust	
14	NRC-10-fear	
15	NRC-10-joy	
16	NRC-10-sadness	
17	NRC-10-surprise	
18	NRC-10-trust	

of the sentiment version with 91.6988%. Spanish Fast-400 and Multi MUSE-400 had much better improvements for the sentiment version than the normal version with 89.8010% and 93.8307%, respectively. Using sentiment features to improve spam review detection showed significant differences of 1.0994%, 2.6674%, 9.4430%, and 8.7448% for Arabic BERT-100, English MUSE-400, Spanish Fast-400, and Multi MUSE-400, respectively.

Table 7.12 Comparison results of the normal and sentiment features versions of the datasets using HHO-TwinSVM-Fs.

Datasets	Normal Version		With Sentiment Features	
	Avg	Std Dev	Avg	Std Dev
Arabic-BERT-100	92.9741	4.1355	**94.0735**	**2.8301**
English-MUSE-400	89.0314	1.1229	**91.6988**	**1.3835**
Spanish-Fast-400	80.3580	2.4633	**89.8010**	**1.1403**
Multi-MUSE-400	85.0859	0.8379	**93.8307**	**2.3007**

7.3.6 Analysis of results, reviews, multilingual and sentiment features effects

The previous four subsections show the results of different experiments for spam reviews detection. Beginning with the standard SVM combined with different metaheuristic algorithms,

then TwinSVM and HHO compared with other combinations, comparison of different classification methods with HHO, and ending with adding new sentiment features to enhance the detection phase. The flow and order of experiments addressed this way in order to examine the performance of the proposed HHO-TwinSVM-Fs against other approaches, including, standard SVM, other metaheuristics, and various well-known classifiers. Eventually, the HHO-TwinSVM-Fs showed excellent performance in all experiments.

The pre-trained word embedding helped the detection phase due to different reasons, such as more relevant features being chosen, performing well on small datasets, and handling multilingual data. Thus, using pre-trained word embedding not only improved the performance of classification but also reduced the time and computational resources needed for such tasks. In this study, three pre-trained word embedding have been used, namely, BERT, FastText, and MUSE. Each of these has the ability to handle multiple languages simultaneously. However, it's been noticed that each method shows different performance based on the language, where Arabic was the best results using BERT, Spanish using FastText, and English and Multilingual using the MUSE method. Such differences in results added more reason for us to employ different pre-trained word embedding methods.

Furthermore, consumers should read reviews before purchasing a product or service to find more information about it. In addition, organizations can improve their products and then rethink their strategies or modify their designs. A change in the context of reviews also occurred during and after the pandemic.

Since review characteristics including style, context, and structure have changed, our review analysis must be updated. The pre-trained word embeddings were optimal for our study due to the use of context-relevant words as well as dealing with multilingual data. Also, using sentiment features to improve spam review detection. Sentiment features play a key role in increasing the accuracy of detection, especially in different languages. As a result, we can notice that spam reviews tend to exaggerate to describe products either positively or negatively. However, the reviews with positive sentiment were more spam than the negative ones.

Based on this, we highlighted the best features of our proposed approach, whether it be textual or sentimental. The HHO algorithm was used to rank features, alongside parameter optimization, to enhance detection performance. In other words, by identifying the most relevant features (words). Therefore, knowing the relevant features can help with both feature engineering and data preprocessing since it can reveal underlying relationships and patterns in the data. Further, the spam review detection model can be improved in dimensionality, generalization, and interpretability. However, relying on sentiment features for spam detection alone is not enough. Reviews that constitute spam are often characterized

as fake, biased, or misleading, and may be intended to promote or denigrate a particular product or service. That's why we kept the context and textual features and integrated them with features based on sentiment to achieve an excellent combination as shown in the previous experiment.

In this third approach, the experiments were divided into several phases: applying standard SVM, our proposed HHO-TwinSVM-FS approach, comparing with well-established classification methods, and finally incorporating sentiment features to improve detection. Overall, the proposed HHO-TwinSVM-FS approach achieved the best results in all experiments, with 92.9741%, 89.0314%, 80.3580%, and 85.0859% for Arabic, English, Spanish and Multilingual datasets, respectively. Then with the help of sentiment features the results have been improved by 1.0994%, 2.6674%, 9.4430%, and 8.7448% for Arabic BERT-100, English MUSE-400, Spanish Fast-400, and Multi MUSE-400, respectively.

The adoption of this third approach represents a substantial advancement, addressing several limitations encountered by the preceding methodologies discussed in chapters 5 and 6, as well as surpassing the benchmarks set by the current state-of-the-art techniques in the literature. This progression results in making a more robust model to detect spam reviews in multilingual environments. By utilizing the sentiment as a feature, the model shows enhancement in identifying the pattern of different aspects such as context, linguistics, and deceptive tactics. Thus, it moves beyond traditional feature sets and embraces the emotional dimension, allowing for a more comprehensive analysis of the data. Further, providing a model adept at learning and adapting to new patterns that ensures a robust and dynamic defense against emerging spamming techniques in multilingual environments.

Chapter 8

Conclusions and Future Work

8.1 Conclusions

With the expansion of internet in the last century, it has become one of the most innovative and valuable sources of information. Information that can bring knowledge and facts about different fields such as, medical, engineering, energy, agriculture, business, and so on. In business, information can be about market research, advertising, collaboration and communication, financing, and customer feedback. Such feedback on the internet can come in the form of reviews. Reviews are opinions expressed by consumers about products and services they have used. Many individuals can benefit from these reviews in different ways. For example, consumers read reviews related to the product or service they want to buy in order to make a wise purchasing decision. This way they will learn the quality of that product or service before purchasing it. Thus, save a lot of money and effort by only reading a few reviews. Another example of ways to benefit from reviews, is how organizations use the reviews as feedback about their product lines. As a result, they can modify their plans and strategies in order to enhance or redesign products before it's too late or making bad decisions.

In previous years, COVID-19 forced people to stay at home, which led to a huge increase in online reviews. The importance and popularity of reviews lead to the rapid increase of spam reviews. Spam reviews are fake, fraudulent, and false reviews aimed at misleading customers to make a profit or harming competitors. Many negative effects can be caused by using spam reviews such as loss of consumer trust, credibility of the business, and destruction of the online market. Therefore, it is important to identify these spam reviews. Text classification using machine learning algorithms is one most effective techniques to detect spam reviews in the literature.

Thus, in this thesis, three efficient approaches have been proposed to distinguish spam and the sentiment of reviews. Firstly, in order to detect spam reviews a combination of Harris Hawks Optimization (HHO) and Weighted Support Vector Machines (WSVM) were applied for feature weighting and optimizing parameters. Secondly, the TwinSVM with the Salp Swarm Algorithm (SSA) for parameter optimization has been used to conduct sentiment analysis. Finally, TwinSVM and HHO were employed for feature selection and hyperparameter tuning to detect spam reviews through sentiment analysis. To deal with the reviews written in other languages, three different languages have been used, which are English, Spanish, and Arabic. Further, to handle multiple language text, a variety of pre-trained word embeddings has been utilized, which include BERT. FastText, and MUSE as word representation. Based on the different languages and pre-trained embeddings, 24 datasets have been created with two dimensions (100/400). The results show that all proposed approaches outperformed the other algorithms. For example, in the first approach, the HHO-WSVM achieved the finest results in all datasets with 88.163%, 71.913%, 89.565%, and 84.270%, for English, Spanish, Arabic, and Multilingual datasets, respectively. SSA-TwinSVM, in the second approach, acquired the best results in three out of four datasets with 87.600%, 86.330%, and 87.100% for Arabic, English, and Multilingual datasets, respectively. As for the Spanish datasets, WOA-TwinSVM has the highest accuracy with 86.467%. Regarding the third approach, HHO-TwinSVM-Fs outperformed the remaining algorithms in Arabic, English, Spanish, and Multilingual datasets with 92.9741%, 89.0314%, 80.3580%, and 85.0859%, respectively. In this approach, sentiment-based features integrated with original datasets and enhanced the detection performance by 1.0994%, 2.6674%, 9.4430%, and 8.7448% for Arabic BERT-100, English MUSE-400, Spanish Fast-400, and Multi MUSE-400, respectively. An analysis of the review's style, context, and preferences was provided at the end of each approach. In the context of the concurrent execution of two tasks, specifically optimization coupled with either feature weighting or selection, our observations reveal that HHO surpasses SSA in performance. Nonetheless, it is noteworthy that when focusing on a singular task, SSA demonstrates superior results.

The performance improvement achieved by the third approach, HHO-TwinSVM-Fs, which integrates the sentiment-based features with the original datasets and applied feature selection, is particularly noteworthy. Also, the detailed analysis of review styles, contexts, and preferences highlights the changes in reviews over the previous years due to COVID-19. The analysis presented at the conclusion of each approach provides valuable insights into the effectiveness of the proposed methods.

In summary, this research not only presented novel and effective techniques for spam reviews detection through sentiment analysis but also, contributed to the understanding of the

impact of language diversity and pre-trained word embeddings on the overall performance of the proposed approaches. The results obtained affirm the significance and practical applicability of the developed methods in handling real-world scenarios involving multilingual and diverse review datasets.

Therefore, considering deploying the approach with its configuration scheme can present the prospect of a robust model for effectively detecting spam reviews. In terms of real-world applicability, the model shows promise in identifying spam reviews in scenarios where such detection is crucial, such as online platforms, e-commerce websites, review aggregation services, or social networks. Ultimately, deploy the model on a deployment framework, namely Flask or Django. The consistently high accuracy rates demonstrated across diverse datasets suggest that the model is robust and versatile. However, its performance in a real scenario would depend on factors such as the nature of the reviews, the prevalence of multilingual content, and the specific characteristics of the target platform.

Additionally, two papers originating from the thesis have been published in well-recognized journals, as outlined below:

- *Al-Zoubi, Ala'M., Antonio M. Mora, and Hossam Faris. "Spam Reviews Detection in the Time of COVID-19 Pandemic: Background, Definitions, Methods and Literature Analysis." Applied Sciences 12, no. 7 (2022): 3634.*

- *Ala'M, Al-Zoubi, Antonio M. Mora, and Hossam Faris. "A multilingual spam reviews detection based on pre-trained word embedding and weighted swarm support vector machines." IEEE Access (2023).*

In the first paper, a survey on the detection of spam reviews during the COVID-19 pandemic is presented, offering background information, definitions, methods, and a literature analysis. The paper was published in the Applied Sciences Journal and held a Q3 rank in JCR, with a 2.7 impact factor at the time of publication.

The second paper is derived from the first approach (Chapter 5) that used the weighted swarm support vector machines with pre-trained word embedding for spam reviews detection in a multilingual environment. The paper was published in the IEEE Access Journal and held a Q1 rank in JCR, with a 3.9 impact factor at the time of publication.

Also, I wish to bring to your attention that the third paper is presently undergoing an initial revisions phase of evaluation at the International Journal of Machine Learning and Cybernetics Journal.

8.2 Future Work

Despite the performance of SVM and TwinSVM with the swarm-based optimization algorithms in text classification tasks, particularly, spam review detection and sentiment analysis problems, there are still several areas that require further investigation. This decision was made based on considerations of scope, resources, and the primary focus of our study. Furthermore, the following points provide details on potential future work:

- Improve the performance of SVMs and TwinSVMs using different types of metaheuristic algorithms to optimize other factors, including kernel function and regularization parameter. The selection of an appropriate kernel function plays a pivotal role in the performance of SVMs and TwinSVMs, which can enhance their ability to capture complex relationships within the data. Similarly, the regularization parameter has also the potential to improve the SVMs and TwinSVMs by controlling the trade-off between achieving a low training error and preventing overfitting. Leveraging metaheuristic algorithms to fine-tune this parameter could efficiently provide an optimal model generalization across different datasets and problems.

- The detection of spam reviews through sentiment analysis in a multilingual environment is still at an early stage. There is a need for further studies to investigate the relationship between detection and sentiment in different languages. The existing body of research in this area is limited, and additional studies are important to describe the complex relationship between spam review detection and sentiment across various languages. Given the diverse linguistic landscape, understanding how the detection of spam reviews through sentiment analysis techniques performed in different languages is crucial. This includes examining the impact of language-specific features, sentiment lexicons, and idiomatic expressions on identifying spam reviews. Moreover, the scarcity of comprehensive datasets in multiple languages poses a challenge in training and evaluating models for spam review detection. Future research endeavors should address this limitation by generating more extensive and diverse datasets across languages. Thus, provides more exploration of the complexities associated with multilingual sentiment-based spam detection.

- Perform more analysis to improve spam review detection using different text classification tasks such as genre classification, named entity recognition (NER), and Part-of-speech (POS) tagging. Beyond traditional approaches, delving into these classifications can offer a multi-faceted perspective on the detection process. For example, investigating spam review detection within the context of genre classification

enhances the nuanced understanding of how the content's genre influences detection performance. As for using NER spam reviews detection, it expands the identification and classifying of other entities such as product names, locations, and persons within reviews, which eventually, produces more context-aware and sophisticated detection systems. Furthermore, analyzing spam review detection through POS tagging offers insights into the grammatical structure of reviews. Accordingly, identifying patterns in the distribution of POS can uncover more linguistic cues indicative of spam reviews.

- Creating more specialized pre-trained word embedding models that relate directly to spam reviews detection and sentiment analysis. In this way, the embeddings will then be more specific to the domain. By crafting embeddings that are directly associated with spam reviews through sentiment analysis, the resulting models can exhibit heightened domain specificity, offering a better understanding of the content. Hence, by training embeddings specifically for spam review detection based on sentiment analysis, the model can better capture the semantics unique to these tasks. Further, providing the model with a richer set of features for decision-making that leads to more accurate and context-aware predictions.

- Consideration of additional languages to enhance sentiment analysis and spam review for large and diverse multilingual datasets. The inclusion of more languages acknowledges the rich cultural and linguistic diversity present in large datasets. Considering additional languages ensures that spam review detection based on sentiment analysis models is globally applicable and not confined to specific linguistic contexts. Expanding the scope to more languages in spam review detection based on sentiment analysis contributes to a more inclusive and user-centric experience.

- Apply a Large Language Model (LLM) capable of analyzing and understanding the context of user-generated content related to spam review detection and sentiment analysis. Eventually, it should comprehend the semantic relationships between words in diverse languages within a given context.

References

[1] Abu Hammad, A. S. (2013). An approach for detecting spam in arabic opinion reviews.

[2] Abualigah, L., Shehab, M., Alshinwan, M., and Alabool, H. (2020). Salp swarm algorithm: a comprehensive survey. *Neural Computing and Applications*, 32(15):11195–11215.

[3] Abusnaina, A. A., Ahmad, S., Jarrar, R., and Mafarja, M. (2018). Training neural networks using salp swarm algorithm for pattern classification. In *Proceedings of the 2nd international conference on future networks and distributed systems*, pages 1–6.

[4] Adeniyi, D. A., Wei, Z., and Yongquan, Y. (2016). Automated web usage data mining and recommendation system using k-nearest neighbor (knn) classification method. *Applied Computing and Informatics*, 12(1):90–108.

[5] Agüero-Torales, M. M., Salas, J. I. A., and López-Herrera, A. G. (2021). Deep learning and multilingual sentiment analysis on social media data: An overview. *Applied Soft Computing*, 107:107373.

[6] Aguilar, S. J. and Baek, C. (2020). Sexual harassment in academe is underreported, especially by students in the life and physical sciences. *PloS one*, 15(3):e0230312.

[7] Agustiningsih, K. K., Utami, E., and Alsyaibani, M. A. (2022). Sentiment analysis of covid-19 vaccines in indonesia on twitter using pre-trained and self-training word embeddings. *Jurnal Ilmu Komputer dan Informasi*, 15(1):39–46.

[8] Al-Ahmad, B., Al-Zoubi, A., Abu Khurma, R., and Aljarah, I. (2021). An evolutionary fake news detection method for covid-19 pandemic information. *Symmetry*, 13(6):1091.

[9] Alamoodi, A. H., Zaidan, B. B., Zaidan, A. A., Albahri, O. S., Mohammed, K., Malik, R. Q., Almahdi, E. M., Chyad, M. A., Tareq, Z., Albahri, A. S., et al. (2021). Sentiment analysis and its applications in fighting covid-19 and infectious diseases: A systematic review. *Expert systems with applications*, 167:114155.

[10] Alba, E., Garcia-Nieto, J., Jourdan, L., and Talbi, E.-G. (2007). Gene selection in cancer classification using pso/svm and ga/svm hybrid algorithms. In *2007 IEEE congress on evolutionary computation*, pages 284–290. IEEE.

[11] Alharbi, N. M., Alghamdi, N. S., Alkhammash, E. H., and Al Amri, J. F. (2021). Evaluation of sentiment analysis via word embedding and rnn variants for amazon online reviews. *Mathematical Problems in Engineering*, 2021.

[12] Aljarah, I., Al-Zoubi, A., Faris, H., Hassonah, M. A., Mirjalili, S., and Saadeh, H. (2018a). Simultaneous feature selection and support vector machine optimization using the grasshopper optimization algorithm. *Cognitive Computation*, 10(3):478–495.

[13] Aljarah, I., Mafarja, M., Heidari, A. A., Faris, H., Zhang, Y., and Mirjalili, S. (2018b). Asynchronous accelerating multi-leader salp chains for feature selection. *Applied Soft Computing*, 71:964–979.

[14] Allard, T., Dunn, L. H., and White, K. (2020). Negative reviews, positive impact: Consumer empathetic responding to unfair word of mouth. *Journal of Marketing*, 84(4):86–108.

[15] Alqatawna, J., Faris, H., Jaradat, K., Al-Zewairi, M., Adwan, O., et al. (2015). Improving knowledge based spam detection methods: The effect of malicious related features in imbalance data distribution. *International Journal of Communications, Network and System Sciences*, 8(05):118.

[16] Alzubi, O. A., Alzubi, J. A., Al-Zoubi, A., Hassonah, M. A., and Kose, U. (2021). An efficient malware detection approach with feature weighting based on harris hawks optimization. *Cluster Computing*, pages 1–19.

[17] Anas, S. M. and Kumari, S. (2021). Opinion mining based fake product review monitoring and removal system. In *2021 6th International Conference on Inventive Computation Technologies (ICICT)*, pages 985–988. IEEE.

[18] Anass, F., Jamal, R., Mahraz, M. A., Ali, Y., and Tairi, H. (2020). Deceptive opinion spam based on deep learning. In *2020 Fourth International Conference On Intelligent Computing in Data Sciences (ICDS)*, pages 1–5. IEEE.

[19] Ansari, S. and Gupta, S. (2021). Customer perception of the deceptiveness of online product reviews: A speech act theory perspective. *International Journal of Information Management*, 57:102286.

[20] Arcila-Calderón, C., Blanco-Herrero, D., Frías-Vázquez, M., and Seoane-Pérez, F. (2021). Refugees welcome? online hate speech and sentiments in twitter in spain during the reception of the boat aquarius. *Sustainability*, 13(5):2728.

[21] Asani, E., Vahdat-Nejad, H., and Sadri, J. (2021). Restaurant recommender system based on sentiment analysis. *Machine Learning with Applications*, 6:100114.

[22] Asasi, M. S., Ahanch, M., and Holari, Y. T. (2018). Optimal allocation of distributed generations and shunt capacitors using salp swarm algorithm. In *Electrical Engineering (ICEE), Iranian Conference on*, pages 1166–1172. IEEE.

[23] Ashwini, M. and Padma, M. (2020). Efficiently analyzing and detecting fake reviews through opinion mining. *Int J Comput Sci Mob Comput*, 9(7).

[24] Aye, C. M. and Oo, K. M. (2014). Review spammer detection by using behaviors based scoring methods. In *Proceedings of international conference on advances in engineering and technology*.

[25] Aye, Y. M. and Aung, S. S. (2018). Senti-lexicon and analysis for restaurant reviews of myanmar text. *International Journal of Advanced Engineering, Management and Science*, 4(5):240004.

[26] Bairathi, D. and Gopalani, D. (2019). Salp swarm algorithm (ssa) for training feed-forward neural networks. In *Soft computing for problem solving*, pages 521–534. Springer.

[27] Bajaj, S., Garg, N., and Singh, S. K. (2017). A novel user-based spam review detection. *Procedia computer science*, 122:1009–1015.

[28] Bamakan, S. M. H., Nurgaliev, I., and Qu, Q. (2019). Opinion leader detection: A methodological review. *Expert Systems with Applications*, 115:200–222.

[29] Banerjee, S. and Chua, A. Y. (2017). Theorizing the textual differences between authentic and fictitious reviews: Validation across positive, negative and moderate polarities. *Internet Research*.

[30] Banerjee, S. and Chua, A. Y. (2021). Calling out fake online reviews through robust epistemic belief. *Information & Management*, 58(3):103445.

[31] Barbado, R., Araque, O., and Iglesias, C. A. (2019). A framework for fake review detection in online consumer electronics retailers. *Information Processing & Management*, 56(4):1234–1244.

[32] Barushka, A. and Hajek, P. (2019). Review spam detection using word embeddings and deep neural networks. In *IFIP International Conference on Artificial Intelligence Applications and Innovations*, pages 340–350. Springer.

[33] Bates, D. (2013). Samsung ordered to pay $340,000 after it paid people to write negative online reviews about htc phones.

[34] Beigi, G., Hu, X., Maciejewski, R., and Liu, H. (2016). An overview of sentiment analysis in social media and its applications in disaster relief. *Sentiment analysis and ontology engineering*, pages 313–340.

[35] Bhuvaneshwari, P., Rao, A. N., and Robinson, Y. H. (2021). Spam review detection using self attention based cnn and bi-directional lstm. *Multimedia Tools and Applications*, pages 1–18.

[36] Bidgolya, A. J. and Rahmaniana, Z. (2020). A robust opinion spam detection method against malicious attackers in social media. *arXiv preprint arXiv:2008.08650*.

[37] Blal, I. and Sturman, M. C. (2014). The differential effects of the quality and quantity of online reviews on hotel room sales. *Cornell Hospitality Quarterly*, 55(4):365–375.

[38] Blitzer, J., Dredze, M., and Pereira, F. (2007). Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In *Proceedings of the 45th annual meeting of the association of computational linguistics*, pages 440–447.

[39] Branch, M. (2012). A multi layer perceptron neural network trained by invasive weed optimization for potato color image segmentation. *Trends Appl. Sci. Res*, 7(6):445–455.

[40] Brown, J. J. and Reingen, P. H. (1987). Social ties and word-of-mouth referral behavior. *Journal of Consumer research*, 14(3):350–362.

[41] Buckley, J. (2019). Tripadvisor defends itself against claim that up to one in seven reviews might be fake.

[42] Budhi, G. S., Chiong, R., Pranata, I., and Hu, Z. (2021a). Using machine learning to predict the sentiment of online reviews: a new framework for comparative analysis. *Archives of Computational Methods in Engineering*, 28(4):2543–2566.

[43] Budhi, G. S., Chiong, R., and Wang, Z. (2021b). Resampling imbalanced data to detect fake reviews using machine learning classifiers and textual-based features. *Multimedia Tools and Applications*, pages 1–19.

[44] Byun, H., Jeong, S., and Kim, C.-k. (2021). Sc-com: Spotting collusive community in opinion spam detection. *Information Processing & Management*, 58(4):102593.

[45] Cambria, E. and White, B. (2014). Jumping nlp curves: A review of natural language processing research. *IEEE Computational intelligence magazine*, 9(2):48–57.

[46] Chen, R. Y., Guo, J. Y., and Deng, X. L. (2014a). Detecting fake reviews of hype about restaurants by sentiment analysis. In *International Conference on Web-Age Information Management*, pages 22–30. Springer.

[47] Chen, T., He, T., Benesty, M., Khotilovich, V., Tang, Y., Cho, H., Chen, K., et al. (2015). Xgboost: extreme gradient boosting. *R package version 0.4-2*, 1(4):1–4.

[48] Chen, W.-J., Shao, Y.-H., Deng, N.-Y., and Feng, Z.-L. (2014b). Laplacian least squares twin support vector machine for semi-supervised classification. *Neurocomputing*, 145:465–476.

[49] Chen, W.-J., Shao, Y.-H., Li, C.-N., and Deng, N.-Y. (2016). Mltsvm: A novel twin support vector machine to multi-label learning. *Pattern Recognition*, 52:61–74.

[50] Chen, W.-J., Shao, Y.-H., Li, C.-N., Liu, M.-Z., Wang, Z., and Deng, N.-Y. (2020). ν-projection twin support vector machine for pattern classification. *Neurocomputing*, 376:10–24.

[51] Cheng, Y.-H. and Ho, H.-Y. (2015). Social influence's impact on reader perceptions of online reviews. *Journal of Business Research*, 68(4):883–887.

[52] Chevalier, J. A. and Mayzlin, D. (2006). The effect of word of mouth on sales: Online book reviews. *Journal of marketing research*, 43(3):345–354.

[53] Choo, E., Yu, T., and Chi, M. (2015). Detecting opinion spammer groups through community discovery and sentiment analysis. In *IFIP annual conference on data and applications security and privacy*, pages 170–187. Springer.

[54] CIRS (2018). China's first e-commerce law published.

[55] CMA (2015). Online reviews and endorsements report on the cma's call for information.

[56] Comito, C., Forestiero, A., and Pizzuti, C. (2019). Word embedding based clustering to detect topics in social media. In *2019 IEEE/WIC/ACM International Conference on Web Intelligence (WI)*, pages 192–199. IEEE.

[57] Conneau, A., Lample, G., Ranzato, M., Denoyer, L., and Jégou, H. (2017). Word translation without parallel data. *arXiv preprint arXiv:1710.04087*.

[58] Corizzo, R., Zdravevski, E., Russell, M., Vagliano, A., and Japkowicz, N. (2020). Feature extraction based on word embedding models for intrusion detection in network traffic. *Journal of Surveillance, Security and Safety*, 1(2):140–150.

[59] Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine learning*, 20(3):273–297.

[60] Cowan, D. A. (1986). Developing a process model of problem recognition. *Academy of Management Review*, 11(4):763–776.

[61] de Gregorio, F., Fox, A. K., and Yoon, H. J. (2021). Pseudo-reviews: Conceptualization and consumer effects of a new online phenomenon. *Computers in Human Behavior*, 114:106545.

[62] De Pelsmacker, P., Van Tilburg, S., and Holthof, C. (2018). Digital marketing strategies, online reviews and hotel performance. *International Journal of Hospitality Management*, 72:47–55.

[63] Deng, L., Wei, J., Liang, S., Wen, Y., and Liao, X. (2020). Review spam detection based on multi-dimensional features. In *International Conference on AI and Mobile Services*, pages 106–123. Springer.

[64] Deniz, A., Angin, M., and Angin, P. (2021). Evolutionary multiobjective feature selection for sentiment analysis. *IEEE Access*, 9:142982–142996.

[65] Dessì, D., Recupero, D. R., and Sack, H. (2021). An assessment of deep learning models and word embeddings for toxicity detection within online textual comments. *Electronics*, 10(7):779.

[66] Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

[67] Dhiman, H. S., Deb, D., Muyeen, S., and Kamwa, I. (2021). Wind turbine gearbox anomaly detection based on adaptive threshold and twin support vector machines. *IEEE Transactions on Energy Conversion*, 36(4):3462–3469.

[68] Dickerson, J. P., Kagan, V., and Subrahmanian, V. (2014). Using sentiment to detect bots on twitter: Are humans more opinionated than bots? In *2014 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2014)*, pages 620–627. IEEE.

[69] Ding, S., An, Y., Zhang, X., Wu, F., and Xue, Y. (2017). Wavelet twin support vector machines based on glowworm swarm optimization. *Neurocomputing*, 225:157–163.

[70] Ding, S., Zhang, X., and Yu, J. (2016). Twin support vector machines based on fruit fly optimization algorithm. *International Journal of Machine Learning and Cybernetics*, 7(2):193–203.

[71] Divya, P. and Devi, B. A. (2022). Hybrid metaheuristic algorithm enhanced support vector machine for epileptic seizure detection. *Biomedical Signal Processing and Control*, 78:103841.

[72] Dong, M., Yao, L., Wang, X., Benatallah, B., Huang, C., and Ning, X. (2020). Opinion fraud detection via neural autoencoder decision forest. *Pattern Recognition Letters*, 132:21–29.

[73] Dorigo, M., Birattari, M., and Stutzle, T. (2006). Ant colony optimization. *IEEE computational intelligence magazine*, 1(4):28–39.

[74] Duan, W., Gu, B., and Whinston, A. B. (2008). Do online reviews matter?—an empirical investigation of panel data. *Decision support systems*, 45(4):1007–1016.

[75] Dwoskin, E. and Timberg, C. (2018). How merchants use facebook to flood amazon with fake reviews.

[76] El-Fergany, A. A. and Hasanien, H. M. (2020). Salp swarm optimizer to solve optimal power flow comprising voltage stability analysis. *Neural Computing and Applications*, 32(9):5267–5283.

[77] Elgeldawi, E., Sayed, A., Galal, A. R., and Zaki, A. M. (2021). Hyperparameter tuning for machine learning algorithms used for arabic sentiment analysis. In *Informatics*, volume 8, page 79. MDPI.

[78] Elmurngi, E. and Gherbi, A. (2017). Detecting fake reviews through sentiment analysis using machine learning techniques. *IARIA/data analytics*, pages 65–72.

[79] Elmurngi, E. and Gherbi, A. (2018). Fake reviews detection on movie reviews through sentiment analysis using supervised learning techniques. *International Journal on Advances in Systems and Measurements*, 11(1 & 2):196–207.

[80] Esfe, M. H., Ahangar, M. R. H., Rejvani, M., Toghraie, D., and Hajmohammad, M. H. (2016). Designing an artificial neural network to predict dynamic viscosity of aqueous nanofluid of tio2 using experimental data. *International communications in heat and mass transfer*, 75:192–196.

[81] Evans, A. M., Stavrova, O., and Rosenbusch, H. (2021). Expressions of doubt and trust in online user reviews. *Computers in Human Behavior*, 114:106556.

[82] Ezpeleta, E., Zurutuza, U., and Gómez Hidalgo, J. M. (2016). Does sentiment analysis help in bayesian spam filtering? In *International Conference on Hybrid Artificial Intelligence Systems*, pages 79–90. Springer.

[83] Fahfouh, A., Riffi, J., Mahraz, M. A., Yahyaouy, A., and Tairi, H. (2020). Pv-dae: A hybrid model for deceptive opinion spam based on neural network architectures. *Expert Systems with Applications*, 157:113517.

[84] Fang, X. and Zhan, J. (2015). Sentiment analysis using product review data. *Journal of Big Data*, 2(1):1–14.

[85] Faris, H., Hassonah, M. A., Al-Zoubi, A., Mirjalili, S., and Aljarah, I. (2018). A multi-verse optimizer approach for feature selection and optimizing svm parameters based on a robust system architecture. *Neural Computing and Applications*, 30(8):2355–2369.

[86] Fayaz, M., Khan, A., Rahman, J. U., Alharbi, A., Uddin, M. I., and Alouffi, B. (2020). Ensemble machine learning model for classification of spam product reviews. *Complexity*, 2020.

[87] Fayazbakhsh, S. and Sinha, J. (2012). Review spam detection: a network-based approach. *Final project report: CSE*, 590.

[88] Fazzolari, M., Buccafurri, F., Lax, G., and Petrocchi, M. (2021). Experience: Improving opinion spam detection by cumulative relative frequency distribution. *Journal of Data and Information Quality (JDIQ)*, 13(1):1–16.

[89] Feng, Z.-k., Niu, W.-j., Wan, X.-y., Xu, B., Zhu, F.-l., and Chen, J. (2022). Hydrological time series forecasting via signal decomposition and twin support vector machine using cooperation search algorithm for parameter identification. *Journal of Hydrology*, 612:128213.

[90] Fernández, R. (2023, Accessed 2023). Los idiomas más hablados en el mundo en 2023.

[91] Forestiero, A. (2017). Bio-inspired algorithm for outliers detection. *Multimedia Tools and Applications*, 76(24):25659–25677.

[92] Forestiero, A. (2021). Metaheuristic algorithm for anomaly detection in internet of things leveraging on a neural-driven multiagent system. *Knowledge-Based Systems*, 228:107241.

[93] Fornaciari, T. and Poesio, M. (2014). Identifying fake amazon reviews as learning from crowds.

[94] Forsberg, A. J. L. and Tullberg, B. S. (1995). The relationship between cumulative number of cohabiting partners and number of children for men and women in modern sweden. *Ethology and Sociobiology*, 16(3):221–232.

[95] Gani, A. (2015). Amazon sues 1,000 'fake reviewers'.

[96] Gao, S., Hao, J., and Fu, Y. (2015). The application and comparison of web services for sentiment analysis in tourism. In *2015 12th International Conference on Service Systems and Service Management (ICSSSM)*, pages 1–6. IEEE.

[97] Gao, Y., Gong, M., Xie, Y., and Qin, A. (2021). An attention-based unsupervised adversarial model for movie review spam detection. *arXiv preprint arXiv:2104.00955*.

[98] Gao, Y., Xie, L., Zhang, Z., and Fan, Q. (2020). Twin support vector machine based on improved artificial fish swarm algorithm with application to flame recognition. *Applied Intelligence*, 50(8):2312–2327.

[99] Gardner, M. W. and Dorling, S. (1998). Artificial neural networks (the multilayer perceptron)—a review of applications in the atmospheric sciences. *Atmospheric environment*, 32(14-15):2627–2636.

[100] Gautam, G. and Yadav, D. (2014). Sentiment analysis of twitter data using machine learning approaches and semantic analysis. In *2014 Seventh International Conference on Contemporary Computing (IC3)*, pages 437–442. IEEE.

[101] Ghai, R., Kumar, S., and Pandey, A. C. (2019). Spam detection using rating and review processing method. In *Smart Innovations in Communication and Computational Sciences*, pages 189–198. Springer.

[102] Glover, F. and Laguna, M. (1998). Tabu search. In *Handbook of combinatorial optimization*, pages 2093–2229. Springer.

[103] Graff, M., Miranda-Jimenez, S., Tellez, E. S., and Moctezuma, D. (2020). Evomsa: A multilingual evolutionary approach for sentiment analysis [application notes]. *IEEE Computational Intelligence Magazine*, 15(1):76–88.

[104] Habib, M., Faris, H., Hassonah, M. A., Alqatawna, J., Sheta, A. F., and Ala'M, A.-Z. (2018). Automatic email spam detection using genetic programming with smote. In *2018 Fifth HCT Information Technology Trends (ITT)*, pages 185–190. IEEE.

[105] Hajek, P., Barushka, A., and Munk, M. (2020). Fake consumer review detection using deep neural networks integrating word embeddings and emotion mining. *Neural Computing and Applications*, 32(23):17259–17274.

[106] Hakak, S., Alazab, M., Khan, S., Gadekallu, T. R., Maddikunta, P. K. R., and Khan, W. Z. (2021). An ensemble machine learning approach through effective feature extraction to classify fake news. *Future Generation Computer Systems*, 117:47–58.

[107] Harfoushi, O., Hasan, D., and Obiedat, R. (2018). Sentiment analysis algorithms through azure machine learning: Analysis and comparison. *Modern Applied Science*, 12(7):49.

[108] Hassan, R. and Islam, M. R. (2021). Impact of sentiment analysis in fake online review detection. In *2021 International Conference on Information and Communication Technology for Sustainable Development (ICICT4SD)*, pages 21–24. IEEE.

[109] Heidari, A. A., Mirjalili, S., Faris, H., Aljarah, I., Mafarja, M., and Chen, H. (2019). Harris hawks optimization: Algorithm and applications. *Future generation computer systems*, 97:849–872.

[110] Heydari, A., ali Tavakoli, M., Salim, N., and Heydari, Z. (2015). Detection of review spam: A survey. *Expert Systems with Applications*, 42(7):3634–3642.

[111] Hossain, E., Sharif, O., Hoque, M. M., and Sarker, I. H. (2020). Sentilstm: A deep learning approach for sentiment analysis of restaurant reviews. *arXiv preprint arXiv:2011.09684*.

[112] Houssein, E. H. (2019a). Machine learning and meta-heuristic algorithms for renewable energy: a systematic review. *Advanced Control and Optimization Paradigms for Wind Energy Systems*, pages 165–187.

[113] Houssein, E. H. (2019b). Particle swarm optimization-enhanced twin support vector regression for wind speed forecasting. *Journal of Intelligent Systems*, 28(5):905–914.

[114] Houssein, E. H., Ewees, A. A., and ElAziz, M. A. (2018). Improving twin support vector machine based on hybrid swarm optimizer for heartbeat classification. *Pattern Recognition and Image Analysis*, 28(2):243–253.

[115] Houssein, E. H., Mahdy, M. A., Shebl, D., and Mohamed, W. M. (2021). A survey of metaheuristic algorithms for solving optimization problems. In *Metaheuristics in Machine Learning: Theory and Applications*, pages 515–543. Springer.

[116] Hu, N., Bose, I., Koh, N. S., and Liu, L. (2012). Manipulation of online reviews: An analysis of ratings, readability, and sentiments. *Decision support systems*, 52(3):674–684.

[117] Hu, N., Liu, L., and Zhang, J. J. (2008). Do online reviews affect product sales? the role of reviewer characteristics and temporal effects. *Information Technology and management*, 9(3):201–214.

[118] Hu, S., Kumar, A., Al-Turjman, F., Gupta, S., Seth, S., et al. (2020). Reviewer credibility and sentiment analysis based user profile modelling for online product recommendation. *IEEE Access*, 8:26172–26189.

[119] Huang, J., Qian, T., He, G., Zhong, M., and Peng, Q. (2013). Detecting professional spam reviewers. In *International Conference on Advanced Data Mining and Applications*, pages 288–299. Springer.

[120] Hussain, N., Mirza, H. T., Hussain, I., Iqbal, F., and Memon, I. (2020a). Spam review detection using the linguistic and spammer behavioral methods. *IEEE Access*, 8:53801–53816.

[121] Hussain, N., Mirza, H. T., Iqbal, F., Hussain, I., and Kaleem, M. (2020b). Detecting spam product reviews in roman urdu script. *The Computer Journal*.

[122] Irissappane, A. A., Yu, H., Shen, Y., Agrawal, A., and Stanton, G. (2020). Leveraging gpt-2 for classifying spam reviews with limited labeled data via adversarial training. *arXiv preprint arXiv:2012.13400*.

[123] James, G., Witten, D., Hastie, T., and Tibshirani, R. (2013). *An introduction to statistical learning*, volume 6. Springer.

[124] Javed, M. S., Majeed, H., Mujtaba, H., and Beg, M. O. (2021). Fake reviews classification using deep learning ensemble of shallow convolutions. *Journal of Computational Social Science*, pages 1–20.

[125] Jayathunga, D. P., Ranasinghe, R. I. S., and Murugiah, R. (2021). A comparative study of supervised machine learning techniques for deceptive review identification using linguistic inquiry and word count. In *International Conference on Computational Intelligence in Information System*, pages 97–105. Springer.

[126] Jensi, R. and Jiji, G. W. (2016). An enhanced particle swarm optimization with levy flight for global optimization. *Applied Soft Computing*, 43:248–261.

[127] Jha, A. and Mamidi, R. (2017). When does a compliment become sexist? analysis and classification of ambivalent sexism using twitter data. In *Proceedings of the second workshop on NLP and computational social science*, pages 7–16.

[128] Jiang, B., hao Cao, R., and Chen, B. (2013). Detecting product review spammers using activity model. In *2013 international conference on advanced computer science and electronics information (ICACSEI 2013)*. Atlantis Press.

[129] Jimenez-Castano, C., Alvarez-Meza, A., and Orozco-Gutierrez, A. (2020). Enhanced automatic twin support vector machine for imbalanced data classification. *Pattern Recognition*, 107:107442.

[130] Jindal, N. and Liu, B. (2007). Review spam detection. In *Proceedings of the 16th international conference on World Wide Web*, pages 1189–1190.

[131] Kaibi, I., Nfaoui, E. H., and Satori, H. (2020). Sentiment analysis approach based on combination of word embedding techniques. In *Embedded Systems and Artificial Intelligence*, pages 805–813. Springer.

[132] Kamyab, M., Liu, G., and Adjeisah, M. (2021). Attention-based cnn and bi-lstm model based on tf-idf and glove word embedding for sentiment analysis. *Applied Sciences*, 11(23):11255.

[133] Kanan, T., Sadaqa, O., Almhirat, A., and Kanan, E. (2019). Arabic light stemming: A comparative study between p-stemmer, khoja stemmer, and light10 stemmer. In *2019 Sixth International Conference on Social Networks Analysis, Management and Security (SNAMS)*, pages 511–515. IEEE.

[134] Kang, H., Yoo, S. J., and Han, D. (2012). Senti-lexicon and improved naïve bayes algorithms for sentiment analysis of restaurant reviews. *Expert Systems with Applications*, 39(5):6000–6010.

[135] Karami, A. and Zhou, B. (2015). Online review spam detection by new linguistic features. *iConference 2015 Proceedings*.

[136] Karsi, R., Zaim, M., and El Alami, J. (2017). Impact of corpus domain for sentiment classification: An evaluation study using supervised machine learning techniques. In *Journal of Physics: Conference Series*, volume 870, page 12005.

[137] Karumanchi, A., Fu, L., and Deng, J. (2018). Prediction of review sentiment and detection of fake reviews in social media. In *Proceedings of the international conference on information and knowledge engineering (ike)*, pages 181–186. The Steering Committee of The World Congress in Computer Science, Computer ....

[138] Katzenbeisser, S., Kinder, J., and Veith, H. (2011). *Malware Detection*, pages 752–755. Springer US, Boston, MA.

[139] Kauffmann, E., Peral, J., Gil, D., Ferrández, A., Sellers, R., and Mora, H. (2020). A framework for big data analytics in commercial social networks: A case study on sentiment analysis and fake review detection for marketing decision-making. *Industrial Marketing Management*, 90:523–537.

[140] Kaur, G. and Malik, K. (2021). A comprehensive overview of sentiment analysis and fake review detection. *Mobile Radio Communications and 5G Networks*, pages 293–304.

[141] Kaur, R. and Kautish, S. (2022). Multimodal sentiment analysis: A survey and comparison. *Research Anthology on Implementing Sentiment Analysis Across Multiple Disciplines*, pages 1846–1870.

[142] Kennedy, J. and Eberhart, R. (1995). Particle swarm optimization. In *Proceedings of ICNN'95-international conference on neural networks*, volume 4, pages 1942–1948. IEEE.

[143] Kennedy, S., Walsh, N., Sloka, K., Foster, J., and McCarren, A. (2020). Fact or factitious? contextualized opinion spam detection. *arXiv preprint arXiv:2010.15296*.

[144] Khan, L., Amjad, A., Ashraf, N., and Chang, H.-T. (2022). Multi-class sentiment analysis of urdu text using multilingual bert. *Scientific Reports*, 12(1):1–17.

[145] Khemchandani, R., Chandra, S., et al. (2007). Twin support vector machines for pattern classification. *IEEE Transactions on pattern analysis and machine intelligence*, 29(5):905–910.

[146] Kim, W. G., Li, J. J., and Brymer, R. A. (2016). The impact of social media reviews on restaurant performance: The moderating role of excellence certificate. *International Journal of Hospitality Management*, 55:41–51.

[147] Kiruthika, S. and Priyan, V. V. (2021). Detection of online fake reviews based on text emotion. In *Journal of Physics: Conference Series*, volume 1916, page 012153. IOP Publishing.

[148] Klein, G., Pliske, R., Crandall, B., and Woods, D. D. (2005). Problem detection. *Cognition, Technology & Work*, 7(1):14–28.

[149] Korovkinas, K. (2020). A hybrid method for textual data classification based on support vector machine with particle swarm optimization metaheuristic and k-means clustering. In *Joint Proceedings of Baltic DB&IS 2020 Conference Forum and Doctoral Consortium co-located with the 14th International Baltic Conference on Databases and Information Systems (BalticDB&IS 2020) Tallinn, Estonia, June 16-19, 2020*, pages 81–88. CEUR-WS.

[150] Korovkinas, K., Danėnas, P., and Garšva, G. (2020). Support vector machine parameter tuning based on particle swarm optimization metaheuristic. *Nonlinear analysis: modelling and control*, 25(2):266–281.

[151] Kowsari, K., Jafari Meimandi, K., Heidarysafa, M., Mendu, S., Barnes, L., and Brown, D. (2019). Text classification algorithms: A survey. *Information*, 10(4):150.

[152] Krishna, A., Akhilesh, V., Aich, A., and Hegde, C. (2019). Sentiment analysis of restaurant reviews using machine learning techniques. In *Emerging Research in Electronics, Computer Science and Technology*, pages 687–696. Springer.

[153] Krishnaveni, N. and Radha, V. (2021). Performance evaluation of clustering-based classification algorithms for detection of online spam reviews. In *Data Intelligence and Cognitive Informatics*, pages 255–266. Springer.

[154] Krouska, A., Troussas, C., and Virvou, M. (2020). Deep learning for twitter sentiment analysis: the effect of pre-trained word embedding. In *Machine learning paradigms*, pages 111–124. Springer.

[155] Kumar Tripathi, A., Sharma, K., and Bala, M. (2019). Fake review detection in big data using parallel bbo. *International Journal of Information Systems & Management Science*, 2(2).

[156] Kusner, M., Sun, Y., Kolkin, N., and Weinberger, K. (2015). From word embeddings to document distances. In *International conference on machine learning*, pages 957–966. PMLR.

[157] Kusumasondjaja, S., Shanka, T., and Marchegiani, C. (2012). Credibility of online reviews and initial trust: The roles of reviewer's identity and review valence. *Journal of Vacation Marketing*, 18(3):185–195.

[158] Lee, S.-Y., Qiu, L., and Whinston, A. (2018). Sentiment manipulation in online platforms: An analysis of movie tweets. *Production and Operations Management*, 27(3):393–416.

[159] Li, F. H., Huang, M., Yang, Y., and Zhu, X. (2011). Learning to identify review spam. In *Twenty-second international joint conference on artificial intelligence*.

[160] Li, H., Chen, Z., Liu, B., Wei, X., and Shao, J. (2014a). Spotting fake reviews via collective positive-unlabeled learning. In *2014 IEEE international conference on data mining*, pages 899–904. IEEE.

[161] Li, J., Lv, P., Xiao, W., Yang, L., and Zhang, P. (2021a). Exploring groups of opinion spam using sentiment analysis guided by nominated topics. *Expert Systems with Applications*, 171:114585.

[162] Li, J., Ott, M., Cardie, C., and Hovy, E. (2014b). Towards a general rule for identifying deceptive opinion spam. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1566–1576.

[163] Li, J., Zhang, P., and Yang, L. (2021b). An unsupervised approach to detect review spam using duplicates of images, videos and chinese texts. *Computer Speech & Language*, 68:101186.

[164] Li, K., Zhou, G., Yang, Y., Li, F., and Jiao, Z. (2020a). A novel prediction method for favorable reservoir of oil field based on grey wolf optimizer and twin support vector machine. *Journal of Petroleum Science and Engineering*, 189:106952.

[165] Li, L., Yang, L., and Zeng, Y. (2021c). Improving sentiment classification of restaurant reviews with attention-based bi-gru neural network. *Symmetry*, 13(8):1517.

[166] Li, S., Li, G., Law, R., and Paradies, Y. (2020b). Racism in tourism reviews. *Tourism Management*, 80:104100.

[167] Li, Y. and Yang, T. (2018). Word embedding for understanding natural language: a survey. In *Guide to big data applications*, pages 83–104. Springer.

[168] Ligthart, A., Catal, C., and Tekinerdogan, B. (2021). Analyzing the effectiveness of semi-supervised learning approaches for opinion spam classification. *Applied Soft Computing*, 101:107023.

[169] Liu, X. and Xu, H. (2018). Application on target localization based on salp swarm algorithm. In *2018 37th Chinese Control Conference (CCC)*, pages 4542–4545. IEEE.

[170] Liu, Y., Pang, B., and Wang, X. (2019). Opinion spam detection by incorporating multimodal embedded representation into a probabilistic review graph. *Neurocomputing*, 366:276–283.

[171] Liu, Y., Wang, L., Shi, T., and Li, J. (2022). Detection of spam reviews through a hierarchical attention architecture with n-gram cnn and bi-lstm. *Information Systems*, 103:101865.

[172] Lo, A. S. and Yao, S. S. (2019). What makes hotel online reviews credible? *International Journal of Contemporary Hospitality Management*.

[173] Lu, Y., Zhang, L., Xiao, Y., and Li, Y. (2013). Simultaneously detecting fake reviews and review spammers using factor graph model. In *Proceedings of the 5th annual ACM web science conference*, pages 225–233.

[174] Luca, M. and Zervas, G. (2016). Fake it till you make it: Reputation, competition, and yelp review fraud. *Management Science*, 62(12):3412–3427.

[175] Luo, Y. and Xu, X. (2021). Comparative study of deep learning models for analyzing online restaurant reviews in the era of the covid-19 pandemic. *International Journal of Hospitality Management*, 94:102849.

[176] Mani, S., Kumari, S., Jain, A., and Kumar, P. (2018). Spam review detection using ensemble machine learning. In *International Conference on Machine Learning and Data Mining in Pattern Recognition*, pages 198–209. Springer.

[177] Manias, G., Mavrogiorgou, A., Kiourtis, A., and Kyriazis, D. (2020). An evaluation of neural machine translation and pre-trained word embeddings in multilingual neural sentiment analysis. In *2020 IEEE International Conference on Progress in Informatics and Computing (PIC)*, pages 274–283. IEEE.

[178] Martens, D. and Maalej, W. (2019). Towards understanding and detecting fake reviews in app stores. *Empirical Software Engineering*, 24(6):3316–3355.

[179] Mehbodniya, A., Rao, M. V., David, L. G., Nigel, K. G. J., and Vennam, P. (2022). Online product sentiment analysis using random evolutionary whale optimization algorithm and deep belief network. *Pattern Recognition Letters*, 159:1–8.

[180] Mei, B. and Xu, Y. (2020). Multi-task v-twin support vector machines. *Neural Computing and Applications*, 32(15):11329–11342.

[181] Melleng, A., Jurek-Loughrey, A., and Deepak, P. (2019). Sentiment and emotion based representations for fake reviews detection. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019)*, pages 750–757.

[182] Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

[183] Mirjalili, S., Gandomi, A. H., Mirjalili, S. Z., Saremi, S., Faris, H., and Mirjalili, S. M. (2017). Salp swarm algorithm: A bio-inspired optimizer for engineering design problems. *Advances in engineering software*, 114:163–191.

[184] Mirjalili, S. and Lewis, A. (2016). The whale optimization algorithm. *Advances in engineering software*, 95:51–67.

[185] Mirjalili, S. Z., Mirjalili, S., Saremi, S., Faris, H., and Aljarah, I. (2018). Grasshopper optimization algorithm for multi-objective optimization problems. *Applied Intelligence*, 48(4):805–820.

[186] Mirończuk, M. M. and Protasiewicz, J. (2018). A recent overview of the state-of-the-art elements of text classification. *Expert Systems with Applications*, 106:36–54.

[187] Mitchell, M. (1998). *An introduction to genetic algorithms*. MIT press.

[188] Mohamed, E. H., Moussa, M. E., and Haggag, M. H. (2020). An enhanced sentiment analysis framework based on pre-trained word embedding. *International Journal of Computational Intelligence and Applications*, 19(04):2050031.

[189] Mohamed, M. A., Diab, A. A. Z., and Rezk, H. (2019). Partial shading mitigation of pv systems via different meta-heuristic techniques. *Renewable energy*, 130:1159–1175.

[190] Mohammadi, M., Rashid, T. A., Karim, S. H. T., Aldalwie, A. H. M., Tho, Q. T., Bidaki, M., Rahmani, A. M., and Hosseinzadeh, M. (2021). A comprehensive survey and taxonomy of the svm-based intrusion detection systems. *Journal of Network and Computer Applications*, 178:102983.

[191] Mohammadi, S. and Mousavi, M. R. (2020). Investigating the impact of ensemble machine learning methods on spam review detection based on behavioral features. *Journal of Soft Computing and Information Technology*, 9(3):132–147.

[192] Mohawesh, R., Tran, S., Ollington, R., and Xu, S. (2021). Analysis of concept drift in fake reviews detection. *Expert Systems with Applications*, 169:114318.

[193] Möhring, M., Keller, B., Schmidt, R., Gutmann, M., and Dacko, S. (2021). Hotfred: A flexible hotel fake review detection system. In *Information and Communication Technologies in Tourism 2021*, pages 308–314. Springer.

[194] Mukherjee, A., Liu, B., and Glance, N. (2012). Spotting fake reviewer groups in consumer reviews. In *Proceedings of the 21st international conference on World Wide Web*, pages 191–200.

[195] Mukherjee, A., Venkataraman, V., Liu, B., and Glance, N. (2013). What yelp fake review filter might be doing? In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 7.

[196] Muller, V. (2021). How covid-19 changed online reviews.

[197] Munasatya, N. and Novianto, S. (2020). Natural language processing untuk sentimen analisis presiden jokowi menggunakan multi layer perceptron. *Techno. Com*, 19(3):237–244.

[198] Munzel, A. (2016). Assisting consumers in detecting fake reviews: The role of identity information disclosure and consensus. *Journal of Retailing and Consumer Services*, 32:96–108.

[199] Mustaffa, Z., Sulaiman, M. H., and Kahar, M. N. M. (2015). Ls-svm hyper-parameters optimization based on gwo algorithm for time series forecasting. In *2015 4th international conference on software engineering and computer systems (ICSECS)*, pages 183–188. IEEE.

[200] Navarro, J., Deruyver, A., and Parrend, P. (2018). A systematic survey on multi-step attack detection. *Computers & Security*, 76:214–249.

[201] Neethu, M. and Rajasree, R. (2013). Sentiment analysis in twitter using machine learning techniques. In *2013 Fourth International Conference on Computing, Communications and Networking Technologies (ICCCNT)*, pages 1–5. IEEE.

[202] Nejad, S. J., Ahmadi-Abkenari, F., and Bayat, P. (2020). Opinion spam detection based on supervised sentiment analysis approach. In *2020 10th International Conference on Computer and Knowledge Engineering (ICCKE)*, pages 209–214. IEEE.

[203] Nguyen, L.-T. D., Nguyen, N. D., and Bui, K.-H. N. (2021). An embedding method for sentiment classification across multiple languages. In *2021 13th International Conference on Knowledge and Systems Engineering (KSE)*, pages 1–6. IEEE.

[204] Ni, J., Li, J., and McAuley, J. (2019). Justifying recommendations using distantly-labeled reviews and fine-grained aspects. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 188–197.

[205] N.KUMARAN, CHAPALAMADUGU HARITHA CHOWDARY, D. S. (2021). Detection of fake online reviews using semi supervised and supervised learning. *IJARST*.

[206] Noekhah, S., binti Salim, N., and Zakaria, N. H. (2020). Opinion spam detection: Using multi-iterative graph-based model. *Information Processing & Management*, 57(1):102140.

[207] Obiedat, R., Harfoushi, O., Qaddoura, R., Al-Qaisi, L., and Al-Zoubi, A. (2021). An evolutionary-based sentiment analysis approach for enhancing government decisions during covid-19 pandemic: the case of jordan. *Applied Sciences*, 11(19):9080.

[208] Obiedat, R., Qaddoura, R., Ala'M, A.-Z., Al-Qaisi, L., Harfoushi, O., Alrefai, M., and Faris, H. (2022). Sentiment analysis of customers' reviews using a hybrid evolutionary svm-based approach in an imbalanced data distribution. *IEEE Access*, 10:22260–22273.

[209] Oh, Y. W. and Park, C. H. (2021). Machine cleaning of online opinion spam: developing a machine-learning algorithm for detecting deceptive comments. *American behavioral scientist*, 65(2):389–403.

[210] Ott, M., Cardie, C., and Hancock, J. T. (2013). Negative deceptive opinion spam. In *Proceedings of the 2013 conference of the north american chapter of the association for computational linguistics: human language technologies*, pages 497–501.

[211] Ott, M., Choi, Y., Cardie, C., and Hancock, J. T. (2011). Finding deceptive opinion spam by any stretch of the imagination. *arXiv preprint arXiv:1107.4557*.

[212] Oueslati, O., Khalil, A. I. S., and Ounelli, H. (2018). Sentiment analysis for helpful reviews prediction. *International Journal*, 7(3).

[213] Pandey, A. C. and Rajpoot, D. S. (2019). Spam review detection using spiral cuckoo search clustering method. *Evolutionary Intelligence*, 12(2):147–164.

[214] Patil, P. P., Phansalkar, S., Ahirrao, S., and Pawar, A. (2021). Alosi: Aspect-level opinion spam identification. In *Data Science and Security*, pages 135–147. Springer.

[215] Peng, Q. (2013). Store review spammer detection based on review relationship. In *International Conference on Conceptual Modeling*, pages 287–298. Springer.

[216] Peng, Q. and Zhong, M. (2014). Detecting spam review through sentiment analysis. *J. Softw.*, 9(8):2065–2072.

[217] Pessutto, L. R. C., Vargas, D. S., and Moreira, V. P. (2020). Multilingual aspect clustering for sentiment analysis. *Knowledge-Based Systems*, 192:105339.

[218] Pires, T., Schlinger, E., and Garrette, D. (2019). How multilingual is multilingual bert? *arXiv preprint arXiv:1906.01502*.

[219] Plotkina, D., Munzel, A., and Pallud, J. (2020). Illusions of truth—experimental insights into human and algorithmic detections of fake online reviews. *Journal of Business Research*, 109:511–523.

[220] Porter, M. F. (2001). Snowball: A language for stemming algorithms.

[221] Pradhan, S., Amatya, E., and Ma, Y. (2021). Manipulation of online reviews: Analysis of negative reviews for healthcare providers. In *International Conference on Information Resources Management*. AIS eLibrary.

[222] Qaddoura, R., Al-Zoubi, M., Faris, H., Almomani, I., et al. (2021a). A multi-layer classification approach for intrusion detection in iot networks based on deep learning. *Sensors*, 21(9):2987.

[223] Qaddoura, R., Faris, H., Aljarah, I., and Castillo, P. A. (2020). Evocluster: An open-source nature-inspired optimization clustering framework in python. In *International conference on the applications of evolutionary computation (part of EvoStar)*, pages 20–36. Springer.

[224] Qaddoura, R., Faris, H., Aljarah, I., and Castillo, P. A. (2021b). Evocluster: an open-source nature-inspired optimization clustering framework. *SN Computer Science*, 2(3):1–12.

[225] Qi, Z., Tian, Y., and Shi, Y. (2013a). Robust twin support vector machine for pattern classification. *Pattern Recognition*, 46(1):305–316.

[226] Qi, Z., Tian, Y., and Shi, Y. (2013b). Structural twin support vector machine for classification. *Knowledge-Based Systems*, 43:74–81.

[227] Rajamohana, S. and Umamaheswari, K. (2017). Hybrid optimization algorithm of improved binary particle swarm optimization (ibpso) and cuckoo search for review spam detection. In *Proceedings of the 9th International Conference on Machine Learning and Computing*, pages 238–242.

[228] Rajamohana, S., Umamaheswari, K., and Keerthana, S. V. (2017a). An effective hybrid cuckoo search with harmony search for review spam detection. In *2017 Third International Conference on Advances in Electrical, Electronics, Information, Communication and Bio-Informatics (AEEICB)*, pages 524–527. IEEE.

[229] Rajamohana, S. P., Umamaheswari, K., and Abirami, B. (2017b). Adaptive binary flower pollination algorithm for feature selection in review spam detection. In *2017 International Conference on Innovations in Green Energy and Healthcare Technologies (IGEHT)*, pages 1–4. IEEE.

[230] Rani, M. S. and Sumathy, S. (2018). Online social networking services and spam detection approaches in opinion mining-a review. *International Journal of Web Based Communities*, 14(4):353–378.

[231] Rapid7 (2021). Cthreat-detection.

[232] Rasool, A., Jiang, Q., Qu, Q., and Ji, C. (2021). Wrs: a novel word-embedding method for real-time sentiment with integrated lstm-cnn model. In *2021 IEEE International Conference on Real-time Computing and Robotics (RCAR)*, pages 590–595. IEEE.

[233] Rastogi, A., Mehrotra, M., and Ali, S. S. (2021). Effect of various factors in context of feature selection on opinion spam detection. In *2021 11th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*, pages 778–783. IEEE.

[234] Ravi, K. and Ravi, V. (2015). A survey on opinion mining and sentiment analysis: tasks, approaches and applications. *Knowledge-Based Systems*, 89:14–46.

[235] Rayana, S. and Akoglu, L. (2015). Collective opinion spam detection: Bridging review networks and metadata. In *Proceedings of the 21th acm sigkdd international conference on knowledge discovery and data mining*, pages 985–994.

[236] Ren, Y., Ji, D., and Yin, L. (2014). Deceptive reviews detection based on semi-supervised learning algorithm. *J. Sichuan Univ.(Eng. Sci. Ed.)*, 46(3):62–69.

[237] Reporter, T. (2012). Tripadvisor told to stop claiming reviews are 'trusted and honest.

[238] Rezaeinia, S. M., Rahmani, R., Ghodsi, A., and Veisi, H. (2019). Sentiment analysis based on improved pre-trained word embeddings. *Expert Systems with Applications*, 117:139–147.

[239] Rezk, H. and Fathy, A. (2017). A novel optimal parameters identification of triple-junction solar cell based on a recently meta-heuristic water cycle algorithm. *Solar Energy*, 157:778–791.

[240] Rodrigues, A. P., Fernandes, R., Shetty, A., Lakshmanna, K., Shafi, R. M., et al. (2022). Real-time twitter spam detection and sentiment analysis using machine learning and deep learning techniques. *Computational Intelligence and Neuroscience*, 2022.

[241] Rokade, P. P. and Aruna, K. D. (2019). Business intelligence analytics using sentiment analysis-a survey. *International Journal of Electrical and Computer Engineering*, 9(1):613.

[242] Rubin, V. L., Conroy, N., Chen, Y., and Cornwell, S. (2016). Fake news or truth? using satirical cues to detect potentially misleading news. In *Proceedings of the second workshop on computational approaches to deception detection*, pages 7–17.

[243] Sadewo, W., Rustam, Z., Hamidah, H., and Chusmarsyah, A. R. (2020). Pancreatic cancer early detection using twin support vector machine based on kernel. *Symmetry*, 12(4):667.

[244] Sahu, P. C., Prusty, R. C., and Panda, S. (2018). Salp swarm optimized multistage pdf plus (1+ pi) controller in agc of multi source based nonlinear power system. In *International Conference on Soft Computing Systems*, pages 789–800. Springer.

[245] Salehi-Esfahani, S. and Ozturk, A. B. (2018). Negative reviews: Formation, spread, and halt of opportunistic behavior. *International Journal of Hospitality Management*, 74:138–146.

[246] Salminen, J., Kandpal, C., Kamel, A. M., Jung, S.-g., and Jansen, B. J. (2022). Creating and detecting fake reviews of online products. *Journal of Retailing and Consumer Services*, 64:102771.

[247] Sarmadi, H. and Karamodin, A. (2020). A novel anomaly detection method based on adaptive mahalanobis-squared distance and one-class knn rule for structural health monitoring under environmental effects. *Mechanical systems and signal processing*, 140:106495.

[248] Sartakhti, J. S., Afrabandpey, H., and Saraee, M. (2017). Simulated annealing least squares twin support vector machine (sa-lstsvm) for pattern classification. *Soft Computing*, 21(15):4361–4373.

[249] Saumya, S. and Singh, J. P. (2018). Detection of spam reviews: a sentiment analysis approach. *Csi Transactions on ICT*, 6(2):137–148.

[250] Saumya, S. and Singh, J. P. (2020). Spam review detection using lstm autoencoder: an unsupervised approach. *Electronic Commerce Research*, pages 1–21.

[251] Schuckert, M., Liu, X., and Law, R. (2016). Insights into suspicious online ratings: direct evidence from tripadvisor. *Asia Pacific Journal of Tourism Research*, 21(3):259–272.

[252] Şen, D., Dönmez, C. Ç., and Yıldırım, U. M. (2020). A hybrid bi-level metaheuristic for credit scoring. *Information Systems Frontiers*, 22(5):1009–1019.

[253] Shahariar, G., Biswas, S., Omar, F., Shah, F. M., and Hassan, S. B. (2019). Spam review detection using deep learning. In *2019 IEEE 10th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*, pages 0027–0033. IEEE.

[254] Shahi, T., Sitaula, C., and Paudel, N. (2022). A hybrid feature extraction method for nepali covid-19-related tweets classification. *Computational Intelligence and Neuroscience*, 2022.

[255] Shan, G., Zhou, L., and Zhang, D. (2021). From conflicts and confusion to doubts: Examining review inconsistency for fake review detection. *Decision Support Systems*, 144:113513.

[256] Shao, Y.-H., Chen, W.-J., Zhang, J.-J., Wang, Z., and Deng, N.-Y. (2014). An efficient weighted lagrangian twin support vector machine for imbalanced data classification. *Pattern Recognition*, 47(9):3158–3167.

[257] Shao, Y.-H., Zhang, C.-H., Wang, X.-B., and Deng, N.-Y. (2011). Improvements on twin support vector machines. *IEEE transactions on neural networks*, 22(6):962–968.

[258] Sharef, N. M., Zin, H. M., and Nadali, S. (2016). Overview and future opportunities of sentiment analysis approaches for big data. *J. Comput. Sci.*, 12(3):153–168.

[259] Sharif, W., Tun, U., Onn, H., Tri, I., and Yanto, R. (2019). An optimised support vector machine with ringed seal search algorithm for efficient text classification. *J. Eng. Sci. Technol*, 14(3):1601–1613.

[260] Sharma, K., Qian, F., Jiang, H., Ruchansky, N., Zhang, M., and Liu, Y. (2019). Combating fake news: A survey on identification and mitigation techniques. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 10(3):1–42.

[261] Sharma, S. and Jain, A. (2020). Role of sentiment analysis in social media security and analytics. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 10(5):e1366.

[262] Shehnepoor, S., Salehi, M., Farahbakhsh, R., and Crespi, N. (2017). Netspam: A network-based spam detection framework for reviews in online social media. *IEEE Transactions on Information Forensics and Security*, 12(7):1585–1595.

[263] Shetgaonkar, P., Rodrigues, J. T., Aswale, S., Gonsalves, V. L. K., Rodrigues, J. C., and Naik, A. (2021). Fake review detection using sentiment analysis and deep learning. In *2021 International Conference on Technological Advancements and Innovations (ICTAI)*, pages 140–145. IEEE.

[264] Singh, A. and Chatterjee, K. (2022). A comparative approach for opinion spam detection using sentiment analysis. In *Proceedings of First International Conference on Computational Electronics for Wireless Communications*, pages 511–522. Springer.

[265] Singhal, P. and Bhattacharyya, P. (2016). Borrow a little from your rich cousin: Using embeddings and polarities of english words for multilingual sentiment classification. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 3053–3062.

[266] Sitaula, C., Basnet, A., Mainali, A., and Shahi, T. (2021). Deep learning-based methods for sentiment analysis on nepali covid-19-related tweets. *Computational Intelligence and Neuroscience*, 2021.

[267] Sivakumar, S. and Rajalakshmi, R. (2021). Analysis of sentiment on movie reviews using word embedding self-attentive lstm. *International Journal of Ambient Computing and Intelligence (IJACI)*, 12(2):33–52.

[268] Smith, C. and Brooks, D. J. (2012). *Security science: The theory and practice of security*. Butterworth-Heinemann.

[269] Sun, Z.-X., Hu, R., Qian, B., Liu, B., and Che, G.-L. (2018). Salp swarm algorithm based on blocks on critical path for reentrant job shop scheduling problems. In *International conference on intelligent computing*, pages 638–648. Springer.

[270] Sundar, A. P., Lilt, F., Zou, X., and Gao, T. (2020). Deepdynamic clustering of spam reviewers using behavior-anomaly-based graph embedding. In *GLOBECOM 2020-2020 IEEE Global Communications Conference*, pages 01–06. IEEE.

[271] Świetlicka, I., Kuniszyk-Jóźkowiak, W., and Świetlicki, M. (2022). Artificial neural networks combined with the principal component analysis for non-fluent speech recognition. *Sensors*, 22(1):321.

[272] Tang, J., Xue, Y., Wang, Z., Hu, S., Gong, T., Chen, Y., Zhao, H., and Xiao, L. (2022). Bayesian estimation-based sentiment word embedding model for sentiment analysis. *CAAI Transactions on Intelligence Technology*, 7(2):144–155.

[273] Tang, X., Qian, T., and You, Z. (2020). Generating behavior features for cold-start spam review detection with adversarial learning. *Information Sciences*, 526:274–288.

[274] Thapa, S., Adhikari, S., Ghimire, A., and Aditya, A. (2020). Feature selection based twin-support vector machine for the diagnosis of parkinson's disease. In *2020 IEEE 8th R10 humanitarian technology conference (R10-HTC)*, pages 1–6. IEEE.

[275] Tian, Y., Mirzabagheri, M., Tirandazi, P., and Bamakan, S. M. H. (2020). A non-convex semi-supervised approach to opinion spam detection by ramp-one class svm. *Information Processing & Management*, 57(6):102381.

[276] Tian, Z., Luo, C., Qiu, J., Du, X., and Guizani, M. (2019). A distributed deep learning system for web attack detection on edge devices. *IEEE Transactions on Industrial Informatics*, 16(3):1963–1971.

[277] Torres, E. N., Singh, D., and Robertson-Ring, A. (2015). Consumer reviews and the creation of booking transaction value: Lessons from the hotel industry. *International Journal of Hospitality Management*, 50:77–83.

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

[279] Tubishat, M., Idris, N., and Abushariah, M. A. (2018). Implicit aspect extraction in sentiment analysis: Review, taxonomy, oppportunities, and open challenges. *Information Processing & Management*, 54(4):545–563.

[280] Valant, J. (2015). Online consumer reviews: The case of misleading or fake reviews. *European Parliamentary Research Service*, 2.

[281] Van Laarhoven, P. J. and Aarts, E. H. (1987). Simulated annealing. In *Simulated annealing: Theory and applications*, pages 7–15. Springer.

[282] Vapnik, V. (1999). An overview of statistical learning theory. *IEEE Transactions on Neural Networks*, 5:988–999.

[283] Vapnik, V. (2013). *The nature of statistical learning theory*. Springer Science & Business Media.

[284] Vilares, D., Hermo, M., Alonso, M. A., Gómez-Rodríguez, C., and Vilares, J. (2014). Lys at clef replab 2014: Creating the state of the art in author influence ranking and reputation classification on twitter. *Clef (working notes)*, pages 1468–1478.

[285] Visani, C., Jadeja, N., and Modi, M. (2017). A study on different machine learning techniques for spam review detection. In *2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS)*, pages 676–679. IEEE.

[286] Wang, G., Xie, S., Liu, B., and Philip, S. Y. (2011). Review graph based online store review spammer detection. In *2011 IEEE 11th international conference on data mining*, pages 1242–1247. IEEE.

[287] Wang, H. and Zhou, Z. (2017). An improved rough margin-based ν-twin bounded support vector machine. *Knowledge-Based Systems*, 128:125–138.

[288] Wang, J. and Liang, X. (2013). Discovering the rating pattern of online reviewers through data coclustering. In *2013 IEEE International Conference on Intelligence and Security Informatics*, pages 374–376. IEEE.

[289] Wang, J., Wen, R., Wu, C., and Xiong, J. (2020). Analyzing and detecting adversarial spam on a large-scale online app review system. In *Companion Proceedings of the Web Conference 2020*, pages 409–417.

[290] Wassan, S., Chen, X., Shen, T., Waqar, M., and Jhanjhi, N. (2021). Amazon product sentiment analysis using machine learning techniques. *Revista Argentina de Clínica Psicológica*, 30(1):695.

[291] Wen, J., Hu, J., Shi, H., Wang, X., Yuan, C., Han, J., and Guo, T. (2020). Fusion-based spammer detection method by embedding review texts and weak social relations. In *2020 IEEE Intl Conf on Parallel & Distributed Processing with Applications, Big Data & Cloud Computing, Sustainable Computing & Communications, Social Computing & Networking (ISPA/BDCloud/SocialCom/SustainCom)*, pages 329–336. IEEE.

[292] Whitney, L. (2013). Companies to pay $350,000 fine over fake online reviews.

[293] Wu, Y., Ngai, E. W., Wu, P., and Wu, C. (2020a). Fake online reviews: Literature review, synthesis, and directions for future research. *Decision Support Systems*, 132:113280.

[294] Wu, Z., Cao, J., Wang, Y., Wang, Y., Zhang, L., and Wu, J. (2020b). hpsd: A hybrid pu-learning-based spammer detection model for product reviews. *IEEE Transactions on Cybernetics*, 50(4):1595–1606.

[295] Xi, E., Bing, S., and Jin, Y. (2017). Capsule network performance on complex data. *arXiv preprint arXiv:1712.03480*.

[296] Xu, D. (2022). Efficient index and retrieval algorithm of geospatial big data based on particle swarm optimization. In *Proceedings of the 3rd Asia-Pacific Conference on Image Processing, Electronics and Computers*, pages 975–979.

[297] Xu, K. and Wan, X. (2017). Towards a universal sentiment classifier in multiple languages. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 511–520.

[298] Ya, Z., Qingqing, Z., Yuhan, W., and Shuai, Z. (2020). Lda_rad: A spam review detection method based on topic model and reviewer anomaly degree. In *Journal of Physics: Conference Series*, volume 1550, page 022008. IOP Publishing.

[299] Yang, Y., Park, S., and Hu, X. (2018). Electronic word of mouth and hotel performance: A meta-analysis. *Tourism management*, 67:248–260.

[300] Yao, J., Zheng, Y., and Jiang, H. (2021). An ensemble model for fake online review detection based on data resampling, feature pruning, and parameter optimization. *IEEE Access*, 9:16914–16927.

[301] Yu, B., Zhou, J., Zhang, Y., and Cao, Y. (2017). Identifying restaurant features via sentiment analysis on yelp reviews. *arXiv preprint arXiv:1709.08698*.

[302] Yuan, F., Guo, J., Xiao, Z., Zeng, B., Zhu, W., and Huang, S. (2019). A transformer fault diagnosis model based on chemical reaction optimization and twin support vector machine. *Energies*, 12(5):960.

[303] Zahoor, K., Bawany, N. Z., and Hamid, S. (2020). Sentiment analysis and classification of restaurant reviews using machine learning. In *2020 21st International Arab Conference on Information Technology (ACIT)*, pages 1–6. IEEE.

[304] Zelenka, J., Azubuike, T., and Pásková, M. (2021). Trust model for online reviews of tourism services and evaluation of destinations. *Administrative Sciences*, 11(2):34.

[305] Zeybek, S., Pham, D. T., Koç, E., and Seçer, A. (2021). An improved bees algorithm for training deep recurrent networks for sentiment classification. *Symmetry*, 13(8):1347.

[306] Zhang, S., Zhang, X., Chan, J., and Rosso, P. (2019a). Irony detection via sentiment-based transfer learning. *Information Processing & Management*, 56(5):1633–1644.

[307] Zhang, X., Wang, J., Liu, Z., and Wang, J. (2019b). Weak feature enhancement in machinery fault diagnosis using empirical wavelet transform and an improved adaptive bistable stochastic resonance. *ISA transactions*, 84:283–295.

[308] Zhao, H., Liu, Z., Yao, X., and Yang, Q. (2021). A machine learning-based sentiment analysis of online product reviews with a novel term weighting and feature selection approach. *Information Processing & Management*, 58(5):102656.

[309] Zhao, R. (2018). Mafengwo accused of faking 85

[310] Zhou, L. and Zhang, Q. (2021). Recognition of false comments in e-commerce based on deep learning confidence network algorithm. *Information Systems and e-Business Management*, pages 1–18.

[311] Zhu, F. and Zhang, X. (2010). Impact of online consumer reviews on sales: The moderating role of product and consumer characteristics. *Journal of marketing*, 74(2):133–148.

[312] Zhu, X. and Goldberg, A. B. (2009). Introduction to semi-supervised learning. *Synthesis lectures on artificial intelligence and machine learning*, 3(1):1–130.

[313] Zunic, A., Corcoran, P., Spasic, I., et al. (2020). Sentiment analysis in health and well-being: systematic review. *JMIR medical informatics*, 8(1):e16023.