Improving Arabic Spam classification in social media using hyperparameters tuning and Particle Swarm Optimization

Amr Mohamed El Koshiry[*,1,2], Entesar H. Ibraheem Eliwa[3,4], Ahmed Omar[4]

[1]Department of Curricula and Teaching Methods, College of Education, King Faisal University,
P.O. Box: 400 Al-Ahsa, 31982, Saudi Arabia
[2]Faculty of Specific Education, Minia university, Egypt
[3]Department of Mathematics and Statistics, College of Science, King Faisal University,
P.O. Box: 400 Al-Ahsa, 31982, Saudi Arabia,
[4]Department of Computer Science, Faculty of Science, Minia University,
P.O. Box:91519, Minia, Egypt

Emails: aalkoshiry@kfu.edu.sa; eheliwa@kfu.edu.sa; ahmed.omar@mu.edu.eg

Abstract

Online social networks continue to evolve, serving a variety of purposes, such as sharing educational content, chatting, making friends and followers, sharing news, and playing online games. However, the widespread flow of unwanted messages poses significant problems, including reducing online user interaction time, extremist views, reducing the quality of information, especially in the educational field. The use of coordinated automated accounts or robots on social networking sites is a common tactic for spreading unwanted messages, rumors, fake news, and false testimonies for mass communication or targeted users. Since users (especially in the educational field) receive many messages through social media, they often fail to recognize the content of unwanted messages, which may contain harmful links, malicious programs, fake accounts, false reports, and misleading opinions. Therefore, it is vital to regulate and classify disturbing texts to enhance the security of social media. This study focuses on building an Arabic disturbing message dataset extracted from Twitter, which consists of 14,250 tweets. Our proposed methodology includes applying new tag identification technology to collected tweets. Then, we use prevailing machine learning algorithms to build a model for classifying disturbing messages in Arabic, using effective parameter tuning methods to obtain the most suitable parameters for each algorithm. In addition, we use particle swarm optimization to identify the most relevant features to improve the classification performance. The results indicate a clear improvement in the classification performance from 0.9822 to 0.98875, with a 50% reduction in the feature set. Our study focuses on Arabic spam messages, classifying spam messages, tuning effective parameters, and selecting features as key areas of investigation.

Keywords: Arabic Spam; Spam Classification; Hyperparameters Tuning; Feature Selection.

1. Introduction

In recent times, the proliferation of social media has been remarkable. With the emergence of social media platforms like Facebook, Twitter, Instagram, and TikTok, an enormous number of individuals worldwide have become active participants on social media. The extent of content generated on these platforms is astonishing, as users share a broad range of information, from personal updates, political opinions, memes, educational material, to viral videos. This unprecedented surge in the creation and sharing of content on social media has altered the way we communicate, connect and consume information. While the rise of social media has undoubtedly led to several advantages, such as increased connectivity and access to information, it has also presented significant challenges, such as the propagation of misinformation, spam, cyberbullying, and privacy concerns [1][2]. Spam refers to unsolicited communications that are distributed in significant quantities, encompassing various types of

Doi: https://doi.org/10.54216/FPA.160101
Received: July 21, 2023 Revised: November 19, 2023 Accepted: April 02, 2024

content such as phone numbers, popular hashtags, harmful shortened URLs, images that hide URLs, healthcare tips, pornographic materials, stock market schemes, fraudulent advertisements, fake reviews, misleading news, and political manipulation. The main objective of spammers is to generate income. Moreover, spammers conduct illegal activities, such as advertising, phishing, espionage, cyberbullying, and perpetrating violence against women by gaining the trust of unsuspecting communities [3]. Social spam presents a continuous obstacle for online information systems, encompassing unsolicited messages and reviews on various platforms, including email and social networks. The origins of the term "spam" can be traced back to 1996, and it has since become a significant issue for search engines and social media enterprises. In recent years, prominent corporations have prioritized the identification and mitigation of social spam, allocating considerable resources to researching this field. The diverse manifestations of social spam may include tweets, messages, fictitious reviews, false friends, and malicious links [4]. The rapid spread of medical misinformation and unverified content concerning the COVID-19 pandemic on social media is a significant cause for concern. It is imperative to minimize the prevalence of rumors and false information during this crisis, as it has the potential to induce fear, anxiety, and distress among individuals, possibly resulting in the onset of psychiatric disorders [5]. Spam has a considerable effect on academic communities, including those comprised of students, teachers, and researchers. The abundance of unwelcome messages can be a source of distraction, taking up valuable time that could otherwise be used for learning, collaboration, or discussion. Furthermore, spam communications may harbor malware or phishing links that can compromise users' data security, leaving them vulnerable to identity theft, financial fraud, or loss of data. Additionally, the sheer volume of spam content can cause servers to become overwhelmed, ultimately leading to a slow-down or crash of email systems, which can hinder access to crucial information and negatively impact communication. To minimize the impact of spam on academic groups, it is imperative to deploy effective spam filters, train users on secure email practices, and cultivate a sense of responsible email behavior. [6]. Numerous studies have been conducted to detect spam in English language, but there has been relatively little research on Arabic, which presents unique challenges [7]. Arabic is a Semitic language that is closely associated with Arabic culture and Islam and serves as the language of holy texts for Muslims worldwide (an estimated 1.9 billion individuals). Moreover, Arabic is the mother tongue of approximately 422 million people and there are over 226 million Arabic-speaking internet users. In recent years, the volume of online Arabic content has increased considerably, accounting for more than 3% of all online content and ranking ninth overall. Unfortunately, about one-third of all Arabic content on the internet is of low quality and is produced by social media users. This highlights the pressing need for reliable and efficient approaches to analyze and classify Arabic text [8]. In the field of machine learning classification, the main objective is to construct a model that can accurately predict the classification of new data points based on patterns and relationships present within a given dataset. However, to achieve the best possible classification performance, it is often necessary to carefully choose the relevant features and hyperparameters for the model. The process of hyperparameter tuning involves selecting the optimal values for various parameters used to configure a machine learning model, which can be a time-consuming and computationally expensive process, but is critical for achieving optimal performance. Additionally, feature selection is a crucial technique used to identify the most informative features in a dataset that are relevant for the classification task, with the goal of reducing overfitting, improving model accuracy, and increasing efficiency [9][10]. In this academic paper, our initial objective was to create a new dataset of Arabic spam, which covers a wide range of contemporary topics, including online learning and COVID-19, by utilizing a new hybrid annotation technique that facilitates the annotation process. We then evaluated the effectiveness of popular machine learning algorithms for classifying Arabic spam. Additionally, we sought to improve the classification accuracy by employing three hyperparameter-tuning methods from the algorithm perspective, and the three most used feature representation techniques, coupled with a feature selection technique, from the data perspective. Our contributions to this field can be summarized as follows:

- Development of a freely available Arabic spam dataset using a novel annotation approach that combines unsupervised and manual annotation.
- Comparison of various classification algorithms for detecting Arabic spam.
- A comprehensive comparative analysis of three hyperparameter tuning algorithms.
- Enhancement of spam classification accuracy by fine-tuning classification algorithms.
- Finally, by using PSO feature selection, we were able to improve the classification accuracy to 0.9878 while using only half of the features.

The structure of the paper is delineated as follows. The relevant literature is expounded upon in Section 2, while Section 3 provides an overview of the primary techniques. Our approach to Arabic spam analysis is elucidated in Section 4. Finally, Sections 5 and 6 are respectively devoted to presenting the results of our experiments and drawing conclusions based on those results.

Doi: https://doi.org/10.54216/FPA.160101
Received: July 21, 2023 Revised: November 19, 2023 Accepted: April 02, 2024

2. Related Work

[11] In their study, researchers created a dataset of 9,697 Arabic posts and comments obtained from Algerian Facebook pages and classified them into 1,112 spam comments and 8,585 non-spam comments. They also developed a balanced version of the dataset that contained an equal number of spam and non-spam comments. To prepare the dataset for analysis, they performed several preprocessing steps and proposed nine features to represent the data. The researchers evaluated the dataset using seven machine learning classification algorithms, and found that in the unbalanced version, the J48 algorithm performed the best with an accuracy of 0.9173, while in the balanced version, it had an accuracy of 0.7657.

In their study, [12] utilized a dataset originally collected by [13], which included 3,503 Arabic tweets. The tweets were divided into two categories: 1,944 spam tweets and 1,559 non-spam tweets. The authors applied several preprocessing techniques to the dataset and used two word-embedding methods, namely CBOW and Skip-gram, to represent the extracted features. Additionally, they employed three machine learning classification algorithms, namely Support Vector Machines (SVM), Naive Bayes (NB), and Decision Trees (DT), to classify the features. The experimental results indicated that the SVM algorithm, in combination with the Skip-gram feature representation, produced the highest accuracy of 0.8732.

[4] A dataset was constructed by utilizing a particular hashtag on Twitter and then manually labeling it into two categories: spam tweets and non-spam tweets, with each category consisting of 2,500 tweets. The study employed three machine learning algorithms, namely Naive Bayes, Logistic Regression, and Stochastic Gradient Descent algorithms, along with two optimization algorithms, Whale Optimization WOA and Genetic Algorithm (GA), to develop a model for identifying spam tweets. The findings of the study indicate that the Logistic Regression algorithm outperformed the other algorithms with an accuracy of 0.895. However, after utilizing WOA, the accuracy improved to 0.911.

In their study, [14] presented a novel approach for identifying spam messages in SMS using a hybrid deep learning model. The model was developed by merging a dataset of 2,730 Arabic messages collected from local smartphones with an English SMS spam dataset retrieved from the UCI Repository, which comprised a total of 8,304 messages classified into 785 spam and 7,519 non-spam messages. The authors evaluated the model's performance by applying nine machine learning algorithms and two deep learning models - CNN and LSTM. The results showed that the hybrid model combining CNN and LSTM achieved the highest accuracy of 0.9837.

The authors of [15] introduced an extensive dataset of Arabic advertisement (Spam) tweets, which contained 134,222 tweets, out of which 12,541 were spam tweets and 121,681 were non-spam tweets. The dataset was manually annotated, and the authors conducted a thorough analysis of the tweet characteristics to determine the targets and topics, as well as the characteristics of spam accounts. To detect spam tweets in the dataset, the authors utilized Support Vector Machines (SVMs) and contextual embedding-based models. Their approach achieved a macro-averaged F1 score of 0.981.

In [16], the authors translated the spam dataset created by [17] from English to Arabic, resulting in a dataset containing 1600 tweets. To address the lack of annotated Arabic resources for deception detection, the authors proposed a solution that involved exploring and suggesting a set of Arabic semantic features inspired by rhetoric phrase dependency algorithms. They implemented this approach using a semi-supervised SVM, which helped improve the system accuracy to 0.8599.

In [18], the authors conducted a study on the characteristics of spam profiles on Twitter in four different languages, including Arabic. The dataset used in the study was collected using Twitter API and manually annotated by three experts in the domain of spam on social media. Preprocessing techniques were applied to clean the dataset. The authors used five well-known classification algorithms and five filter-based feature selection methods to perform their experiments. The results showed that kNN performed better than the other classifiers, achieving an accuracy of 0.979. Additionally, using the feature selection method ReliefF helped to further improve the accuracy to 0.984.

In [19], the authors generated an Arabic spam tweets dataset by collecting and manually annotating Arabic spam tweets. To tackle the issue of imbalance, they utilized data augmentation techniques to increase the number of spam tweets in the dataset. The final dataset comprised 6,228 tweets, divided into 1,648 spam and 4,580 non-spam tweets. The authors applied three machine learning algorithms to the dataset, both before and after augmentation. Their experiments revealed that Linear SVC with the augmented dataset achieved the highest accuracy, with a value of 0.923.

In [20], the authors introduced a keyword-based approach for detecting Arabic spam reviews. This method involved extracting crucial subsets of words from the original text using TF-IDF matrix and filter methods. They applied this approach to a dataset of 3,000 Arabic comments that were extracted from Facebook pages. The authors

Doi: https://doi.org/10.54216/FPA.160101
Received: July 21, 2023 Revised: November 19, 2023 Accepted: April 02, 2024

used four different machine learning algorithms, including C4.5, kNN, SVM, and Naïve Bayes classifiers, in the detection process. The experiments showed that the Decision Tree classifier performed better than the other classification algorithms, achieving a detection accuracy of 0.9263.

Table 1 summarizes the previous studies in terms of dataset size, platform, feature representation techniques, Classification Algorithms, and accuracy.

Table 1: Summary of Previous Studies

Study	Dataset size	Platform	Feature Representation	Classification Algorithms	Accuracy
[11]	9,697	Facebook	Nine features	J48, JRip, NB, RF, SVM, kNN, MLP	0.9173(unbalanced), 0.7657 (balanced)
[12]	3,503	Twitter	CBOW and Skip-gram	SVM, NB, Decision Trees	0.8732
[4]	5,000	Twitter	TF-IDF	NB, LR, SGD	0.895 (LR), 0.911(Logistic Regression + WOA)
[14]	11,034	SMS	Word embeddings	Nine ML algorithms, CNN, LSTM	0.9837
[15]	134,222	Twitter	Contextual embeddings, SVM	SVM, contextual embeddings	0.981 (macro-averaged F1)
[16]	1600	Twitter	Arabic semantic features	Semi-supervised SVM	0.8599
[18]	498	Twitter	Bag of words	SVM, NB, kNN, RF, MLP	97.9%
[19]	6,228	Twitter	TF-IDF	SVM, RF, Linear SVC	0.923
[20]	3,000	Facebook	TF-IDF, filter methods	C4.5, kNN, SVM, NB	0.9263

3. Background

3.1. Hyperparameter tuning

Optimizing the performance of machine learning algorithms for classification tasks involves a critical step known as hyperparameter tuning. Hyperparameters are predetermined values that must be established before the training process begins, and they can significantly influence the final outcome of the algorithm. The process of hyperparameter tuning entails selecting the most effective values for these parameters in order to maximize the model's performance on a validation set [21][22].

Precise adjustment of hyperparameters can substantially enhance the performance of a model. Nonetheless, this tuning process can be both resource-intensive and time-consuming, as it often entails the training and evaluation of numerous model variations with different parameter settings. To overcome this challenge, researchers have created various methods for automating hyperparameter tuning, including grid search, random search, and Bayesian optimization. By implementing these techniques, the time and resources needed for hyperparameter tuning can be significantly reduced while simultaneously improving the overall performance of the model [23].

Automated techniques can also be employed for hyperparameter tuning of machine learning models, in addition to manual tuning. Three common automated techniques for this purpose include grid search, random search, and genetic algorithms. Grid search involves the definition of a set of possible values for each hyperparameter, followed by evaluating the model's performance for every possible combination of hyperparameter values in a grid. While this method ensures the optimal values are found within the search space, it can be computationally expensive and exhaustive for models with numerous hyperparameters [22].

Random search, on the other hand, involves the random sampling of hyperparameter values from a defined search space, followed by evaluating the model's performance for each sampled combination of hyperparameters. This method is more efficient than grid search when the search space is extensive, and it typically results in superior hyperparameters when compared to grid search. Genetic algorithms (GA) are optimization algorithms that imitate the natural selection process to determine the optimal solution to a problem. Regarding hyperparameter tuning, GA involves defining a population of potential solutions (i.e., hyperparameter configurations) and using selection,

Doi: https://doi.org/10.54216/FPA.160101
Received: July 21, 2023 Revised: November 19, 2023 Accepted: April 02, 2024

mutation, and crossover operations to evolve the population over numerous generations until an optimal solution is found. Although this technique can effectively find global optima, it can be computationally expensive [24].

3.2. Cross Validation

Cross-validation (CV) is a statistical approach utilized to assess the accuracy of machine learning models when data is restricted. The model's performance on new data is uncertain after training it, and evaluating its accuracy on unseen data is necessary. To accomplish this, cross-validation is utilized to assess the model's effectiveness by allocating a section of the data for testing and validation. K-Fold cross-validation is a commonly used technique that involves dividing the dataset into K folds or sections. The model is trained on K-1 folds while one-fold is utilized for validation. This process is repeated, with each fold serving as a validation set, resulting in K scores. To obtain the final score for the model, the scores from each fold are averaged, as illustrated in Figure 1 [25].

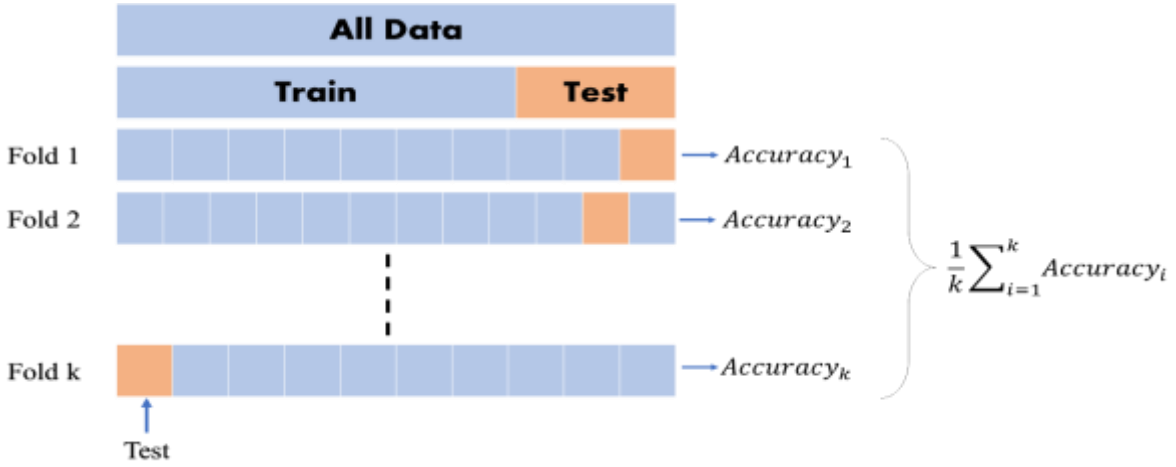

Figure 1: Cross Validation

3.3. Particle Swarm Optimization (PSO)

Particle Swarm Optimization (PSO) is a well-known meta-heuristic algorithm inspired by nature that is widely used as an effective optimization tool in various applications. PSO has produced more variations than any other meta-heuristic algorithm due to the flexibility of its parameters and concepts. The application of PSO for feature selection, which is inspired by social behaviors observed in bird flocking, has sparked significant research interest. PSO is a computationally efficient type of swarm intelligence optimization algorithm that converges rapidly. In PSO, each solution is represented as a particle within a swarm, each with its own velocity and position. The position and velocity of each particle are updated based on its own experience and that of its neighbors. Personal best and global best refer to the particle's previous best position and the best position achieved by the entire population of particles, respectively [8].

The objective of the PSO algorithm is to identify the optimal solution by modifying the velocity and position of each particle based on its personal best and global best solutions. The algorithm terminates when a predetermined stopping criterion is satisfied, such as reaching the maximum number of iterations or achieving the best fitness value. PSO is known for its computational efficiency and fast convergence, which makes it a desirable algorithm for feature selection. In PSO-based feature selection, each feature is represented as a particle, and the goal is to identify a subset of features that maximizes the model's classification accuracy. PSO has been successfully applied to feature selection in diverse domains, including bioinformatics, image processing, and text classification [26].

3.4. Latent Dirichlet Allocation (LDA)

Latent Dirichlet Allocation (LDA) is a popular probabilistic generative model used in natural language processing for topic modeling. Its fundamental assumption is that every document in a corpus represents a combination of various topics, each defined as a probability distribution over words. LDA's objective is to reveal these hidden topics by examining how words co-occur across different documents. Essentially, LDA seeks to deduce the probability distribution over topics for each document in the corpus, along with the probability distribution over

Doi: https://doi.org/10.54216/FPA.160101
Received: July 21, 2023 Revised: November 19, 2023 Accepted: April 02, 2024

words for each topic. The resulting topic model can be leveraged for a range of applications such as text classification, document clustering, and information retrieval. However, due to its high computational complexity, LDA is usually employed on smaller datasets or with specialized software or hardware implementations. [27].

4. Methodology

The Arabic spam classification method proposed in this study consists of four primary phases: (1) Building the dataset, (2) Annotating the data, (3) Preprocessing the data, and (4) Developing the spam classification model. Each of these phases will be described in detail in the subsequent sections.

4.1. Dataset Building Phase

We employed the use of Twitter API to acquire tweets that contained particular keywords pertaining to spam text. Table 2 displays a selection of the spam keywords and their corresponding translations in English. In contrast, we obtained non-spam text from publicly verified Arabic accounts and pages.

Table 2: Sample of Arabic Spam Keywords with English Translation

Keyword	Translation
ريتويت	Retweet
شير	Share
اكسب	Win
كسبت	you won
ممكن تكسب	you can earn
شير	Share
اغتنم الفرصة	Seize the opportunity
لايك	Like
تابعونى	Follow me
منشن	Mention
مكافأة / جائزة مالية /نقدية	financial reward

To ensure that the tweets and posts were effectively processed and classified, the collected data comprised 20,240 tweets. However, before processing and classification, it was necessary to remove all white spaces and inconsistent strings that remained after deleting non-Arabic characters. Following this step, filtering was implemented to eliminate any duplicate and irrelevant content, as these types of tweets could have a negative impact on the accuracy of the dataset. Consequently, the final dataset comprised approximately 14,251 tweets that exclusively featured Arabic content, thereby ensuring that the dataset was unbiased and capable of producing accurate results. Table 3 displays a sample of the collected tweets alongside their corresponding English translations.

Table 3: A Sample of the collected tweets with English translation

Arabic Text	Translation
يلا الكل يعمل منشن عشان الكل يستفاد من العرض يلا فرصة ... بمناسبة العيد كل سنة وكل أصحابنا وحبايبنا طيبين	Come on, everyone mentions others so that everyone can benefit from the offer. It's an opportunity on the occasion of Eid, every year and our friends and loved ones are doing well.
الدفع المعنوي لا يوصف منشن لصاحبك 💙🫶 قولو شكرا لانك اجدع سند	The moral support cannot be described. Mention your friend 💙🫶 and say thank you for being the strongest support
شير على أوسع نطاق يا جماعه ومشكورين 🌺	Share to the widest extent, guys, and thank you 🌺
السحب الكتروني من المسابقة رتويت وتابع	The competition is a retweet and follow
التعليم: البابل شيت للثانوية تمت مراجعته بدقة قبل تصحيحه للتأكد من بيانات الطالب - اليوم السابع	Ministry of Education: The Babylon sheet for high school has been carefully reviewed before correcting it to ensure the student's data - Al-Yawm Al-Sabea
مجلس الوزراء ينفى تراجع الاهتمام الحكومي بمنظومة التعليم الفني	The Cabinet denies the government's lack of interest in the technical education system

Doi: https://doi.org/10.54216/FPA.160101
Received: July 21, 2023 Revised: November 19, 2023 Accepted: April 02, 2024

4.2. Dataset Annotation Phase

Our study introduces a novel annotation technique (TLM) that utilizes three distinct methods: Topic modeling, Lexicon approach, and Manual annotation. By incorporating the first two methods, we aimed to minimize the amount of manual effort required, while still ensuring the accuracy of the spam labels. Initially, we utilized LDA (Latent Dirichlet Allocation) to analyze the collected tweets and identify topics that are more likely to be associated with either spam or non-spam content. For instance, tweets that feature excessive promotional or advertising language may be more inclined towards being classified as spam.

To annotate our Arabic spam tweet dataset using the lexicon approach, we compiled a comprehensive list of words and phrases that are commonly associated with spam[1]. This list was compiled from a range of sources, including previous research studies and online resources. Once the lexicon was compiled, we applied it to our dataset using text analysis tools capable of detecting the presence of spam-related words and phrases in each tweet. If the tweet contained any of these words or phrases, we added the corresponding lexicon label as "spam". If none of the words or phrases were detected, we labeled the tweet as "non-spam". The final method employed in our study involved manually annotating tweets as either spam or non-spam. For this, we enlisted the help of three Arabic native speakers who were provided with clear guidelines and a set of pre-defined criteria to follow, which ensured consistency in their judgments. The criteria included specific keywords or phrases commonly associated with spam, excessive use of hashtags or links, repetitive or nonsensical content, the presence of suspicious links, misleading or false information, offensive or abusive language, and the use of automatic or robotic tweet-sending tools. These criteria served as a starting point for the manual annotation process and were refined or expanded as needed based on the specific context or goals of the task.

In conclusion, the combination of automated topic modeling using LDA, lexicon-based classification, and manual annotation resulted in an effective process for Arabic spam annotation. This process ensured high levels of accuracy and efficiency in spam annotation tasks. Upon completion of the annotation phase, we obtained an Arabic spam dataset containing 14,250 tweets. The dataset was divided into 6,770 non-spam tweets and 7,481 spam tweets. A sample of the annotated dataset can be found in Table 4, and its English translated version is available in Table 5.

Table 4 :A Sample of Annotated Arabic Spam Tweets

Tweet	Label
اغتنم الفرصة؛ شاليهات لاثرون يوفر شاليه لمدة ثلاث ايام العيد؛ للحجز ارجو الاتصال على ارقام ادارة	Spam
مسابقة سحب على مبلغ. 500. لعدد 5 فائزين. 1 2 3 4 5 الشروط. #متابعة. #رتويت. منشن شخصين #مسابقات.#بطاقة شحن.#سحب_على_مبلغ.#فعاليات.#فعاليه.#رمضان	Spam
مسابقة لفائزين بمبلغ مالي شروط سهله وبسيطة ريتويت للتغريده ضيف بندر @BNR__33	Spam
اكبر مشروع تكرير فى مصر وأفريقيا باستثمارات ٣.٧ مليار دولار والأضخم حجم بالشرق الأوسط	Non-Spam
ميكنة كافة الخدمات الحكومية وتقديمها الكترونيا للمواطنين عام 2025	Non-Spam
وصول بنسبة الصادرات مرتفعة المكون التكنولوجي من إجمالي الصادرات الصناعية المصرية إلى 6% في 2030	Non-Spam

Table 5 :English translation of Arabic Spam tweets

Tweet	Label
Seize the opportunity; Lathron Chalets provides a chalet for three days during Eid. To make a reservation, please call the management numbers.	Spam
Competition: draw for a prize of $500 for 5 winners. Conditions are easy and simple: follow, retweet, and mention two people. #Contests #Phone_card #Draw_for_a_amount #Events #Ramadan	Spam
Competition for winners of a cash prize. Easy and simple conditions: Retweet the tweet, and add Bandar @BNR__33.	Spam
The largest refining project in Egypt and Africa with investments of $3.7 billion and the largest size in the Middle East.	Non-Spam
Automating all government services and providing them electronically to citizens by 2025.	Non-Spam

[1] https://github.com/AhmedCS2015/Arabic-Spam/blob/main/Arabic%20Spam%20Lexicon.txt

Doi: https://doi.org/10.54216/FPA.160101
Received: July 21, 2023 Revised: November 19, 2023 Accepted: April 02, 2024

Increase the percentage of technological exports from the total Egyptian industrial exports to 6% by 2030."	Non-Spam

4.3. Data preprocessing phase

Text classification heavily relies on data preprocessing, which aims to both reduce the number of features in the dataset and improve the classifier's efficiency in terms of classification accuracy and resource usage. Arabic social media tweets often contain various forms of noise, including extra symbols, elongations, diacritical marks, repeated letters, or mixed language, which can negatively affect the accuracy of the classifier. Therefore, it is crucial to clean the text by removing these types of noise as part of the pre-processing phase which involves:

1. Eliminate stop words from each tweet, prepositions, articles, and conjunctions should be removed. Examples of such words include "في" "الى," "من," and "عن."
2. Text normalization involves transforming words into their formal written form. This includes replacing certain letters such as “أ”, “إ” and “آ” with “ا”, “ه” with “ة”, and “ى” with “ي”.
3. Another important step in pre-processing Arabic text is to remove diacritics, such as "اَلسَّلَام عَلَيْكُمْ" will be “السلام عليكم”.
4. Remove unnecessary repetition in words, repeated letters such as "ريتوويت" should be replaced with the correct single letter, such as "ريتويت".
5. Irrelevant noise, such as special characters (e.g., *, #, /, _, -), should also be removed.

4.4. Feature Extraction phase

In the context of text analysis, text data refers to combinations of words that exist in a dictionary or list. The process of feature extraction involves converting this text content into numerical features, which enables the creation of a consistent representation of documents in each dataset. Typically, the resulting features are organized into a matrix with M columns and N rows, where each column corresponds to a selected feature and each row corresponds to a text in the training set. The weight of each feature in a text is indicated by the value in the corresponding cell of the matrix. There are several methods for assigning weight to features, with the most used ones including:

- The Bag of Words (BoW) method is a way to represent text numerically by converting it into a vector of numbers that reflect the frequency of individual tokens in the text documents. This representation doesn't preserve the order of the words and disregards any syntactic structures that might be present [28][2]. e BoW representation of a text is a vector:

$$\overrightarrow{w} = [t_1, \dots, t_n] \qquad (1)$$

  Where n represents the size of the vocabulary, and t_i reflects the importance of that word in the text.

- The term frequency-inverse document format (TF-IDF) is a method for analyzing and quantifying the frequency of a word (also called a term) in a document, as well as its occurrence across multiple documents [28]. The term frequency is computed using a formula shown in the following equation:

$$TF = \frac{F(t,d)}{N} \qquad (2)$$

  where F(t, d) is the frequency of a term t in document d, and N is the total number of words in the document. The inverse document frequency (IDF) is applied to the vocabulary words that don't include stop words. The IDF formula is shown below:

$$IDF = \frac{\log(N)}{NT} \qquad (3)$$

  where N is the total number of documents and NT is the number of documents containing the word. Finally, the TF-IDF value is computed using the following equation, as shown in the following equation:

$$TFIDF = TF \times IDF \qquad (4)$$

- N-gram representation is a collection of n tokens that occur in a specific order in a text dataset. This representation can capture both syntactic and thematic information more accurately than other methods. For instance, if n = 2, a sequence of two-word pairs is created for each sentence [29].

Doi: https://doi.org/10.54216/FPA.160101
Received: July 21, 2023 Revised: November 19, 2023 Accepted: April 02, 2024

4.5. Spam classification phase

In our investigation, we utilized the most used machine learning algorithm with default settings and 10-fold cross-validation to prevent both overfitting and underfitting. Our primary objective was to enhance the performance of spam classification from two different perspectives. Firstly, we concentrated on algorithmic advancements, employing hyperparameter tuning to identify the optimal parameters that provide the highest accuracy. Secondly, we explored ways to improve the dataset by using a feature selection technique to choose the most pertinent features for the task at hand. We will provide more detailed information about these procedures in subsequent sections of the study. Our goal was to achieve the highest possible accuracy in spam classification by combining these two approaches. The classification process is summarized in Figure 2.

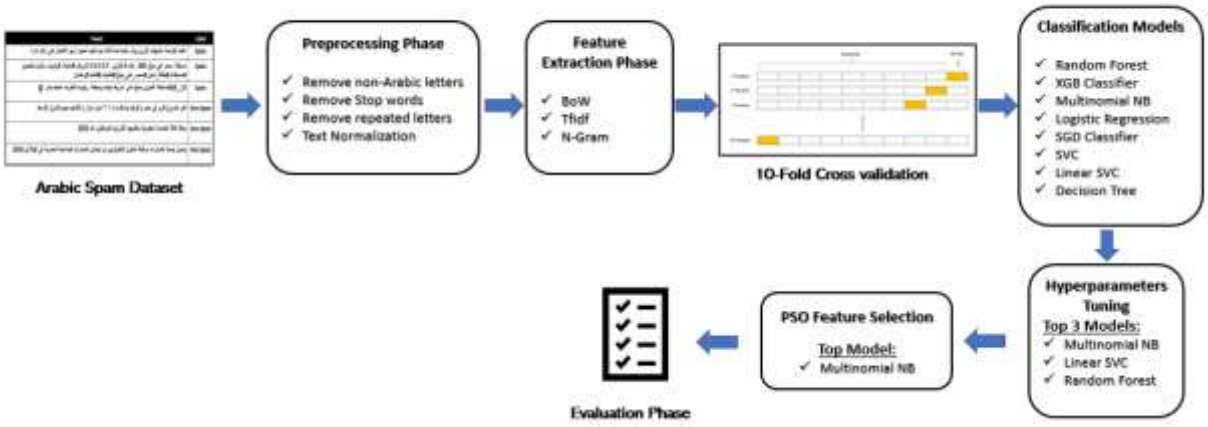

Figure 2: Arabic Spam Classification process

Figure 3 represents the pseudocode of the proposed Arabic Spam classification using a hyperparameters tuning and PSO algorithm.

1.1. Arabic Spam Classification

To determine the best classification algorithm and feature representation for classifying Arabic spam tweets, we applied the eight most common machine learning algorithms with their default parameters and 10-fold cross-validation. The algorithms used were: Random Forest, XGB Classifier, Multinomial NB, Logistic Regression, SGD Classifier, SVC, Linear SVC, and Decision Tree. We evaluated the algorithms using three different feature representations: Bag of Words (BoW), Term Frequency-Inverse Document Frequency (tf-idf), and N-gram with n=1,2 and 3.

To enhance the performance of our model, we choose the top three algorithms and their corresponding feature representation based on their performance. Then, we apply hyperparameter tuning to each of the selected algorithms to identify the optimal parameters that can fine-tune the model and yield better performance. Finally, the best-tuned algorithm is combined with Particle Swarm Optimization (PSO) feature selection to identify the most relevant features and optimize the classification performance.

```
# Step 1: Initialize
Initialize algorithms = ['Random Forest', 'XGB Classifier', 'Multinomial NB', 'Logistic Regression', 'SGD Classifier', 'SVC', 'Linear SVC', 'Decision Tree']
Initialize representations = ['BoW', 'tf-idf', 'N-gram n=1', 'N-gram n=2', 'N-gram n=3']
Initialize performance_scores = {}  # Dictionary to store algorithm and representation performance

# Step 2: Evaluate
For each algorithm in algorithms:
    For each representation in representations:
        scores = PerformCrossValidation(algorithm, representation)
        performance_scores[(algorithm, representation)] = CalculateMeanScore(scores)
```


Doi: https://doi.org/10.54216/FPA.160101
Received: July 21, 2023 Revised: November 19, 2023 Accepted: April 02, 2024

```
# Step 3: Select Top Three
top_three = SelectTopThree(performance_scores)

# Step 4: Hyperparameter Tuning
best_hyperparameters = {}
For pair in top_three:
   best_hyperparameters[pair] = TuneHyperparameters(pair)

# Step 5: PSO Feature Selection
best_tuned_pair = SelectBestTunedPair(best_hyperparameters)
optimized_features = ApplyPSOFeatureSelection(best_tuned_pair)

# Step 6: Final Model Evaluation
final_performance = EvaluateFinalModel(best_tuned_pair, optimized_features)

# Step 7: Conclusion
SelectBestModel(final_performance)
#  End of Process
```

Figure 3: Pseudocode for Determining the Best Classification Algorithm and Feature Representation

2. Experimental Results

2.1. Using Default Hyperparameters

Initially, we evaluated the accuracy of each machine learning model in classifying the Arabic reviews by utilizing the default hyperparameters. The default hyperparameters were set as per the specifications of the Python scikit-learn library package[30]. A comparison of the accuracy of the 8 algorithms with the different feature representations is shown in Table 6.

Table 6: Results of ML Algorithms on Arabic Spam Classification using Different Feature Representations

Algorithm	Bow	Tfidf	n-gram (1,2)	n-gram (1,3)
Random Forest	0.9603	0.9639	**0.9641**	**0.9667**
XGB Classifier	0.9307	0.9269	0.9279	0.9246
Multinomial NB	**0.9731**	**0.9716**	**0.9732**	**0.9727**
Logistic Regression	0.9620	0.9553	0.9450	0.9368
SGD Classifier	0.9610	0.9646	0.9600	0.9569
SVC	0.9314	0.9581	0.9473	0.9388
Linear SVC	**0.9643**	**0.9661**	0.9610	0.9579
Decision Tree	0.9209	0.9207	0.9219	0.9222

As shown in Table 6, Multinomial NB has the highest accuracy score across all four text preprocessing techniques. It performs consistently well across all the techniques and is particularly effective with n-gram (1,2) and n-gram (1,3). This suggests that the algorithm is good at capturing the relationship between adjacent words in the tweet. Random Forest has the second-highest accuracy score, and it performs well across all four techniques. Linear SVC has the third-highest accuracy score and performs well with Bow, Tfidf, and n-gram (1,2). However, its performance drops significantly with n-gram (1,3), which suggests that the algorithm may struggle to handle longer sequences of words.

Logistic Regression has a high accuracy score with Bow but performs poorly with other techniques. XGB Classifier, SGD Classifier, SVC, and Decision Tree all have relatively low accuracy scores across all four techniques. This suggests that these algorithms may not be well-suited for our Arabic spam classification tasks. The results suggest that Multinomial NB, Random Forest, and Linear SVC are the most effective algorithms for our Arabic spam classification tasks and that the choice of text preprocessing technique can have a significant impact on the performance of the algorithms.

Doi: https://doi.org/10.54216/FPA.160101
Received: July 21, 2023 Revised: November 19, 2023 Accepted: April 02, 2024

2.2. Using Hyperparameter Tuning Techniques

In this experiment, we implemented three techniques to fine-tune the hyperparameters of the best three algorithms from the previous experiment, by the accuracy calculation. The techniques are Grid Search, Random Search, and Genetic Algorithm Search.

For Multinomial NB, Table 7 displays three techniques for tuning hyperparameters, along with their accuracy scores, the best feature representation technique, and the optimal set of hyperparameters that yielded the best model configuration and highest accuracy for each technique.

Table 7:Performance Comparison of Hyperparameter Techniques with Multinomial NB

Hyperparameter techniques	Accuracy	Feature Representation	Optimal Hyperparameters
Grid Search	**0.9822**	N-Gram(1,2)	'alpha': 0.5, 'fit_prior': True
Random Search	0.9745	BoW	'alpha': 0.9, 'fit_prior': False, 'class_prior': None,
GA	0.9824	N-Gram(1,2)	'alpha': 0.3411897355372475, 'fit_prior': False

we can see that all three hyperparameter tuning techniques led to improved accuracy compared to the default parameters. Grid Search and GA achieved the highest accuracy, both at 0.982, while Random Search was slightly lower at 0.9745. It is also interesting to note that both Grid Search and GA achieved almost the same accuracy but with different optimal hyperparameters.

Similarly, Table 8 shows the results for the Random Forest algorithm:

Table 8: Performance Comparison of Hyperparameter Techniques with Random Forest

Hyperparameter techniques	Accuracy	Feature Representation	Optimal Hyperparameters
Grid Search	**0.9817**	N-Gram(1,2)	'bootstrap': False, 'criterion': 'entropy', 'max_features': 'sqrt'
Random Search	0.9717	N-Gram(1,2)	'bootstrap': False, 'criterion': 'entropy', 'max_features': 'log2'
GA	0.9756	BoW	'bootstrap': True, 'n_estimators': 112, 'max_features': 'sqrt',

From the table, we can see that the Grid Search technique resulted in the highest accuracy score of 0.9817 using N-Gram (1,2) feature representation. Random Search and GA techniques resulted in lower accuracy scores of 0.9717 and 0.9756, respectively.

The results for the third-best algorithm, Linear SVC, are shown in Table 9.

Table 9: Performance Comparison of Hyperparameter Techniques with Linear SVC

Hyperparameter techniques	Accuracy	Feature Representation	Optimal Hyperparameters
Grid Search	0.9750	N-Gram(1,2)	'C': 1, 'fit_intercept': False, 'loss': 'hinge', 'multi_class': 'ovr', 'random_state': None
Random Search	0.9709	tfidf	random_state': None, 'multi_class': 'ovr', 'loss': 'squared_hinge', 'fit_intercept': False, 'C': 2
GA	0.9735	tfidf	'C': 1.4471826366267675, 'fit_intercept': False, 'loss': 'hinge', 'random_state': 5

In this case, the default accuracy achieved using the default hyperparameters and feature representation was 0.961. However, after performing hyperparameter tuning using Grid Search, Random Search, and GA, the fine-tuned accuracies achieved were 0.975, 0.9709, and 0.9735, respectively. This represents a significant improvement in accuracy compared to the default accuracy. For example, Grid Search achieved an improvement of 1.4 percentage points, Random Search achieved an improvement of 0.99 percentage points, and GA achieved an improvement of 1.74 percentage points.

Doi: https://doi.org/10.54216/FPA.160101
Received: July 21, 2023 Revised: November 19, 2023 Accepted: April 02, 2024

Table 10, and Figure 4 summarize the performance of three algorithms using their default parameters, as well as after applying hyperparameter tuning (with the highest accuracy achieved through tuning techniques and the optimal feature representation over all experiments.

Table 10: Performance Comparison of top 3 Algorithms with Default and Fine-tuned Hyperparameter

Algorithm	Best Default Accuracy	Best Fine-tune Accuracy	optimal Feature representation
Multinomial NB	0.9732	**0.9822**	N-Gram(1,2)
Random Forest	0.9667	0.9817	N-Gram(1,2)
Linear SVC	0.9661	0.9750	N-Gram(1,2)

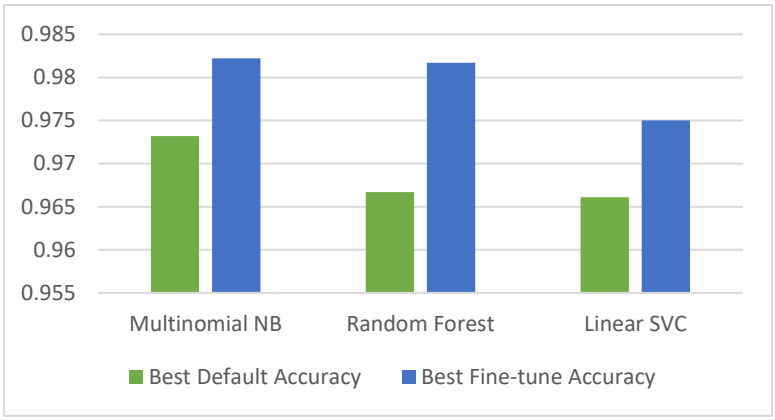

Figure 4: Performance Comparison of top 3 Algorithms with Default and Fine-tuned Hyperparameter

All three algorithms achieved high accuracy scores, with the best fine-tuning accuracy surpassing the best default accuracy for each algorithm. This indicates that applying hyperparameter tuning can improve the performance of the models. Furthermore, it is interesting to note that all three algorithms performed optimally using N-Gram (1,2) as the feature representation. N-Gram (1,2) refers to a combination of unigrams (single words) and bigrams (two consecutive words).

This suggests that this feature representation may be effective for our dataset and can be considered as a feature engineering technique for similar datasets in the future. Overall, Table 8 and Figure 4 provide valuable insights into the performance of these three algorithms on our Arabic spam dataset and highlight the importance of hyperparameter tuning and feature representation in improving model accuracy. Based on these results, it can be observed that Multinomial NB achieved the highest best fine-tuning accuracy compared to the other two algorithms. Therefore, we will use Multinomial NB for the next experiment.

2.3. Using PSO Feature selection

In the previous experiment, we optimized the Arabic spam classification by fine-tuning the algorithm and selecting the best feature representation. In this experiment, we aimed to optimize the classification from the dataset perspective by utilizing Particle Swarm Optimization (PSO) as a feature selection technique to identify the most relevant features. Following a series of experiments, we have determined that the optimal number of iterations and population size for our Arabic spam classification are both 100. Figure 4 illustrates the results for each iteration.

Doi: https://doi.org/10.54216/FPA.160101
Received: July 21, 2023 Revised: November 19, 2023 Accepted: April 02, 2024

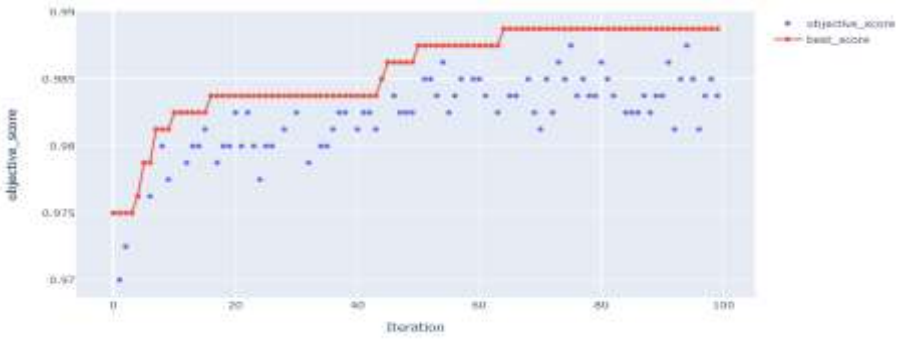

Figure 5: The result of PSO iterations

Based on the information presented in Figure 5, it appears that the highest accuracy achieved during the training iterations was 0.98875, which was obtained at iteration 74. This value is the highest among all other accuracy values obtained during the training process. Notably, this result was obtained using a subset of features containing only 38,465 features, representing only 50.15% of the full features set of 76,671 features. By using this smaller subset of features, the training time and storage requirements were reduced. It is worth noting that this approach enabled the accuracy to be improved from 0.9822 to 0.98875.

3. Limitations

1. **Limited Dataset Size**: The dataset used in this study consists of 14,250 tweets. While this is a reasonable size, it may not capture the full diversity of Arabic spam content. Future research could benefit from larger and more diverse datasets to improve model generalization.
2. **Annotation Subjectivity**: The proposed annotation technique combines topic modeling, lexicon approach, and manual annotation. However, manual annotation can be subjective, leading to potential bias or inconsistencies in the dataset. Efforts to minimize annotation subjectivity should be considered in future work.
3. **Algorithm Sensitivity**: Although the Multinomial NB algorithm performed well in this study, its performance may be sensitive to the specific dataset and feature representations. A more extensive exploration of algorithm sensitivity and robustness is needed.
4. **Feature Selection Impact**: While the PSO feature selection technique improved accuracy, its impact on model interpretability and feature importance is not discussed. Future research should address the trade-offs between accuracy and interpretability in feature selection.
5. **Resource Requirements**: While the PSO feature selection reduced training time and storage requirements, the specific resource savings and trade-offs are not quantified. A more detailed analysis of resource requirements and efficiency gains would provide valuable insights.
6. **External Validation**: The paper primarily focuses on internal model evaluation. External validation on independent datasets is essential to confirm the model's generalizability and real-world applicability.

The groundbreaking study provides valuable insights into optimizing Arabic spam classification. However, it is essential to acknowledge the limitations mentioned above, as they could impact the model's performance, generalization, and real-world applicability. Addressing these limitations will contribute to the development of more robust and reliable tools for improving Arabic spam classification.

4. Conclusion

This article presents a comprehensive approach for developing a high-quality Arabic spam classification model. The approach is composed of four primary phases, which are dataset building, data annotation, data preprocessing, and spam classification model construction. Each phase is described in detail, including the techniques and tools used. The dataset used in this study consists of 14,250 tweets, and the proposed annotation technique (TLM) incorporates three methods: topic modeling, lexicon approach, and manual annotation, and has been proven effective for Arabic spam annotation. Furthermore, the study assessed the performance of eight common machine learning algorithms using three different feature representations, followed by hyperparameter tuning and feature selection to improve the model's accuracy. The results indicate that the Multinomial NB algorithm performed consistently well across all text preprocessing techniques, and it was particularly effective with n-gram (1,2) and n-gram (1,3). The utilization of the PSO feature selection technique has reduced training time and storage requirements, resulting in an improvement of accuracy from 0.9822 to 0.98875. The proposed approach and dataset are expected to be useful for researchers and practitioners who work on Arabic spam classification.

Doi: https://doi.org/10.54216/FPA.160101
Received: July 21, 2023 Revised: November 19, 2023 Accepted: April 02, 2024

Declaration of generative AI and AI-assisted technologies in the writing process

During the preparation of this work the author(s) used ChatGPT in order to improve language and readability. After using this tool/service, the author(s) reviewed and edited the content as needed and take(s) full responsibility for the content of the publication.

Funding: This work was supported by the Deanship of Scientific Research, Vice President for Graduate Studies and Scientific Research, King Faisal University, Saudi Arabia [Project No.: GRANT2,730]

Institutional Review Board Statement: Not applicable.
Informed Consent Statement: "This article does not contain any studies with human participants or animals performed by any of the authors."

Data Availability Statement: The dataset used in this study is public and all test data are available at this portal (https://github.com/AhmedCS2015/Arabic-Spam)

Acknowledgments: This work was supported by the Deanship of Scientific Research, Vice Presidency for Graduate Studies and Scientific Research, King Faisal University, Saudi Arabia [Project No.: GRANT2,730]

Conflicts of Interest: The authors declare no conflict of interest.

References

[1] Statista, "Number of global social network users 2017-2027," 2023. https://www.statista.com/statistics/278414/number-of-worldwide-social-network-users/ (accessed Feb. 01, 2023).

[2] A. Omar, T. M. Mahmoud, T. Abd-El-Hafeez, and A. Mahfouz, "Multi-label Arabic text classification in Online Social Networks," *Inf. Syst.*, vol. 100, p. 101785, 2021, doi: 10.1016/j.is.2021.101785.

[3] P. V. Bindu, R. Mishra, and P. S. Thilagam, "Discovering spammer communities in twitter," *J. Intell. Inf. Syst.*, vol. 51, no. 3, pp. 503–527, 2018, doi: 10.1007/s10844-017-0494-z.

[4] F. Alqahtani, "Optimizing Spam Detection in Twitter by Using Naïve Bayes, Logistic Regression and Stochastic Gradient Descent with Whale Optimization Algorithm and Genetic Algorithm," *J. Xi'an Univ. Archit. Technol.*, vol. XII, no. III, pp. 2742–2747, 2020, doi: 10.37896/jxat12.03/225.

[5] M. Westerlund, "The emergence of deepfake technology: A review," *Technol. Innov. Manag. Rev.*, vol. 9, no. 11, pp. 39–52, 2019, doi: 10.22215/TIMREVIEW/1282.

[6] J. P. Carpenter, "Spam and Educators ' Twitter Use : Methodological Challenges and Considerations," pp. 460–469, 2020.

[7] A. Omar, T. M. Mahmoud, and T. Abd-El-Hafeez, *Building Online Social Network Dataset for Arabic Text Classification*, vol. 723. 2018. doi: 10.1007/978-3-319-74690-6_48.

[8] A. Omar and A. E. Hassanien, "An Optimized Arabic Sarcasm Detection in Tweets using Artificial Neural Networks," *5th Int. Conf. Comput. Informatics, ICCI 2022*, no. March 2022, pp. 251–256, 2022, doi: 10.1109/ICCI54321.2022.9756102.

[9] A. Omar and T. M. Mahmoud, *Comparative Performance of Machine Learning and Deep Learning Algorithms for Arabic Hate Speech Detection in OSNs*, vol. 1. Springer International Publishing, 2020. doi: 10.1007/978-3-030-44289-7.

[10] X. Deng, Y. Li, J. Weng, and J. Zhang, "Feature selection for text classification: A review," *Multimed. Tools Appl.*, vol. 78, no. 3, pp. 3797–3816, 2019, doi: 10.1007/s11042-018-6083-5.

[11] M. Mataoui, O. Zelmati, D. Boughaci, M. Chaouche, and F. Lagoug, "A proposed spam detection approach for Arabic social networks content," *Proc. 2017 Int. Conf. Math. Inf. Technol. ICMIT 2017*, vol. 2018-Janua, pp. 222–226, 2017, doi: 10.1109/MATHIT.2017.8259721.

[12] S. Al-Azani and E. S. M. El-Alfy, "Detection of Arabic spam tweets using word embedding and machine learning," *2018 Int. Conf. Innov. Intell. Informatics, Comput. Technol. 3ICT 2018*, 2018, doi: 10.1109/3ICT.2018.8855747.

Doi: https://doi.org/10.54216/FPA.160101
Received: July 21, 2023 Revised: November 19, 2023 Accepted: April 02, 2024

[13] H. Almerekhi and T. Elsayed, "Detecting Automatically-Generated Arabic Tweets," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 9460, 2015, pp. 123–134. doi: 10.1007/978-3-319-28940-3_10.

[14] A. Ghourabi, M. A. Mahmood, and Q. M. Alzubi, "A hybrid CNN-LSTM model for SMS spam detection in arabic and english messages," *Futur. Internet*, vol. 12, no. 9, pp. 1–16, 2020, doi: 10.3390/FI12090156.

[15] H. Mubarak, A. Abdelali, S. Hassan, and K. Darwish, *Spam Detection on Arabic Twitter*, vol. 12467 LNCS. Springer International Publishing, 2020. doi: 10.1007/978-3-030-60975-7_18.

[16] A. Ziani *et al.*, "Deceptive Opinions Detection Using New Proposed Arabic Semantic Features," *Procedia CIRP*, vol. 189, pp. 29–36, 2021, doi: 10.1016/j.procs.2021.05.067.

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

[18] A. M. Al-Zoubi, J. Alqatawna, H. Faris, and M. A. Hassonah, "Spam profiles detection on social networks using computational intelligence methods: The effect of the lingual context," *J. Inf. Sci.*, vol. 47, no. 1, pp. 58–81, 2021, doi: 10.1177/0165551519861599.

[19] A. M. Alkadri, A. Elkorany, and C. Ahmed, "Enhancing Detection of Arabic Social Spam Using Data Augmentation and Machine Learning," *Appl. Sci.*, vol. 12, no. 22, 2022, doi: 10.3390/app122211388.

[20] H. Najadat, M. A. Alzubaidi, and I. Qarqaz, "Detecting Arabic Spam Reviews in Social Networks Based on Classification Algorithms," *ACM Trans. Asian Low-Resource Lang. Inf. Process.*, vol. 21, no. 1, pp. 1–13, 2022, doi: 10.1145/3476115.

[21] T. Yu and H. Zhu, "Hyper-Parameter Optimization : A Review of Algorithms," *arXiv Prepr. arXiv2003.05689*, pp. 1–56, 2020.

[22] T. Akiba, S. Sano, T. Yanase, T. Ohta, M. Koyama, and P. Networks, "Optuna : A Next - generation H yperparameter Optimization Framework," pp. 1–10, 2019.

[23] E. Elgeldawi, A. Sayed, A. R. Galal, and A. M. Zaki, "Hyperparameter tuning for machine learning algorithms used for arabic sentiment analysis," *Informatics*, vol. 8, no. 4, pp. 1–21, 2021, doi: 10.3390/informatics8040079.

[24] S. Nematzadeh, F. Kiani, M. Torkamanian-afshar, and N. Aydin, "Tuning hyperparameters of machine learning algorithms and deep neural networks using metaheuristics : A bioinformatics study on biomedical and biological cases," *Comput. Biol. Chem.*, vol. 97, no. December 2021, p. 107619, 2022, doi: 10.1016/j.compbiolchem.2021.107619.

[25] B. G. Marcot and A. M. Hanea, "What is an optimal value of k in k-fold cross-validation in discrete Bayesian network analysis?," *Comput. Stat.*, no. 0123456789, 2020, doi: 10.1007/s00180-020-00999-9.

[26] A. P. Piotrowski, J. J. Napiorkowski, and A. E. Piotrowska, "Population size in Particle Swarm Optimization," *Swarm Evol. Comput.*, vol. 58, no. March 2019, p. 100718, 2020, doi: 10.1016/j.swevo.2020.100718.

[27] V. Govindaraju, I. Nwogu, and S. Setlur, "Chapter 1 - Document Informatics for Scientific Learning and Accelerated Discovery," in *Big Data Analytics*, vol. 33, V. Govindaraju, V. V Raghavan, and C. R. Rao, Eds. Elsevier, 2015, pp. 3–28. doi: https://doi.org/10.1016/B978-0-444-63492-4.00001-0.

[28] N. Orangi-Fard, A. Akhbardeh, and H. Sagreiya, "Predictive Model for ICU Readmission Based on Discharge Summaries Using Machine Learning and Natural Language Processing," *Informatics*, vol. 9, no. 1, 2022, doi: 10.3390/informatics9010010.

[29] J. Awwalu, A. A. Bakar, and M. R. Yaakub, "Hybrid N-gram model using Naïve Bayes for classification of political sentiments on Twitter," *Neural Comput. Appl.*, vol. 31, no. 12, pp. 9207–9220, 2019, doi: 10.1007/s00521-019-04248-z.

[30] Scikit_Learn, "Machine Learning in Python," 2022. https://scikit-learn.org/stable/ (accessed Mar. 01, 2023).

Doi: https://doi.org/10.54216/FPA.160101
Received: July 21, 2023 Revised: November 19, 2023 Accepted: April 02, 2024