IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 25 January 2023, accepted 23 February 2023, date of publication 28 February 2023, date of current version 3 March 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3250256

APPLIED RESEARCH

Intelligent Framework for Detecting Predatory Publishing Venues

WED MAJED BIN ATEEQ[1,2] **AND HEND S. AL-KHALIFA**[2]
[1]Department of Information Technology, Imam Mohammad Ibn Saud Islamic University, Riyadh 11432, Saudi Arabia
[2]Department of Information Technology, King Saud University, Riyadh 11362, Saudi Arabia

Corresponding author: Wed Majed Bin Ateeq (wedateeq@outlook.com)

This work was supported by the NAMA Women Advancement Establishment as the Strategic Sponsor of the Third Forum for Women in Research (QUWA): Women Empowerment for Global Impact with the University of Sharjah.

ABSTRACT Predatory publishing venues publish questionable articles and pose a global threat to the integrity and quality of the scientific literature. They have given rise to the dark side of scholarly publishing and their effects have reached political, societal, economic, and health aspects. Given their consequences and proliferation, several solutions have been developed to help detect them; however, these solutions are manual and time-consuming. While researchers, students, and readers are in need of a tool that automatically detects predatory venues and their violations, in this study, we proposed an intelligent framework that can automatically detect predatory venues and their violations using different artificial intelligence techniques. This work contributes through the following: (1) creating a dataset of 9,866 journals annotated as predatory and legitimate, and (2) proposing an intelligent framework for classifying a venue as legitimate or predatory, with appropriate reasoning. Our framework was evaluated using seven different machine learning and deep learning models, including Support Vector Machine (SVM), K-Nearest Neighbors (KNN), Neural Networks (NNs), Long short-term memory (LSTM), Convolutional Neural Network (CNN), Bidirectional Encoders from Transformers (BERT), A Lite BERT (ALBERT), and different feature representation techniques. The results showed that the CNN model outperformed the other models in journal classification task, with an F1 score of 0.96. For appropriate reasoning of the provisioning task, the SVM model achieved the best micro F1 of 0.67.

INDEX TERMS Automatic detection, deceptive publishing, fake website detection, deep learning, machine learning, predatory venues, pseudo-journals, predatory journal, scholarly publishing, website classification.

I. INTRODUCTION

Science is cumulative in its nature. As scientists engaged in publishing their research, we progressively advance our scientific knowledge. Scholarly publishing affects different aspects, including political, societal, economic, and most importantly, health. Thus, we cannot overestimate the risks to the integrity, quality, evidence-based practices, and academic standards of scientific research [1].

Predatory venues comprise one of the risks that affect scholarly publishing, as Jeffery Beall, the first scholar who worked on predatory publishing, said, ''By far, predatory publishers damage science more than anything else. They do not faithfully manage peer review, allowing questionable science to be published as if it had passed a strong peer review'' [2].

Predatory publishing venues are journals, conferences, or publishers that publish scholarly content of questionable quality for profit without transparency in their policies and operating procedures, as expected from legitimate peer-reviewed venues. They deceive their stakeholders such as authors, readers, funders, or even their recruited editorial board members. It is estimated that predatory venues' publishing increased from 53,000 articles in 2010 to approximately 420,000 articles in 2014 [3]. In addition, based on a recent study published by the InterAcademy Partnership in 2022, it was found that predatory practices impact at least one million researchers and cost billions spent on wasted research [4]. Furthermore, multiple predatory journals in biomedical fields have been identified in the well-known

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

database PubMed [5]. In addition, several medical companies have engaged with predatory publishers, which put patient safety into question [6].

Several attempts have been made to address this problem of predatory venues. Jeffery Beall was the first to publish a number of criteria that characterize predatory venues [7], [8], and a blacklist of questionable venues [9]. Beall's attempts followed by establishing a number of frameworks composed of a set of criteria describing the venues under investigation, some of which are turned into a metric or mechanism, such as Cabell's framework [10] and the framework of ''Principles of Transparency & Best Practice for Scholarly Publications'' [11]. Moreover, different white- and blacklists of legitimate and predatory venues are established e.g. [12], [13], and [14]. However, most of the presented frameworks manually detect predatory venues and have many false positives and negatives [6], [15], [16], [17], [18]. Moreover, they did not provide proper reasoning about why they have identified a publishing venue as predatory. In addition, white- or blacklists need to be updated as new journals appear, and the journals in the list can change their state from predatory to legitimate with time. The number of publishing venues is often too high, and new publishing venues continue to appear every day, so it is difficult to track all of them or have in-depth knowledge of them. Besides these works, there is a study published by Adnan et al. [19] in 2018 aimed to automatically detect predatory journals by utilizing traditional machine learning classifiers. This outlines the value of establishing an intelligent framework that would detect predatory venues and illustrate the violated criteria. Hence, the main problem that we aim to address in this research is that: *given a website of a journal, the goal is to predict automatically whether that website is a predatory venue or not, with appropriate reasoning*.

Therefore, in this paper, we propose an intelligent framework for the auto-detection of predatory venues with appropriate reasoning about violations using different artificial intelligence techniques. The proposed framework tackled the predatory venue detection problem as a website binary classification problem and addressed the problem of providing appropriate reasoning as a website multilabel classification problem.

Therefore, our project aims to answer the following questions:

1) To what extent can the proposed framework use a **machine learning** approach to detect predatory venues and provide appropriate reasoning about violations?
2) To what extent can the proposed framework use a **deep learning** approach to detect predatory venues and provide appropriate reasoning about violations?
3) What is the effectiveness of the proposed framework using machine learning compared to deep learning?

To answer these questions, we constructed a dataset that contained 6,836 journals annotated as predatory or questionable and 1,894 journals annotated by 39 criteria (whether the journal applied the criteria or not). The dataset was constructed in collaboration with freelancers who were provided with training and tests. Additionally, different quality control methods were used to ensure the quality of the annotated data. Then, we experimented with a number of case studies based on the proposed framework, in which we evaluated and compared different machine learning and deep learning approaches along with different feature representation techniques, including different architectures for Support Vector Machine (SVM), K-Nearest Neighbors (KNN), Neural Networks (NNs), Long short-term memory (LSTM), and Convolutional Neural Network (CNN). In addition, we evaluated the recent pre-trained Bidirectional Encoders from Transformers (BERT) and A Lite BERT (ALBERT). As a feature representation technique, we evaluated and compared Term Frequency–Inverse Document Frequency (TF-IDF), information gain, Word2Vec, and Doc2Vec. We evaluated and reported the effectiveness of each case study using a set of evaluation measures, namely precision, recall, F1 score, micro F1, and weighted F1.

Thus, our main objectives are summarized in the following points:

1) Propose an intelligent framework for the auto-detection of predatory venues and provide appropriate reasoning.
2) Build an annotated dataset for predatory publishing venues.
3) Evaluate the effectiveness of the proposed framework.

The rest of this paper is organized as follows. Section II provides a review of the literature on the domain of predatory venues, available approaches for detecting predatory venues, and website classifications. Section III describes the used research methodology. Section IV details the process of constructing our dataset, including collecting journals, compiling legitimacy criteria, and annotating the collected dataset. Section V presents the experimental setup and section VI discusses the results. Finally, the paper concludes with limitations and future work.

II. LITERATURE REVIEW

Predatory venues have been an active research area since Beall published a list of predatory journals and publishers in 2010. To understand the works that tackle the predatory venue problem, we reviewed the predatory venue literature in the first section and present the available approaches for detecting predatory venues. However, the work on automatic detection is relatively limited in this area; thus, we tackled the problem as a website classification problem. In the domain of website classification, we focused on research on fake website detection because they aim to classify a given website as fake or legitimate; hence, their approaches are in line with our goal.

A. PREDATORY VENUES

''Predatory publishers/publishing'' was first coined by Jeffrey Beall in 2010. According to Beall, predatory publishers

took advantage of opportunities that came with the scholarly Open Access (OA) movement. He said that prior to 1998, when scholarly journals were print-based subscription journals, they were of good quality, and peer review was managed seriously. However, there were few low-quality publishers, and most researchers were able to avoid them [20]. The OA movement is based on providing readers access to peer-reviewed articles free of charge and with few restrictions on usage, while authors pay for Article Processing Charges (APC) to publishing venues [21]. Besides Beall, multiple scholars believe that the OA movement is exploited by predatory venues and has given rise to the dark side of scholarly publishing [20]. However, OA publishing is not a bad model, and there are many legitimate OA venues [20].

In an attempt to clearly understand the general concept of predatory venues, it is essential to understand what, in fact, is meant by ''predatory'' in the scholarly publishing field. From this perspective, until now, it is not clear what is meant by ''predatory,'' and there is no universal agreement about the characteristics of predatory venues [6]. However, there is general agreement among some scholars to define predatory venues as venues that claim to conduct proper peer review, while in fact, they do not [6]. Different scholars have tied different characteristics to this notion, such as spamming researchers by sending emails to publish, pretending to have sufficient quality control, poor editorial services and poor copyediting, and charging researchers with excessive and non-disclosed publication fees [22].

Beall defined predatory publishers as venues suspected of unethical practices on three levels: business, research, and publishing [20]. After Beall's movement, scholars started to define and characterize predatory venues as a step toward stopping predatory publishing. For example, in 2017, Polit and Beck explained predatory journals as ''[those who] charge publication fees without conducting adequate reviews and editorial services'' [23]. In 2018, Reves et al. defined predatory publishing as ''an opportunistic exploitation of inexperienced researchers pressured to publish or perish'' [24]. Therefore, some researchers refer to predatory publishing as ''pay-to-publish.'' From this point of view, Strong defined it as journals that publish below minimal standards of quality by using an exploitative business model, deceptive publishing practices, and substandard quality control measures for making a profit [25]. As we can see, the majority of scholars agreed that predatory venues seek financial profits in the form of APC from their authors and subscription fees from their readers for their owners and editors [6], [26].

In the context of scholarly publishing, the validity of the term ''predatory'' has been questioned among scholars. Either because it is not adequately clear what it means [27] or because they want to distinguish between being a predatory venue and using predatory practices [28]. Additionally, another concept is included under this term by some researchers, which is hijacked venues, which mimic the website of a legitimate venue, its name, and ISSN, or making little unnoticeable changes [29]. Other descriptors include ''dubious publishing,'' ''deceptive publishing,'' and ''pseudo-journals'' [21].

The opposite of ''predatory'' venues are ''legitimate'' venues, and there is no universal agreement on what it means. Different organizations have their own strict standards, practices, or criteria that describe the legitimacy of a venue, such as the Directory of Open Access Journals (DOAJ), Committee on Publication Ethics (COPE), World Association of Medical Editors (WAME), and Open Access Scholarly Publishers Association (OASPA) [11]. These companies collaborated and defined shared criteria called ''Principles of Transparency & Best Practice for Scholarly Publications'' as criteria describing legitimate venues [11], for short ''Principles of Transparency.''

Based on the aforementioned points of views, we define predatory venues as the following: predatory publishing venues are journals, conferences, or publishers that publish scholarly content of questionable quality for profit without being transparent in their policies and operating procedures as expected from legitimate peer-reviewed venues, which could deceive their stakeholders, such as authors, readers, funders, or even their recruited editorial board members. Therefore, our definition, precisely the term ''transparent,'' could include hijacked, deceptive, and low-quality venues. However, it does not include the vanity press, where the authors only need to pay to publish their work.

B. DETECTION APPROACHES OF ONLINE PREDATORY VENUES

In an attempt to help scholars identify predatory venues among legitimate ones, Jeffrey Beall published the first blacklist of deceptive OA publishers in 2010 [9]. Since then, this list has been expanded to include more publishers and questionable journals [9]. After he published the list, he was exposed to continued harassment and threats; therefore, he took down his list of predatory journals in 2017 [20]. However, the list has been republished by anonymous scholars and information professionals, and 152 journals have been added to it; it now contains 1462 journals [8]. This story is followed by different movements from different scholars to develop a set of criteria or lists, either blacklists or whitelists, to detect predatory venues. Most notably, there is diversity in the number of criteria as well as their content [30]; however, the lists are criticized because of the overlap between the lists [15], [16], [17], [18]. detecting predatory venues.

Table 1 shows a summary of some of the existing lists.

Besides blacklists and whitelists, scholars' first attempts to tackle predatory venues' problems by trying to establish frameworks that could help other researchers and funders in detecting predatory venues.

Thus, the manual approaches can be divided into two main categories: (1) frameworks and (2) lists. Most frameworks rely on developing a set of criteria (a set of principles) [8], [31], and [21], some of which are converted into metrics (measurements) [32], [33] or mechanisms

TABLE 1. Lists for detecting predatory venues.

Title	Category (number of items)	Approach Type
Beall's [8]	predatory OA journals (1,462)	blacklist
Cabell's [10] [31]	legitimate journals (11,000)	whitelist
	predatory journals (6,800)	blacklist
COPE [13]	legitimate journals (12,640)	whitelist
	legitimate publishers (73)	whitelist
DOAJ [14]	legitimate OA journals (14,340)	whitelist
Journal Publishing Practices and Standards (JPPS) for developing countries **[37]**	legitimate and predatory journals (988)	whitelist and blacklist
OASPA **[38]**	legitimate OA publishers (150)	whitelist
Stop Predatory Journals **[39]**	predatory journals (1,317)	blacklist
	predatory publishers (1,176)	blacklist

TABLE 2. Summary of literature work in the predatory venues field.

Authors	Year	Aim	Results
Beall's [7]	2015	Detect predatory journals	Developing a manual framework of a set of predator criteria contains 52 criteria
Cabell's [10] [31]	2019	Detect predatory journals	Developing a manual framework of a set of predator criteria contains 74 criteria
Principles of Transparency [11]	2014	Detect legitimate journals	Developing a manual framework of a set of legitimacy criteria contains 16 principles
Laine and Winker [34]	2017	Detect predatory journals and provide an error-proof for the violations	Developing a manual mechanism based on Beall, DOAJ, and "Think. Check. Submit" Initiative.
Xuelian et al. **[21]**	2018	Detect predatory journals in the context of nursing and midwifery	Developing a manual mechanism based on 28 reviewed papers.
Dadkhah et al. **[32]**	2016	Rank questionable venues	Developing a manual metric called predatory rate metric
Lang et al. **[33]**	2020	Confront predatory conferences	Developing a manual metric called Conference Impact Factor (CIF)
Dadkhah et al. [5], and Gutierrez et al. [5].	2018	Check the available metrics	34 fake impact factors are found by Dadkhan, and 20 misleading impact factors are found by Gutierrez
Beall's list [8]	2020	Check the available metrics	55 fake impact factors are found
Shahri et al. in **[35]**	2016	Automatic detection of hijacked venues	Proposing a traditional machine learning approach for auto-detect of hijacked journals
Dadkhah et al. in **[36]**	2016	Automatic detection of hijacked venues	Proposing a traditional machine learning approach for auto-detect of hijacked journals
Adnan et al. **[19]**	2018	automatic detection of journals	Proposing a traditional machine learning approach for auto-detect of predatory journals

(algorithms) [34]. However, these frameworks have been criticized because they have been established with little information on development, validation, and reliability [30]. Therefore, some studies have analyzed previously proposed frameworks and checklists [6], [15], [16], [17], [18]. In addition to these manual approaches, some scholars have attempted to solve the problem automatically [35], [36], and [19]. Table 2 summarizes these studies.

Besides Beall's black list, he also published a set of criteria consisting of 52 items [7], [8], where he thought that these criteria form a framework for analyzing OA venues.

The published criteria were divided into five categories: integrity, editor and staff, business management, poor journal practices, and others. Beall's blacklist caused a sensation among scholars and librarians.

As the number of predatory venues continued to increase, a large number of publishing venues with different qualifications tried to register in the well-known indices of scholarly publishing as a way to prove their legitimacy. Four well-known organizations in scholarly publishing have collaborated to identify the criteria that characterize legitimate journals: DOAJ, WAME, COPE, and OASPA. They define the Principles of Transparency as follows: The Principles of Transparency contain 16 principles, and they address different topics, such as the peer review process, publication ethics, and archiving. They form the basis of the criteria used to assess the suitability for membership by DOAJ, OASPA, and COPE and part of the criteria used to evaluate membership requests by WAME. Each organization has additional criteria for assessing membership requests [11].

Cabell's International was established in 1970 by professor David Cabell to help scholars decide where to publish [10]. It provides a searchable database of publishing directories describing different information about venues, such as titles, ISSNs, quality metrics, and fees [10]. It consists of a whitelist of 11,000 journals spanning 18 disciplines [12]. In 2017, after Beall took down his list, the company added a blacklist to its database, which contains more than 6,800 journals [12]. Both OA and non-OA journals were included in the lists. The lists were created based on more than 60 criteria [31], and the violated criteria for each blacklisted journal were presented to the users [12]. The criteria were grouped into eight categories: integrity, peer review, website, publication practices, indexing and metrics, fees, access and copyright, and business practices. However, specialists manually analyze the journals to include them in the lists [10], which are available for institutional licensing only [12].

To develop a mechanism for identifying predatory journals, Laine and Winker [34] reviewed the criteria of Beall, DOAJ, and ''Think. Check. Submit'' Initiative. They developed their algorithm because the available initiatives do not provide error-proof methods for judged journals. Their mechanism

depends on all the reviewed criteria, and the researcher must manually judge based on the number of violations. With the aim of discussing predatory open-access publishing in the context of nursing and midwifery, Xuelian et al. reviewed 28 papers in 2018 [21]. They then developed a framework based on the reviewed papers composed of a set of guidelines. The authors summarized important considerations for nursing researchers. Some of their guidelines is checking the integrity of OA journals and publishers in the DOAJ directory and OASPA, respectively. Other criteria included the content of the venue website, such as the venue title and publication history.

Apart from criteria and lists, some scholars use popular metrics such as Thompson-Reuters' Journal Citation Reports (JCR) to check the legitimacy of venues. According to Beall, the most reliable way to verify the impact factor of a venue is to check JCR. Other portals offer complete metrics of peer-reviewed articles, such as Scopus's Source Normalized Impact per Paper (SNIP) and CiteScore or SCImago Journal and Country Rank (SJR) [21].

Some scholars believe that it is not fair to classify a venue as legitimate or predatory because there are newly established venues that do not have time to reach good quality. They think that the term ''predatory'' merges between deceptive and low-quality venues. In this context, Dadkhah et al. [32] introduced the predatory rate metric to rank questionable venues according to 14 criteria selected from Beall's criteria. In their work, venues could be predatory, applying predatory practices, or legitimate. The predatory rate metric is calculated as the weighted sum of the number of applied criteria. The authors used a case study to demonstrate the applicability of their metric. However, they did not clarify what bases they chose for the criteria. Moreover, this metric was criticized by [6] because of mixing criteria that can provide an indication of quality and criteria that are sufficient by themselves to conclude that the journal is deceptive. Another study was conducted by Lang et al. [33] to confront predatory conferences. They proposed a Conference Impact Factor (CIF) metric to evaluate the effectiveness of scientific conferences and ranked them based on their CIF. CIF is calculated based on the number of papers published in peer-reviewed journals and the journals' impact factors. They demonstrated the applicability of the CIF metric using a case study.

Besides the metrics that can help to judge the impact factor of journals, many misleading metrics were established, examples of fake impact factors are Global Impact Factor (GIF), Citefactor, and Universal Impact Factor (UIF), and there is even a fake Thomson Reuters Company [40]. Dadkhah et al. found 34 fake impact factors after they investigated 300 predatory journals, whereas Gutierrez et al. found 20 misleading impact factors [5]. Furthermore, 55 fake impact factors were available in the last updated version (2019) of Beall's list [8].

Several studies were conducted to analyze the available frameworks, and it was found that the frameworks contained overlap and false positives and negatives. In false positives, predatory journals are classified as legitimate journals, whereas false negatives, legitimate journals are classified as predatory journals. Beall's list was examined by Olivarez et al. [1] by employing three-person independent judgment. Because they think that Beall favors non-OA journals over OA journals, they examined seven OA and 80 non-OA journals using Beall's criteria. These journals were (at the time of the study) a collection in the InCites Journal Citations Reports (JCR) section from Thomson Reuters' Web of Science database and are well known in the field of information science and library science, where the median age was 32 years. They found, whether OA or non-OA, that these well-regarded academic journals could be considered predatory journals based on Beall's list. Therefore, the authors concluded the subjective nature of Beall's criteria and the bias against OA journals.

To validate the lists work by Teixeira da Silva et al. [16] where they used epidemiological measures including likelihood ratio, sensitivity, specificity, and prevalence rate, to assess the reliability of the inclusion criteria of Beall, Cabell, and Crawford. They found that Beall's list had a high positive rate, while Cabell's blacklist and Crawford's gray OA list had fewer false-positive rates.

The aforementioned studies addressed the frameworks for the manual detection of predatory venues and their issues. However, only three studies were found on automatic detection, where two focused on the auto-detection of hijacked venues, and only one study was found on the automatic detection of predatory venues.

For the automatic detection of hijacked venues, Shahri et al. [35] explored using different decision tree algorithms. They trained these algorithms on 104 journals' websites, where 59 were authentic, and 45 were hijacked. Among the algorithms used, the Random Tree was the best, with a 0% error rate; hence, its resulting tree was used for the evaluation. They used ten journals' websites to evaluate the resulting model, which had a 10% error rate. In their study, nine features were used for detecting hijacked journals, which are domain rank in the search engine, age of the domain, entering countries in the journal website, aim and scope, number of broken links, number of published articles in a year, consistency between the country of the server and the country of the journal, number of dead links, and the use of the character ''-'' in the URL. In the same context, Dadkhah et al. [36] experimented with different decision-tree algorithms to detect hijacked journals. In their experiment, they used 28 authenticated and 56 hijacked journals. They then extracted 12 features manually for detecting hijacked journals and found that the Random Tree was the best with a 2.53% error rate. The resulting tree comprised five features: availability of full text, authors' country, rank in search results, domain lifetime, and availability of previous issues. They evaluated the resulting tree on 15 authentic journals and 10 hijacked journals, and it had a 12% error rate.

The study that addressed the automatic detection of predatory venues was done by Adnan et al. [19]. They presented a methodology for the autodetection of predatory journals. The problem was defined as a classification problem, and three traditional machine-learning approaches were used: Support Vector Machine (SVM), K-Nearest Neighbors (KNN), and Naïve Bays (NB). They collected 100 legitimate and 100 predatory journals from the DOAJ directory and Beall list. Two different feature representations were used, heuristics-based and text-based, to investigate their effectiveness in the classification process. The heuristics-based features were the criteria that could be automated from Beall's criteria and were selected manually. The text-based features refer to the textual content extracted from the journals' websites, and the authors used TF-IDF to weight the text-based features while performing different pre-processing steps. Subsequently, for the two feature representations, the authors performed several experiments using information gain as a feature selection method to specify the best feature set size for each classifier. A 10-fold cross-validation was used to validate the results. The SVM classifier outperformed the other classifiers, achieving 98% (F1-score) using heuristic features and 96% (F1-score) using text-based features. However, the work was based on Beall's list and criteria, where Beall's work has been criticized, as stated earlier, for being subjective in nature and having many mistakes in his list.

C. WEBSITES CLASSIFICATION

This section reviews machine learning approaches for binary and multilabel website classification, including traditional machine learning and deep learning approaches, and focuses on automatic fake website detection.

1) TRADITIONAL MACHINE-LEARNING APPROACHES

This section addresses fake website detection using traditional machine learning approaches, which have been used in most studies of web page classification, as stated by Hashemi's survey published in 2020 [41].

For automatic fake website detection, Maktabar et al. [42] utilized six machine-learning algorithms to detect fraudulent websites. They scraped textual data from 430 legitimate and fraudulent websites. For feature representation, they used bag-of-words and part-of-speech tagging. Among the classifiers used, multinomial NB and logistic regression achieved the highest performance in terms of the false-positive rate, accuracy, and F-score. With the aim of real-time fake shop detection, Beltzung et al. [43] experimented with different machine learning methods, including tree-based algorithms and SVM. For the dataset, they scraped 3,801 fake and 2,838 legitimate shops. The TF-IDF feature representation was used to represent the source code structure of the home pages. In their experiments, the EXtreme Gradient Boosting tree algorithm achieved an F1 score of 97%.

In the context of e-commerce, Carpineto and Romano [44] proposed a methodology for detecting fake e-commerce websites that appear in search results. To collect the data, different query terms were searched in different browsers, and the first 100 results were used. They scrapped 1000 legitimate and fake websites from the collected data. They utilized an SVM classifier with 35 manually selected features. In terms of accuracy, recall, and precision, SVM outperformed their baseline (a feature majority vote classifier), NB, and SVM (with TF-IDF of text-only features). They reasoned that text-only features did not contain enough discriminative e-commerce words.

2) DEEP MACHINE-LEARNING APPROACHES

In contrast to traditional machine learning, deep learning has received little research attention in the website classification field [41]. This section presents some of the studies on fake website detection and website classification. In addition, it presents binary and multilabel website classification.

a: BINARY WEBSITE CLASSIFICATION

A popular work on multiclass sentence classification was presented by Kim [45]. We included it, as it was widely used for text classification. Kim used the pre-trained 300-dimensional Word2Vec from Google to represent text as features. The proposed methodology was based on a CNN, where three parallel layers were used with multiple filter widths to extract different features from the processed text. The proposed methodology outperformed 14 other models in four of seven tasks, including sentiment analysis and question classification.

To detect phishing websites, Ali and Ahmed [46] used a hybrid approach of a genetic algorithm and deep neural networks. The genetic algorithm was used for feature selection and weighting, and it selected nine features from a list of manually selected features. They conducted their experiments using 1,353 phishing, legitimate, and suspicious websites. Their approach achieved the highest results in terms of accuracy, sensitivity, and specificity compared with the SVM, kNN, decision tree, and NB classifiers.

Li et al. [47] utilized the semantic and structural features of websites and approached a binary classification problem using a CNN model. They used a Document Object Model (DOM) tree to extract structural features and the Word2Vec model to extract semantic features. They used a dataset of 2,950 webpages forming five categories and compared their method with a text-based CNN. Their approach outperformed the baseline in terms of accuracy, recall, and F1.

To study the effect of using transfer learning in webpage classification, Gupta and Bhatia [48] used an ensemble approach by learning the contextual representation of webpages using a pre-trained BERT model and then applying deep inception modeling with residual connections. They compared the proposed approach with KNN, SVM, deep neural networks, BERT Base default, BERT Base+Nonlinear layer, and BERT Base+CNN. The proposed model achieved the best F1 scores on the five different datasets. For this

experiment, we think that using only the first 128 words per web page to make the classification affected the task badly by getting low scores, as the necessary information could be lost.

b: MULTI-LABEL WEBSITE CLASSIFICATION

The task of multi-label website classification has received little research attention; however, we focused on it as it would help us to establish our methodology. In the multilabel website classification context, Deng and Shen [49] approached the methodology by combining deep learning and machine learning to benefit from the ability to extract high-level features from a large amount of raw data and the ability to process high-dimensional features provided by LSTM and SVM, respectively. The proposed method was better than using SVM or LSTM independently in terms of accuracy.

Another work on webpage multi-label classification was presented by Vinh and Kha [50]. This study aimed to classify Vietnamese online news articles into their topics by using deep neural networks, SVM, and logistic regression. The dataset was scrapped from five online news websites and had 68,363 articles and 30 topics. The authors experimented with two feature representation methods: TF-IDF and n-gram n= (2, 3, or 5). In addition, they attempted to select different feature set sizes from the top-ordered TF-IDF features. They evaluated their work using a micro F1 score, where the best score was obtained using deep neural networks with a TF-IDF of 10,000 feature set size. However, this feature set included all the vocabulary in their work.

In the same context of multi-label classification, Artene et al. [51] used a CNN for multi-label multi-language classification. This study is an extension of their work in 2021 [52]. The authors used an in-house multilabel multilanguage dataset containing 8,798 web pages, 69 labels, and four languages. They used the first 5,000 words from each webpage. They represented the input data using Word2Vec embeddings. In their first study in 2021 [52], their CNN model achieved a micro F1 score of 0.79. In their second work in 2022 [51], they divided the classification problem into two problems: functional classification and subject classification, and they increased the total dataset to 12,432 webpages to improve the results. The F1 scores for functional, subject, and all (functional + subject) were 0.88, 0.84, and 0.74, respectively. However, although they tackled the multiclassification problem for web pages, most of the web pages in their dataset had only one label, and the highest total number of labels was five for only one web page. Moreover, the authors presented another study on the same dataset of 8,798 web pages and 69 labels in 2021 [53], where they incorporated transfer learning and contextualized embeddings in their methodology. Different experiments were conducted to evaluate different models, including the use of the pretrained multilingual BERT model and a hybrid model of the CNN model proposed in [52] with the pretrained multilingual BERT model. Additionally, different inputs were used to evaluate the models, including textual content, web page title, and HTML meta, which were either combined or separated. The best achieved micro F1 score of 0.85 was using all the inputs combined and the hybrid model of CNN and BERT.

D. DISCUSSION

In view of all that has been mentioned so far, predatory venues have serious effects on credibility and integrity of scientific literature. Although different solutions were presented, they were manual and had many false positives and negatives. Moreover, the number of venues was often too high, and new venues continued to appear, making it difficult to track all of them or have in-depth knowledge of them. Collectively, these studies outline the value of presenting an intelligent framework that would automatically detect legitimate and predatory publishing venues and provide appropriate reasoning about the violations of predatory venues.

Despite this, only one study has attempted to address automatic detection of predatory venues. Although this study has attempted to automatically detect predatory journals, it did not provide appropriate reasoning about the violations of predatory venues, and no studies have been found that manually or automatically provide appropriate reasoning about violations, except Cabell's framework, which manually justified the violations. Therefore, in this research, we propose an intelligent framework for the auto-detection of predatory venues with appropriate reasoning about violations using two artificial intelligence approaches: (1) machine learning and (2) deep learning.

We chose to approach our framework using a traditional machine learning approach, as it is widely used in fake website detection and website classification. However, the effect of deep learning techniques on website classification, specifically the impact of the sequence of words on the website's classification, has been overlooked [41]. Thus, we also decided to approach our framework using a deep learning technique to investigate its applicability in our domain and compare it with the machine learning approach.

III. RESEARCH METHODOLOGY

This section aims to present the methodology we followed to tackle the predatory venue problem. First, we briefly present dataset acquisition, legitimacy criteria compilation, and dataset annotation. We then present the proposed framework, and how we evaluated the proposed framework.

A. DATASET CONSTRUCTION

Dataset construction comprises three steps: dataset acquisition, legitimacy criteria compilation, and dataset annotation. The following subsections briefly describe these steps, and the next section (section IV) details the dataset construction process.

1) DATASET ACQUISITION

In this work, we aimed to build our dataset of predatory and legitimate journals, where we benefited from the DOAJ directory, the DOAJ rejected the journal's list, the DOAJ

removed journals list, and the Beal list to collect our legitimate and predatory journals. For extracting the content of the journals, we used the Selenium library [54], which is written in Python language, along with the Zyte API [55] for legal scraping of the textual content of the journals.

2) LEGITIMACY CRITERIA COMPILATION

Our evaluation criteria were compiled based on the Principles of Transparency. Our primary focus during this step was on the criteria that can be judged directly from the venues' websites. We excluded criteria that required further investigation, such as searching the web or an external database. The extracted criteria were used to evaluate the predatory journals to provide reasoning when a journal was classified as questionable.

3) DATASET ANNOTATION

There were two types of annotation processes in this study. The first type involves labeling journals as legitimate or not. We considered the journals extracted from (1) the DOAJ rejected journals list, (2) the DOAJ removed journals list, and (3) the Beal list as predatory/questionable journals. In contrast, the journals extracted from the DOAJ of legitimate journals were annotated as legitimate journals. This is because we used the principles of transparency that the DOAJ uses to evaluate journals. Hence, as the DOAJ evaluated the journals, we saved time by not re-evaluating them.

The second type of annotation annotates the predatory journals using legitimacy criteria to provide appropriate reasoning about the violations. To this end, we benefited from the Toloka platform [56], which is a crowdsourcing platform that offers human annotators many quality control methods and enables us to build annotation interfaces easily using HTML and JavaScript. In addition, every journal website was evaluated by at least three annotators to ensure quality control and different quality control methods were applied to the annotation process.

B. PROPOSED FRAMEWORK

Figure 1 illustrates the proposed framework. It consists of five main components: (1) website scraping, (2) content preprocessing, (3) feature selection, (4) feature extraction, and finally, (5) classification component. The annotated URLs of venues' websites are used by the components of our framework to detect legitimate journals among predatory ones and to detect the violations of the legitimacy criteria using two artificial intelligence techniques: (1) machine learning and (2) deep learning. Furthermore, natural language processing techniques were used to extract features from venues' websites. We briefly explain these components in the following sections.

1) WEBSITE SCRAPING COMPONENT

The website of a given journal consists of multiple pages, which in turn consist of scripts such as HTML, CSS, and JavaScript, which render the textual content of the journal. The textual content should be extracted automatically from the website; hence, it can be examined later using the components of our framework. Web scraping techniques in this component can be used to perform this complex process. We used Scrapy [57] and Selenium [54] with the Zyte API [55] to complete this component.

2) CONTENT PRE-PROCESSING COMPONENT

The textual content extracted from the previous components was unstructured and contained HTML tags, stop words, inflected words, and other unnecessary characters and words. This component performs the first step by converting this textual content into a structured feature space and cleaning it. It performs different pre-processing techniques, such as tokenization, tags, and stop-word removal.

We did not use dense pre-processing to follow Adnan's effective approach in our field [19]. In addition, as no one has experimented with our area, we want to build a base by using the text as is. Moreover, we believe that applying dense pre-processing to textual content could eliminate the writing behavior of predatory journals. We believe this is because some researchers have found that predatory journals' websites contain spelling mistakes or poor grammar [12]. Additionally, while collecting journals, we found that some predatory journals present content in different languages on the same page. Therefore, we did not apply dense pre-processing.

3) FEATURE SELECTION COMPONENT

The previous components produce a high-dimensional feature space from the textual content of a journal's website. It may contain millions of words; hence, memory complexity and processing time will be high. One assumption is that not all of these words are required, and some of them are simply irrelevant, where this assumption is mainly employed in traditional machine learning algorithms [58]. Hence, we deal with textual content as a bag of words (BoW) and filter the words that contribute better based on a scoring function. On the other hand, some researchers think that we need all words in a given text, and we need them to be ordered as they appear in the original text to understand the semantics behind the text [59]. This second assumption is primarily employed in deep-learning approaches.

In our experiments, we chose to try both methods, where we used TF-IDF and information gain scoring functions in some experiments to extract features from the BoW. This technique has been used in Adnan's study and has been shown to provide good results in our task [19]. Additionally, we experimented with n-gram and TF-IDF to preserve the semantics behind the text. Moreover, we tried the second approach, where we passed all the textual content without performing feature selection; then, we extracted a new feature space from it, as explained next.

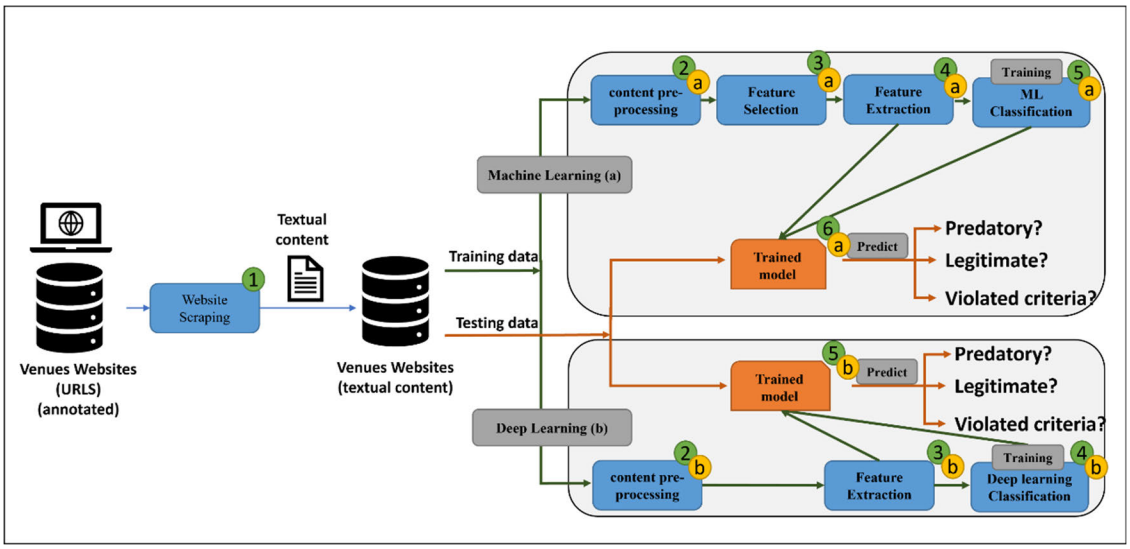

FIGURE 1. Framework for auto-detection of predatory venues.

4) FEATURE EXTRACTION COMPONENT

To work with the second assumption mentioned in the previous component, we used a feature extraction component. This component takes our pre-processed feature space (textual content without applying feature selection) and then creates a new representation of words with small dimensions [59]. With the feature extraction component, we do not need to specify the percentage of words that must be retained for them. The feature-extraction process included two cases. Either it learns with neural networks how to represent words/documents in a new representation that incorporates syntactic, semantic, or polysemousy of the words [59]. Some models in natural language processing perform these tasks, such as Word2Vec, Global Vectors for Word Representation (GloVe), FastText, and Doc2vec [59]. Alternatively, it does not preserve semantics because it is based on counted or weighted words (bag of words) as in the common techniques of feature extraction for traditional machine learning, such as Linear Discriminant Analysis (LDA) [59].

We attempted the first case in our experiment. Specifically, we used two methods to extract the features that are Word2Vec and Doc2vec. We chose Word2Vec because it performs better than TF-IDF feature representation in text classification tasks [60], as it tries to capture the syntactic and semantics of words. We used Doc2vec because we assumed that we can increase the prediction score if we can represent the entire long journal website into one vector that summarizes the website.

5) CLASSIFICATION COMPONENT

Given the final representation of features extracted from a journal website by the previous components, we want to classify the journal as legitimate or predatory. Additionally, for predatory journals, we wanted to detect the legitimacy criteria violations. The classification component performs this task. It will learn how to classify after being trained on examples of legitimate and predatory venues and on detecting the violated criteria as well. After the learning process, the resulting classification model can predict unseen journals. We employed two classification approaches: traditional machine learning and deep learning. This research has two tracks: detecting if a given journal is legitimate or predatory, and providing the appropriate reasoning if the journal is detected as predatory. We defined our problem as a binary classification problem for the first track. For the second track, we defined the problem as a multilabel classification problem. By labels in multilabel, we mean the legitimacy criteria defined in the criteria formulation section.

Different algorithms are available for machine learning and can be used in this component, such as SVM, KNN, and random forest [59]. For our legitimacy classification task (first track), we selected SVM for different reasons. First, we do not need to reduce the number of features by feature selection because SVM can learn independently of the feature space dimensions, even if the dimensions are sparse [58]. Second, Joachims [61], in his extensive experiments on text classification, demonstrated that SVM outperformed different classifiers, including KNN, decision trees, Naïve Bayes, and Rocchio algorithm. Along with SVM, we used the KNN algorithm because Adnan showed in his experiments that SVM and KNN outperformed Naïve Bayes and produced good results. For our legitimacy criteria violation detection task (second track), we evaluated the SVM model for the same reasons as above.

For the legitimacy classification task, we examined three different deep neural network models: Neural Networks (NNs), LSTM, and CNN. However, the CNN model was selected because it is used in the domain of text classification to extract the features of a text [45] and is used in website classification [47], as mentioned in the literature section.

Moreover, we examined the BERT [62], a recent language representation model. BERT achieved state-of-the-art results on 11 different NLP tasks, including sentiment analysis and question answering. In this study, we used the pre-trained BERT and ALBERT models to fine-tune our downstream text classification task (legitimacy classification task). For the legitimacy criteria violations detection task, we evaluated CNN as a deep learning model based on the results obtained in the first task, and because CNN achieved good results for multilabel classification, as mentioned in the literature section in [51].

C. FRAMEWORK EVALUATION

We evaluated the effectiveness of the proposed framework on the test set, which is a blind set that was unseen by the models during training and evaluation and is part of the annotated dataset, by deploying several case studies and utilizing different effectiveness evaluation measures. Each case study applied a different machine learning or deep learning approach. The results of the case studies were then compared using the evaluation measures.

To verify the effectiveness of detecting a journal's legitimacy, we used the F1 score for the positive class (legitimate journals). To evaluate the effectiveness of the criteria violation detection, we used both micro and weighted F1 scores. The selection is based on the fact that our dataset is unbalanced, and every journal has applied from 0 to 39 labels. Both micro-and weighted F1 scores represent a trade-off between precision and recall. However, micro F1 is calculated as a sample-label pair, where weighted F1 is calculated as the average of F1 of labels weighted by their contribution.

IV. DATASET CONSTRUCTION

In this study, we aimed to construct a dataset of online publishing journals with high coverage of different journals along with their violations. As detailed in the following sections, we began this process by collecting journals from lists of different journals. We then constructed our legitimacy evaluation criteria for evaluating the collected journals by following an iterative approach of reviewing and modification. After framing the criteria, we started the annotation process following another iterative approach that involved repeated rounds of reviewing and modification based on the obtained results. Finally, we present statistics for the final annotated dataset.

A. DATASET ACQUISITION

As mentioned earlier, the available lists of legitimate and predatory venues suffer from overlap. Nonetheless, they are not properly labeled by the violating criteria, which is the core objective of our research. In addition, the available lists do not detect the legitimacy of newly emerged journals. Therefore, we aimed to collect 10,000 legitimate and nonlegitimate journals. Hence, we utilized the available venues' lists. The following two sections detail how legitimate and questionable journals were acquired. After collecting the journals' URLs, we scraped them to extract their content so that they could be used later in classification algorithms.

1) LEGITIMATE JOURNALS LIST

In order to build our legitimate journals list, we utilized the DOAJ directory new list [63], as it has more than 12,750 journals that are evaluated as legitimate based on the Principles of Transparency criteria. We used Scrapy [57] to extract journals that accepted scraping. Simultaneously, we checked the content of the scrapped website using the textblob library[1] to extract only English websites. At the end, we reviewed a sample of the extracted websites. We found that some of the websites were not detected as non-English by the library, and some of the websites were no longer journals or ceased. Therefore, we checked all the extracted websites manually, resulting in 5,533 different legitimate journals' websites.

2) QUESTIONABLE JOURNALS LIST

For building a list of questionable journals, we started with Beall's list of predatory journals, which contained 1,462 journals, where we extracted the journals' URLs from the Beall website [8]. However, most of the journals (more than 1,000) on Beall's list are out-of-service. Moreover, some of the journals in Beall's list have been updated, and now maybe they are no longer predatory, which can affect our list. However, we needed a copy from Beall's journals when they were marked as predatory journals, that is, the same year they were classified as predatory by Beall, between 2010 and 2017. Therefore, we used the Wayback Machine[2] because it archives snapshots of internet websites at different times. We hired freelancers from Mostaql[3] to help us find Beall's journals' websites using the Wayback Machine. Three people were hired with fees ranging from 25$ to 60$, and each was asked to retrieve the nearest journal snapshot for 2017. Simultaneously, we reviewed freelancers' work to maintain the quality of the work. We asked the freelancers to review the complete list if they made many mistakes until the job was finished completely. In some cases, a journal could not be found in the Wayback Machine, or the website language was not in English. Additionally, some websites were not for journals; hence, the final result for this list was reduced to 1,290 journals.

We could not use the list of Stop Predatory Journals that we mentioned in Table 1 as it was built from Beall's list, and the JPPS journals' list, as it contains about only 400 journals with a low confidence level to be predatory. While searching for more predatory journals, we found two lists published by the DOAJ [63]. The first contained the journals removed from the DOAJ directory, and the second contained journals whose owners failed to submit a reapplication to DOAJ. In the first list, 4,009 journals were removed from the DOAJ directory because they did not adhere to the DOAJ criteria, and the

[1] https://textblob.readthedocs.io/en/dev/
[2] https://archive.org/web/
[3] An Arabic Freelancing marketplace https://sa.mostaql.com

removal dates ranged from 2014 to 2020. The removed DOAJ list does not represent the current state of the DOAJ directory, and some of the deleted journals were added later. Therefore, we deleted the journals if they were added to the DOAJ directory after their removal date (by checking another list of journals added to DOAJ). In addition, we deleted the journals if the reason for removal was not due to the journal website's textual content, such as ceased publishing, and no longer Open Access journal. The final results contained 2,652 journals. The second list had 2,860 journals removed from DOAJ on Monday, May 9, 2016, because the publisher failed to submit a valid reapplication (a form that asks to prove the application of the criteria) within the given timeframe.

Both lists contained only ISSNs and the titles of the journals in the lists and needed to be found on the Web, as we did not have their URLs. Therefore, to collect the URLs of the first and second DOAJ lists, we hired seven workers from Mostaql, with fees ranging from 30$ to 70$. Their work was to find journals on the web using our detailed instructions, where some workers were given more work than others based on their work quality and efficiency.

Generally, they were asked to use the provided ISSN and journal's title to search in the ISSN portal[4] and Google engine.[5] However, to ensure that an extracted journal is the same as the snapshot when it was deleted or rejected by the DOAJ, the employed workers were asked to find its archived copy from the Wayback Machine using removal or rejection data. This process consumed about three months, where during the process, we reviewed the extracted URLs to make sure they applied our conditions, and we asked the workers to review their work if we found many mistakes in their works. The final removed and rejected lists contained 1,449 and 1,607 journals, respectively.

To summarize, all previous non-legitimate lists, including the Beall and DOAJ lists, have approximately 8,000 non-legitimate journals. From which we extracted a new list of questionable journals that contained 4,333 journals, as presented in Table 3. All these journals are archived in the Wayback Machine, which is a way to ensure long-term preservation, as opposed to a standalone journal URL that can be stopped or updated at any time for any reason, and it may become a legitimate journal at any time after updating the content of the website.

At the end of the legitimate and predatory journals collection process, our constructed list of legitimate and predatory venues contained 9,866 journals. The questionable journals list has 4,333 journals, and the list of legitimate journals has 5,533 journals.

3) JOURNALS' LISTS SCRAPING

Because we need to perform machine and deep learning on the extracted content, we scrape the textual content of the extracted journals and the downloadable files, such as PDF and word files. To perform scrapping, we used Selenium [54] and Zyte API [55], which are legal and, based on our experiments, faster than Scrapy [57].

With regard to the scrapping task, we encountered many challenges. The main challenge was the massive amount of content per journal, as some journals' websites have approximately 10,000 web pages and require hours or even days to be scrapped entirely. Hence, we have reduced the scraping limit from the entire journal website to only the first three levels of journals. We selected the first three levels because we found that they were sufficient to extract critical information for the classification process based on checking different journals' websites. At the end of the scraping process, we scraped 6,836 journals, of which 2,724 were legitimate and 4,112 were predatory, as presented in Table 4. The number of scraped journals was less than the number of collected URLs for various reasons. First, some journals' websites did not accept scrapping, and we respected this. Second, some journals use images instead of texts, which requires an OCR tool to extract the text from the images; thus, we discarded these journals. Third, some journals use complicated structures and need to look through each code individually and build a specific scraper per journal, which consumes a long time. Therefore, we excluded them from the scrapping process.

The final scraped textual content was saved as a JSON object. For a given webpage, we saved different information, including the path from its HOME page to this web page, the title of the webpage as presented in the title bar, the level of the web page (level 0 is for the HOME page), the type of web page (HTML page or file), and the content of the web page.

TABLE 3. Statistics for collected predatory journals.

Data source	The total amount of journals in the source	Extracted journals
Beall journals	1,462	1,290
DOAJ journals (removed by DOAJ)	4,009	1,449
DOAJ journals (failed to submit an application)	2,860	1,607
Total	8,333	4,346
Total without duplicates	-	4,333

B. LEGITIMACY CRITERIA COMPILATION

Different criteria frameworks are available; however, there are no widely agreed criteria among scholars or organizations, and it is still an active research area. The Principles of Transparency contain 16 principles and address various topics. They form the basis of the criteria used to assess the suitability of membership by DOAJ, OASPA, COPE, and WAME. As the Principles of Transparency have been agreed upon by four known organizations, we utilized them to create our legitimacy criteria. Another reason for utilizing the Principles of Transparency is that they will provide us with a list of evaluated legitimate journals if they are extracted from

[4]https://portal.issn.org/
[5]https://www.google.com/

TABLE 4. Statistics about the scrapped journals.

Item	Scrapped journals
Scraping period	**7/10/2021 to 9/2/2022**
Total number of websites	**6,836**
Legitimate journals	**2,724**
Predatory journals	**4,112**
Total number of downloaded files	**7,050**
The total number of requested web pages	**about 3,000,000 requested pages, including invalid or out-of-service web pages**
The total number of the scraped web pages	**646,027 web pages (317,868 Predatory journals and 328,159 Legitimate journals)**
Size of the scrapped content	**11 GB**
Size of the downloaded files	**8 GB**

the DOAJ list, which in turn heavily uses the Principles of Transparency to evaluate a journal before adding it to their directory.

In the next section, we present the Principles of Transparency and derivation of our criteria. Then, we present how we formulated the extracted criteria in different rounds to meet the nature of our annotation/evaluation process.

1) CRITERIA EXTRACTION

The 16 Principles of Transparency are: 1) website, 2) name of the journal, 3) peer review process, 4) ownership and management, 5) governing body, 6) editorial team/contact information, 7) copyright and licensing, 8) author fees, 9) process for identification of and dealing with allegations of research misconduct, 10) publication ethics, 11) publishing schedule, 12) access, 13) archiving, 14) revenue sources, 15) advertising, and 16) direct marketing.

To obtain a detailed description of these criteria, we used three different sources [64], [65], [66]. We derived 58 criteria from the Principles of Transparency using a description of the principles. We then examined these 58 criteria and extracted those that matched our scope. Subsequently, we extracted from these principles the criteria that can be evaluated directly from a journal's website. Other criteria that are not built upon journal content, such as those that discuss the publishers' or editors' behavior, those that need further searching on the web, or those that need an external database search, are not included because this research aims to detect predatory journals automatically from the website textual content. In addition, we excluded the criteria related to phishing websites.

Finally, we extracted 39 criteria from the 58 that can be judged from the website content.

2) CRITERIA FORMULATION

In order to formulate the extracted criteria in a form that attracts annotators and helps them annotate journals accurately and faster, we needed to reformulate the extracted criteria into new groups. This process was performed across two rounds. The first round aimed to put the criteria that could be evaluated at the same time into new groups, while the second round aimed to decompose the created groups into smaller groups in a way that attracts annotators to our task.

In the first round, we decomposed the 39 extracted criteria into five groups and hired five English-speaking workers from Mostaql to work on these groups. We gave them the criteria and asked them to evaluate 20 websites based on the given criteria. Each worker had a different set of topically related criteria and had some evaluated examples. We asked these workers to evaluate the given websites and attach a link to the website to prove their evaluation. When they submitted their work, we reviewed it, and if we noticed that they did not understand a given criterion, we asked them to redo the work and give them some examples and explanations of the criteria provided by COPE [66].

The process of understanding and forming the final criteria is done along with the five works. At the end of this process, we changed the first created five groups and finalized the forming of the criteria into 17 new groups. In this round, the criteria that can be evaluated simultaneously are merged (i.e., have the same or related content).

In the second round of criteria formulation, we consider the nature of evaluators in the annotation frameworks, as we need to evaluate our questionable journals list using an annotation framework. We can have thousands of people, and we want them to understand our task and complete it well. We did not want to confuse them, and we wanted to avoid noisy data in our dataset. Therefore, to help complete our annotation task much faster, with high quality, and contribute to better quality control, we needed to decompose the criteria again into small groups. In this round, we ran the annotation process on 100 journals and noticed that the annotation process was slow; therefore, we divided some groups into smaller ones, such as the peer review group, which was divided into two groups. At the end of this round, we created 20 groups of our criteria, where each criteria group contained one, two, or three criteria. Moreover, we annotated 20 different journals based on our 39 criteria, along with five workers. Table 5 lists the formulated criteria.

C. DATASET ANNOTATION

As we need to classify journals as predatory or legitimate and provide appropriate reasoning about violations, the collected dataset must be labeled as legitimate or not, and non-legitimate journals must be labeled by their violations.

The process of labeling journals as predatory or legitimate is not needed, as legitimate journals were collected

TABLE 5. Formulated criteria.

#	Criteria	Description
1	Aim and scope	The journal's website has an 'Aims & Scope' statement that is clearly defined.
2	peer review	Peer review is mentioned. Peer review process and model are mentioned.
3	Peer review time	No manuscript acceptance or a very short peer review time is guaranteed (24 hours, 7 days).
4	post-publication corrections	A journal should have policies on publishing ethics. These should be clearly visible on its website and should refer to the journal's options for post-publication discussions and corrections.
5	content display way	The way(s) in which content is available to readers is stated.
6	authorship	A journal should have policies on publishing ethics. These should be clearly visible on its website and should refer to journal policies on authorship and contributorship.
7	Complaints and conflicts of interest	A journal should have policies on publishing ethics. These should be clearly visible on its website and should refer to how the journal will handle complaints and appeals. A journal should have policies on publishing ethics. These should be clearly visible on its website and should refer to journal policies on conflicts of interest/competing interests.
8	Journal archive/ publishing schedule	The periodicity of publication is not indicated. The publishing schedule appears erratic from the available journal content.
9	electronic backup/ archiving	A journal's plan for electronic backup and preservation of access to the journal content shall be clearly indicated.
10	copyrights	The policy for copyright shall be clearly stated in the author's guidelines. Publishing licenses are missing or unclear. A user license is missing or unclear.
11	creative commons	If authors are allowed to publish under a Creative Commons license, then any specific license requirements shall be noted. Any policies on posting final accepted versions or published articles on third-party repositories shall be clearly stated.
12	manuscripts	The listed articles are not available at all. The copyright holder is not named on all published articles. Advertisements shall be kept separate from the published content.
13	data sharing and reproducibility	A journal should have policies on publishing ethics. These should be clearly visible on its website and should refer to journal policies on data sharing and reproducibility.
14	ownership and management	Information about the ownership and/or management is missing, unclear, or misleading. The editor-in-chief is also the owner/publisher.
15	editorial board	Full names and affiliations of editorial board members are missing. Full names and affiliations of the journal's editor/s are missing. Full contact information for the editorial office is missing (email, phone, address).
16	ethical oversight	There is no description of how cases of plagiarism are handled. There is no description of how cases of citation manipulation are handled.
17		There is no description of how cases of data falsification/fabrication are handled. There is no description of how cases of consent to publication are handled.
18		There is no description of how the ethical conduct of research using animals is handled. There is no description of how the ethical conduct of research using human subjects is handled.
19	advertising	Journals shall state their advertising policy, including what types of adverts will be considered. Journals shall state their advertising policy, including who makes decisions regarding accepting adverts. Journals shall state their advertising policy, including whether they are linked to content or reader behavior or are displayed at random. Advertisements should not be related in any way to editorial decision making
20	fees and revenue sources	Business models, business partnerships/agreements, or revenue sources shall be clearly stated or otherwise evident on the journal's website. The way(s) in which content's associated costs are not stated, such as whether there are associated subscriptions or pay-per-view fees. Any fees or charges that are required for manuscript processing and/or publishing materials in the journal shall be clearly stated. This must be: – in a place that is easy for potential authors to find prior to submitting their manuscripts for review. – if no such fees are charged, then it should state that.

from the DOAJ directory. Thus, they were evaluated by DOAJ workers based on the Principles of Transparency and other DOAJ-specific criteria. Additionally, non-legitimate journals were either extracted from Beall's list of predatory journals or removed from DOAJ because they did not meet the criteria or failed to submit the reapplication to DOAJ (a form that asks to prove the application of the criteria).

In contrast, we needed to evaluate and annotate questionable journals based on their violations. Therefore, we used the extracted legitimacy criteria to perform annotation. For each group of criteria, we created a specific annotation task with its guidelines, interface, incentives, and quality control methods. We applied different quality control methods to maintain the quality of the annotations obtained. These methods include, but are not limited to,

selecting annotators, controlling annotators' behavior, building instructions, and training annotators. To achieve the required level of quality, we divided the annotation process into different rounds and employed an iterative approach to annotate our data. At the end of each round, we utilized the lessons learned from the round results, and based on them, adjusted the annotation guidelines and tuned the task parameters.

The entire annotation process is detailed in the following sections, starting from creating the initial annotation guideline and ending by providing some statistics about the annotated dataset.

1) ANNOTATION GUIDELINES

Building clear guidelines helps improve the quality of annotation tasks. We employed a reverse engineering approach to build guidelines for the annotation process. We constructed the first draft guidelines, along with the five workers who annotated the first 20 journals. We utilized COPE use cases of criteria violations [59] to define our criteria and provide examples to annotators. At the end, we build annotation guidelines for the 20 groups of criteria. The resulting guidelines were then used to annotate the remaining journals. However, we adjusted the guidelines as needed, based on the questions of the annotators.

2) ANNOTATION INTERFACE

It is important to build an effective annotation interface to enable crowd workers to complete annotation tasks faster, improve the annotation quality, and provide better results. We used the Toloka platform [56] to build our interface. In our annotation interface, when a performer opens the interface, we pack for him three annotation tasks, and (s)he is required to complete all the given three tasks to be able to submit. Thus, performers do not waste time switching between annotation task pages and do not lose their attention. We built a cross-platform interface for the computers, tablets, and smartphones.

3) ANNOTATION INCENTIVES

We motivated crowdworkers by giving them monetary incentives per completed annotation task, and we offered them quality-based earnings to prevent crowdworkers from spending the minimum effort required for the task to be approved and earn money quickly. Thus, the performer earned a fixed payment for his completed tasks and a bonus for his effective work. We measured the effectiveness through our golden set. Table 6 presents the annotation incentives, including fixed payments per worker and quality-based earnings. Moreover, while payment is very important, framing of the task itself is a crucial component of performers' motivation, as shown by the experiment conducted by Dan Ariely [67]. Therefore, we acknowledged that the performers' work is essential and meaningful by briefly telling them through our interfaces how their answers will be used in our research.

4) ANNOTATION QUALITY CONTROL METHODS

Quality control methods are essential in the annotation process, as we have thousands of crowdworkers whose background we do not know and whether they understand our annotation task correctly. As shown in Table 7, we used different quality control methods to control the quality of our annotation tasks, which were divided into three parts:1) quality control methods applied before the annotation process, 2) quality control methods applied during the annotation process, and 3) quality control methods applied after the annotation process.

5) ANNOTATION TASKS

For each group of criteria in Table 5, we created a specific annotation task with its guidelines, interface, training, and quality control methods, as mentioned previously. We ran seven rounds to collect annotations for the collected journals. Where we were learning new lessons from the first rounds, we were then updating the later rounds based on them. The entire annotation process was completed in approximately six months. In the following sections, we describe the annotation rounds, lessons learned during each round, and results of the annotation process.

a: ROUND (1): FREELANCERS' ANNOTATIONS FOR 20 JOURNALS

At the start of this round, we did not have any annotated data, and we had the criteria that were extracted from the Principles of Transparency in Section II. This round began before the step of formulating the criteria were developed. At the start of this round, we aimed to build our golden set such that it comprised 20% of the dataset to be annotated. We benefitted from Appen, which has worked in the crowdsourcing industry for approximately 25 years, where they recommended having at least 20% of the data be golden set to control the quality of the annotation process [69]. Therefore, we aimed to annotate 20 journals as our golden set for annotating the other 80 journals. To this end, we interviewed different workers and explained to them our work. We then ended up hiring five workers from Mostaql to annotate our journals with fees ranging from 30$ to 45$, and they finished their work within two weeks. Every worker was given 20 journals and asked to use 6 to 7 criteria to evaluate the journals. The freelancers' work passed through different rounds of revision, where, in each round, we reviewed the work and asked them to review it entirely if we found many mistakes.

At the end of this round, we collected 20 annotated journals based on our extracted criteria. However, we concluded that we could not continue this way as it would consume a long time to annotate the entire dataset ($5000 \times 1/20 \times 14 = 3{,}500$) days. Therefore, we decided to use the 20 annotated journals as a golden set in a crowdsourcing platform, where we can use it to control the quality of the annotation process. Moreover, as mentioned in the criteria formulation section, we utilized workers' evaluations and proofs to reframe the

TABLE 6. Annotation incentives.

Payment	Type
0.03$	Fixed payment (which we sometimes increased to 0.9$ to attract experienced workers).
0.02$	Bonus for achieving quality of 0~60%
0.03$	Bonus for achieving quality of 61~75%
0.04$	Bonus for achieving quality of 75~80%
0.05$	Bonus for achieving quality of 80~100%

criteria in new groups. Additionally, this round helped us establish a base for the instructions by benefiting from the questions of the workers and their different evaluations.

b: ROUND (2): CROWDWORKERS' ANNOTATIONS FOR 80 JOURNALS

This round aimed to utilize the wisdom of the crowd to annotate the datasets. Thus, we can solve the challenges of the previous round, where, based on our limited time, we cannot finish the entire annotation task with the settings of round (1). Therefore, we utilized the Toloka platform [56] to construct our annotation interface, guidelines, training, and quality control methods for the 20 criteria groups listed in Table 5. We created one project for each criterion group. Each project was composed of annotation tasks to annotate the journals, and each annotation task contained one–four questions based on the number of criteria in the criteria group. When a worker starts an annotation task, we display three annotation tasks, where one of the three tasks is a control task. The control task was randomly selected from the golden set created in round (1) to calculate the work quality of the annotators (percentage of correct annotations). Every annotator can solve one or more annotation tasks based on his work quality. Moreover, we collected approximately two to five annotations for each annotation task, as described in the previous section.

We ran this round to collect 80 annotations, where we used the 20 annotated journals as our golden set/quality control set. All Toloka [56] crowdworkers from around the world could annotate our dataset as long as they had English and passed our tests. However, while we ran this pool, we encountered different questions that showed that some performers did not understand our task. From their basic questions, such as 'What is peer review?' We understand that the workers were not experts in the research field. Moreover, they spent a long time finishing the tasks with low quality (with an accuracy of 40%). Therefore, as clarified in the next round, we stopped all tasks to find a solution.

c: ROUND (3): FINDING EXPERTS

From the previous round, we concluded that we needed expert annotators with a high educational level, such as a master's degree, professional degree, or doctorate degree. Therefore, we conducted a survey on Toloka [56] performers, and asked them different questions, including their age, gender, education, and current employment status. We collected answers from 8,411 workers. During the survey, we dynamically assigned a specific skill to workers with a high educational level. Other questions unrelated to the education level were added to the survey to eliminate spammers and distract the annotators about our aim of skilling highly educated users. At the end of this round, we had about 1,416 users with high education and English language proficiency.

d: ROUND (4): EXPERTS' ANNOTATIONS FOR 80 JOURNALS

This round ran with the same settings as round (2), except that only the experts collected in round (3) could see our annotation tasks. This round started with the peer-review criteria task and ended with the advertising sources criteria task. Finally, we aggregated all task results in this round using David and Skene aggregation, as described in Section IV-C6. During this round, we utilized the questions of the performers and adjusted our guidelines by explaining them based on the questions of the annotators.

After ending this round, we built a robust 100 golden set with a confidence level of 94.45% (accumulated by accuracy averages). However, the tasks slowly ran because we were limited to 1,416 experts who did not regularly access the Toloka platform [56]. In addition, we found that only approximately 700 of them were interested in our tasks. At this pace, we needed about ($5000 \times 1/80 \times 30 = 1{,}875$) days to finish our tasks.

e: ROUND (5): CROWDWORKERS' ANNOTATIONS FOR 400 JOURNALS

In this round, we used the annotations of 100 journals that resulted from the previous round to serve as a 20% control set of 500 journals; hence, we collected annotations for 400 new journals. The round (5) settings were similar to the round (2) settings, except for the number of journals to be evaluated and the golden set size. This round began with the aim-scope criteria task and ended with the ethical oversight criteria task. We aggregated all task results in this round using David and Skene aggregation with a macro average of 93.78%. During this round, we reflected on an explanation of the performers' questions on the guidelines page.

TABLE 7. Annotation quality control methods.

Quality control method	When is it applied?	Description
task (criteria) decomposition, guidelines, effective interface, the 20 golden set that is annotated and reviewed by experts, and the 100 golden set that is annotated with high educational crowdworkers	Before annotation process	See CRITERIA FORMULATION, and ANNOTATION TASKS.
Crowdworkers selection tool	Before annotation process	Select the annotators based on their language and client (device)
Crowdworkers training	Before annotation process	Train the annotators on the same annotation task, where they must complete the training to perform the annotation tasks.
Crowdworkers' skills	During the annotation process	The skill (accuracy) of a given performer is calculated based on his answers to our golden sets. Only accurate performers whose accuracy is equal to or greater than 80 could access our annotation tasks.
External behavior checks	During the annotation process	1) We used ***CAPTCHAS.*** 2) We MONITORED THE ***ANNOTATORS' SPEED.*** 3) We monitored the annotators who are ***SKIPPING NECESSARY ACTIONS.*** For all of these behaviors, we ban the annotators if their behaviors are not accepted.
Internal behavior checks		We used ***overlap checks*** and ***golden sets simultaneously*** to perform internal quality checks. In the overlap checks, we compare the answers to the majority. While in the golden sets, we compare the answers to the golden sets. The annotator has a score that is calculated based on these two checks. This score enables us to ***stop ineffective performers*** and to ***COntrol when to stop collecting the annotations per task.***
Annotation results aggregation	After the annotation process	How to aggregate the noisy responses of multiple workers into a single high accurate answer. We performed annotation results aggregation to aggregate the noisy responses of multiple workers into a single high accurate answer using David and Skene [68].

f: ROUND (6): CROWDWORKERS' ANNOTATIONS FOR 1,950 JOURNALS

In this round, we used the 500 journals' annotations that resulted from the previous round to serve as a 25% control set of 1,949 journals. Hence, we collected annotations for 1,449 new journals. We increased the percentage of the control set to 25% to increase the number of annotators who could simultaneously perform the tasks. This round's settings were similar to those of round (2), except for the number of journals to be evaluated and the golden set size. However, during this round, we found that some tasks ran very slowly, which could have negatively affected our limited time. Therefore, we checked the round settings and found that the required annotation time could be decreased by decreasing the total number of evaluations per task from five to three. Therefore, we applied this change to the running tasks, but we did not redo the completed tasks that used a maximum of five evaluations. This round started with the aim-scope criteria task and ended with the manuscript task. We aggregated all task results in this round using David and Skene aggregation with a macro average of 90.87%. During this round, we reflected on an explanation of the performers' questions on the guidelines' pages.

g: ROUND (7): CROWDWORKERS' ANNOTATIONS FOR 2,300 JOURNALS

In this round, we aimed to annotate the remaining 2,300 journals. Some tasks were started after completing the tasks in round (6). However, due to time and budget constraints, we decided to stop running tasks and adhere to the 1,949 annotated journals. As a learned lesson for future work from the previous rounds, we realized that we could decrease the annotation time if we provided the website as a screenshot of its web pages instead of a URL because the Wayback Machine consumes a long time to serve clients, which did not help attract annotators to our tasks.

6) ANNOTATION RESULTS AGGREGATION

The final step in the annotation process is to aggregate the noisy responses of multiple workers into a single, highly accurate answer and obtain the wisdom of the crowd. As many experiments have demonstrated, careful, aggregated responses from a large crowd of unskilled individuals might be of higher accuracy than the answers of a domain expert [56]. We did not rely only on the majority vote to aggregate the answers because it treats the answers of each performer equally, where the crowd is diverse, and some performers may be stronger than others. Therefore, we used the weighted majority vote algorithm of David and Skene [68] because it carefully accounts for the difference in performance abilities by weighting the answers of each performer based on their ability. In David and Skene's method, performers' abilities are not calculated based only on their answers to our golden set, which could be inaccurate because we know only the answers to a small number of questions. In contrast, it uses the whole dataset to find the performers' abilities and the best annotations that best feed the data [68]. Thus, both the majority vote and answers to the golden set are used to calculate performers' abilities. We presented the macro average of the aggregation results per round above with a discussion of the annotation rounds.

7) ANNOTATION RESULTS NORMALIZATION

The projects and annotation tasks that we had created were in the form of multiple-choice questions, where a question could have many answers (two to nine answers per question). We created the tasks in this manner to help the annotators answer our tasks easily and not confuse them. As a result, we needed to normalize the answers so that each task had two answers (yes, no) to enhance the classification process, as we did not have many examples for each answer.

8) STATISTICS OF THE FINAL ANNOTATED DATASET

We collected 9,866 journals' URLs for legitimacy annotation, of which 4,333 were labeled as questionable journals and the remaining 5,533 were labeled as legitimate journals. However, as mentioned in the journals' list scraping section, we were able to scrape only 6,836 journals, of which 4,112 were labeled as predatory, and 2,724 were labeled as legitimate.

For the legitimacy criteria violations annotations, we annotated 1,942 predatory journals due to time and budget constraints. Twenty criteria groups were used to annotate each journal. Each criteria group contains one–three criteria. Each journal was annotated using 39 criteria/labels. Figure 2 presents a histogram of the number of criteria that the annotated journals have applied. We can see that the distribution is relatively follow a normal distribution, where most of the journals applied 17 to 23 criteria.

Additionally, we can see that the number of journals that applied a small set of criteria are relatively small. In addition, the opposite is true, where the journals that applied a large set of criteria are relatively small. No journal applied all 39 criteria, and the best journal among the collected journals applied only 32 criteria. These statistics match the status of the annotated journals as predatory and provide good indicators for our collected predatory list.

V. EXPERIMENTS SETUP

In this section, we describe the experimental setup that was used to evaluate the proposed framework. First, we present the dataset used in the experiments. Next, we detail the settings of the components of the framework. Subsequently, we present the baselines and evaluation metrics.

A. DATASET

We used our constructed dataset in our experiments. It contains 6,836 legitimate and predatory journals. Where for each journal, we have the content of its website up to level three and its pdf, doc, and docx files. However, based on our experiments on Colab Pro+[6] which provides 50GB RAM, we could not use the downloadable files or level three because of memory constraints. Hence, we experimented only with the first two scrapped content levels.

We split the data into (80%, 20%) splits. We used the training set for both training and hyperparameter tuning through cross validation. The optimized models were then tested on the testing held-out dataset. The average number of words in 50% of the journals was 6,926 words, and our dataset contained 3,003,770 unique words. Figure 3 shows that most of the journals in our dataset have fewer than 4,000 words, whereas a very small number of journals have approximately 10,000 words in their content.

For legitimacy criteria violation, we were able to scrap 1,894 journals from the annotated 1,942 journals. For the dataset split, we used a split mechanism that was different from the previous one. Given that we have approximately 39 classes\labels\criteria and that each journal can apply different criteria, we tried to balance the training and testing set splits. Hence, the same criterion\label was used in both the training and testing sets. We first tried to use the stratified split using all the labels\criteria, but it was not applicable in our case because we needed at least 2^{39} labeled journals to be able to use it [70].

Thus, we attempted to better understand our dataset by creating statistics and figures. Figure 4 presents the statistics about the number of journals that applied each criterion. As can be seen, our dataset is relatively unbalanced. Therefore, we used part of the criteria to split our dataset using a stratified split, as it does not work for all criteria. We began by splitting the data using the least-distributed label applied by only 550 journals. Then we incrementally added more labels from the least distributed labels, and at the same time, we were testing if we could use the stratified split. We stopped when the stratified split was not applicable. We used six labels in the split process to create the training and testing sets. These criteria are 'content as publication periodicity,' 'peer review time,' 'complaints,' 'animals,' 'copyright holder,' and 'human subjects'. The resulting dataset contained 85% (1,609) were training journals and 15% (285) were testing journals.

B. CONTENT PRE-PROCESSING COMPONENT

The content passed to this component contains textual content from the journal websites. We applied several pre-processing steps before feeding them into the feature selection component. The pre-processing steps are as follows:

- **Tags removal:** We applied tag removal because our aim is to use the textual content for the prediction tasks. We filtered out all HTML, JavaScript, and other scripting tags while scraping the content using Beautiful Soup.[7]
- **Stop words removal**: We used the stop words list provided by Sklearn[8] to remove stop words.

The final output of this component is the tokenized text that is passed to the feature selection component. Table 8 presents the textual content statistics of the dataset before and after content pre-processing. This indicates that there were small

[6]https://colab.research.google.com/signup

[7]https://beautiful-soup-4.readthedocs.io/en/latest/

[8]https://scikit-learn.org/stable/modules/feature_extraction.html#stop-words

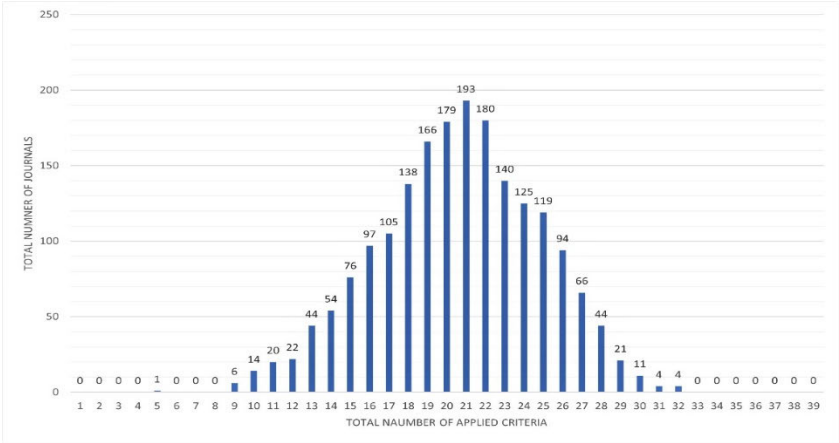

FIGURE 2. Number of criteria that the collected journals have applied.

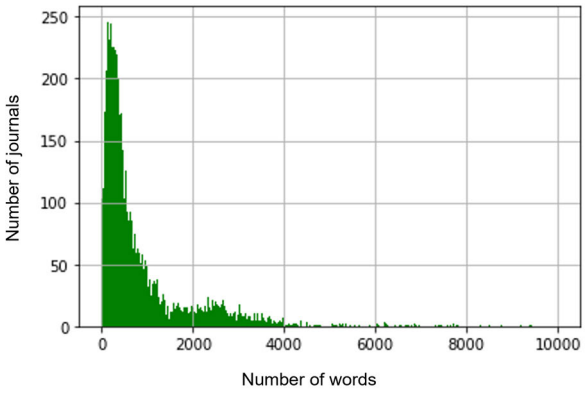

FIGURE 3. Number of words in the collected journals.

TABLE 8. Dataset's textual content statistics before and after performing content pre-processing.

Item	Before pre-processing	After pre-processing
Mean of the number of words	17,446 words	16,807 words
The standard deviation of the number of words	41,911 words	40,517 words
Minimum number of words	4 words	4 words
Maximum number of words	2,054,330 words	1,967,132 words
Number of unique tokens	3,003,770 unique tokens	3,003,512 unique tokens

changes in the statistics. For example, the average number of words in a journal was reduced by 639.

C. FEATURE SELECTION COMPONENT

For the feature selection component, we worked with two assumptions: 1) convert text to BoW and apply some feature selection methods or 2) pass the original tokenized text without applying feature selection to the feature extraction component. The feature selection methods that we experimented with were information gain, TF-IDF, and n-gram. We used the Scikit-learn [71] library for the feature selection component.

An ***n-gram*** represents a contiguous sequence of n words from a given text. ***Information gain*** scores the features for their efficacy in the classification task, and different studies have reported the effectiveness of information gain for feature selection [19]. In our field, the information gain of a feature (word) can be defined as the unpredictability reduction in a journal's classification. We calculated the information gain for all words in our training set.

TF-IDF feature selection has enhanced the performance of the models in the text classification field based on comprehensive experiments [50]. With TF-IDF, the word importance increases with the word's frequency in a given journal and is offset by the word's frequency in all journals. TF-IDF is calculated per word as in (1), where $tf\left(w_k, d_j\right)$ denotes the frequency of a word w_k in document d_j, while $|D|$ represents the total number of documents, $df\ (w_k)$ represents the document frequency of word w_k (i.e., the number of journals that contain the word w_k).

$$\text{tf.idf}\left(w_k, d_j\right) = \text{tf}(w_k, d_j).\log\frac{|D|}{\text{df}(w_k)} \tag{1}$$

We experimented with information gain and TF-IDF scoring functions for machine learning experiments, as Adnan did in his experiments [19]. We used information gain and TF-IDF as follows. First, we created a feature set that contained all the words in the training set ordered using information gain. Next, we let the model determine the best feature set size (5, 50, 100, . . . , 100,000) from the feature set. Third, each journal's words were represented using a vector with a size equal to the selected feature set size, where the elements in the vector were the TF-IDF of the words that appeared in the selected feature set.

Additionally, we experimented with n-gram (n=1 to 4) and TF-IDF, and let the model choose the best value for n (1,2,3,4). The vector size ranged from (5, 50, 100, . . . ,

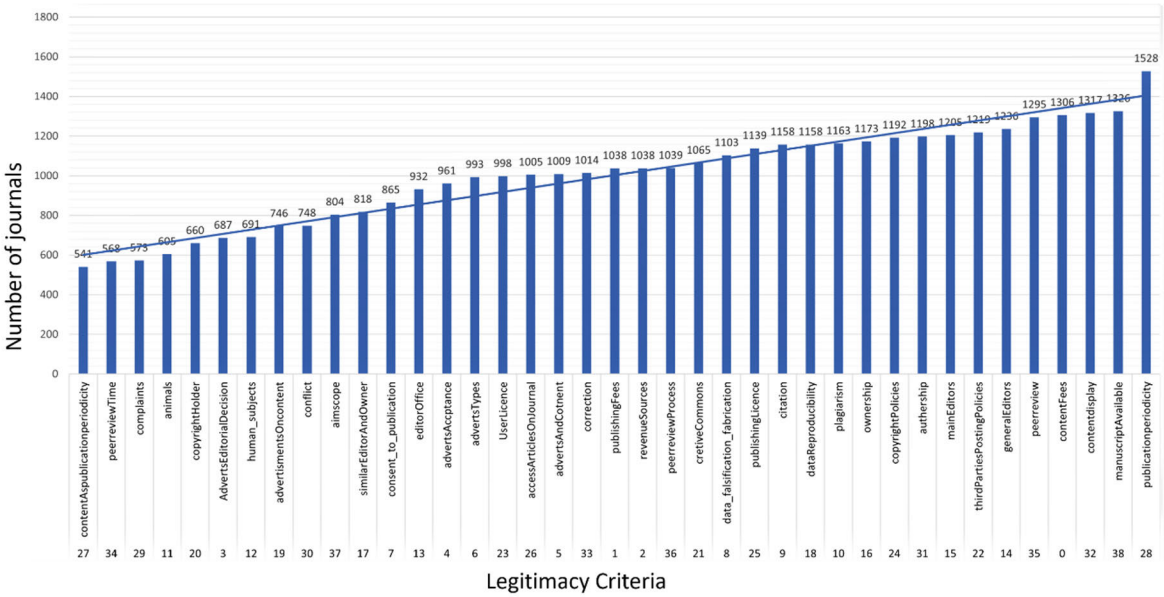

FIGURE 4. Number of journals that applied each criterion/label.

pt

100,000), and we allowed the model to choose the best feature set size. Using this method, each journal was represented as a vector of its n-gram weighted using TF-IDF.

For the deep learning experiments, we experimented with TF-IDF, where we scored the words in a journal using TF-IDF and created a vector for each journal. We experimented with different vector sizes (300, 500, 1000, and 7000) for all the evaluated deep-learning models. We stopped at 7,000 because about half of the journals in our dataset contained about 7,000 words; we also faced memory constraints when we tried above 7,000. For BERT and ALBERT, we used the first 512 words without feature selection because they only accept a maximum length of 512 words.

D. FEATURE EXTRACTION COMPONENT

For both machine learning and deep learning models, this component maps each journal, represented as a sequence of integer indexes of its words, to a 100-dimension or 300-dimension vector space using Word2Vec or Doc2Vec pre-trained models. For the machine learning models, we evaluated Word2Vec and Doc2Vec with the SVM model. For deep learning models, we evaluated Doc2Vec with all the models, whereas we evaluated Word2Vec with the CNN model. We used the Word2Vec and Doc2Vec models, as follows:

1) Word2Vec FEATURE EXTRACTION

To extract Word2Vec features, we trained the CBOW Model and evaluated the models using two popular pre-trained Word2Vec sizes (100 and 300). After training our Word2Vec model, the tokenized content of a given journal was converted using the trained Word2Vec model to ($m \times 100$) or ($m \times 300$) journal matrix, where m is the length of the journal (number of tokenized words), and (100 or 300) is the size of the vector representing a tokenized word. We aggregated the resulting journal matrix for machine learning models to (1×100) or (1×300) because the used machine learning model accepts a one-dimensional vector per journal. For aggregating a journal matrix, we evaluated two different ways: summing the vectors per journal and averaging them. In contrast, we did not aggregate the journal vectors for the evaluated CNN model because CNN accepts two dimensions' representations ($m \times 100$).

2) Doc2Vec FEATURE EXTRACTION

For the Doc2Vec model, we used the PV-DBOW model to extract Doc2Vec features [72] with vector space sizes of 100 and 300. Owing to memory constraints, we could not experiment with a vector size of 500. In the Doc2Vec model, a tokenized content of a given journal was converted using the trained Doc2Vec model to (1×100) or (1×300) journal vector. We used Doc2Vec with all evaluated machine learning and deep learning models.

We used the Genism library[9] to train our Word2Vec and Doc2Vec models. We trained these models on the entire pre-processed dataset, including the testing set, as this does not affect the classification because the models do not see the actual labels of the journals, but only try to learn the semantics or structure behind the text and extract new feature representations for the text. However, we did not perform any feature extraction process for the BERT and ALBERT models because they dynamically extract feature representations from any given journal content using the language gained

[9]https://radimrehurek.com/gensim/apiref.html

TABLE 9. SVM Hyperparameter settings and feature set size.

Case study #	Training data	Feature representation	Grid search hyperparameters	Selected values
1	Full training set	n-gram + TF-IDF	1- N-gram range: (1,2) 2- TF-IDF feature set size: (500, 1000, 2000, 3000, 4000, 10,000) 3-kernel (linear, rbf) 4-gamma (0.001,0.01,0.0001) 'if kernel=rbf' 5- regularization parameter (C): (1,10,100,1000,2000) 'if kernel=rbf'	1- N-gram range: (1,2) 2- TF-IDF: (500) 3- kernel (rbf) 4- gamma (0.001) 5- regularization parameter (C): (2000)
2	Full training set (First level of the dataset: Home page)	n-gram + TF-IDF	1- N-gram range: (1,1), (1,2), (1,3), (1,4) 2- TF-IDF feature set size: (500, 1000, 10000, 20000, 30000, 40000, 50000) 3- kernel (linear, rbf) 4- gamma (0.001,0.01,0.0001) 'if kernel=rbf' 5- regularization parameter (C): (1,10,100,1000,2000) 'if kernel=rbf'	1- N-gram range: (1,3) 2- TF-IDF: (30,000) 3- kernel (rbf) 4- gamma (0.001) 5- regularization parameter (C): (1000)
3	Full training set	Word2Vec with 100 as a vector size	-	-
4	Full training set	Word2Vec with 300 as a vector size	-	-
5	Full training set	Doc2Vec with 100 as a vector size	-	-
6	Full training set	Doc2Vec with 300 as a vector size	-	-
7	A subset of the training set contains 2730 journals per class	n-gram + TF-IDF	1- N-gram range: (1,2) 2- TF-IDF feature set size: (500, 1000, 2000, 3000, 4000, 10,000) 3-kernel (linear, rbf) 4-gamma (0.001,0.01,0.0001) 'if kernel=rbf' 5- regularization parameter (C): (1,10,100,1000,2000) 'if kernel=rbf'	1- N-gram range: (1,2) 2- TF-IDF: (10,000) 3- kernel (rbf) 4- gamma (0.001) 5- regularization parameter (C): (1000)
8	A subset of the training set contains 100 journals per class	n-gram + TF-IDF	1- N-gram range: (1,1), (1,2), (1,3), (1,4) 2- TF-IDF feature set size: (5, 50, 100, ... 100,000) 3-kernel (linear, rbf) 4-gamma (0.001,0.01,0.0001) 'if kernel=rbf' 5- regularization parameter (C): (1,10,100,1000,2000) 'if kernel=rbf'	1- N-gram range: (1,4) 2- TF-IDF: (500) 3- kernel: (rbf) 4- gamma: 0.01 5- regularization parameter (C): (1000)
9	Full training set	Information gain + TF-IDF (removing the first five words)	1- information gain feature set size: (50, 100, ….100,000) with the first five words removed	1- information gain feature set size: (50)

during pre-training on a massive corpus of 3,300 million words [62].

E. CLASSIFICATION COMPONENT

This component receives the final representation of the features extracted from a journal website and classifies the journal as legitimate or predatory. If the journal is classified as a predatory journal, then legitimacy criteria violations are stated. The classification component learns how to classify after being trained on the training data containing examples of legitimate and predatory journals and being trained to detect the violation. After the learning process, the resulting classification model can predict unseen journals.

The classification component addresses two problems: legitimacy detection and legitimacy criteria violation detection. For these two problems, we proposed using two different approaches for classification: traditional machine learning and deep learning. In the legitimacy detection problem, we defined classification as a binary classification problem, where we have two labels: legitimate and predatory. For the legitimacy criteria violation detection, we defined our problem as a multi-label classification problem, where, in the multi-label, we labels refer to legitimacy criteria.

For all our experiments, we used the paid version of Colab, which is Google Colab Pro+ which provides fast TPUs and GPUs, high RAM with 50GB, and a background execution feature for a specific time. We used Scikit-learn [71] for machine learning experiments, while Keras was used[10] for deep learning experiments.

1) LEGITIMACY DETECTION

Legitimacy detection is the process of classifying a journal as either predatory or legitimate. We defined this problem as a binary classification problem, where we have two labels that are legitimate or predatory. As previously mentioned, we used machine learning and deep learning approaches. These approaches are described in detail in the following subsections.

a: MACHINE LEARNING APPROACHES

We evaluated the SVM as a machine learning algorithm. We chose this model because Adnan showed that it is effective for classifying legitimacy. In addition, different studies have shown that SVM produces successful results in text classification and website classification [58], [61], and performs very well with sparse features [58].

Nine case studies of SVM were evaluated, and Table 9 presents the deployed case studies, experimented parameters, and the selected values. The evaluated case studies are as follows:1) SVM using n-gram and TF-IDF feature selection on the entire dataset, 2) SVM using n-gram and TF-IDF feature selection using only the first level of our dataset, 3) SVM using Word2Vec feature extraction with 100 vector size, 4) SVM using Word2Vec feature extraction with 300 vector size, 5) SVM using Doc2Vec feature extraction with 100 vector size, 6) SVM using Doc2Vec feature extraction with 300 vector size, 7) SVM using n-gram and TF-IDF feature selection for a balanced subset of the training set that contains 2730 journals per class, 8) SVM using TF-IDF and n-gram feature selection for a small balanced subset of the training set that contains 100 journals per class, 9) SVM using information gain and TF-IDF feature selection after removing the first words from the information gain list.

In the first case study, we aimed to explore the effect of using TF-IDF feature selection and n-gram, which encode some semantics of the language in the feature vector. The second case study is the same as the first, except that we wanted to explore the effect of using only the first level (home page) of a website. In the third, fourth, fifth, and sixth case studies, we explored the effect of using Word2Vec and Doc2Vec feature extraction techniques on our classification task, which incorporates syntactic, semantic, or polysemy of the words. We evaluated two vector sizes, 100 and 300, for the two feature-extraction techniques. Moreover, SVM only accepts a one-dimensional vector for every journal, whereas the Word2Vec vectors result in (mXn) vectors, where m is the length of the journal and n is the Word2Vec size (100 or 300). Therefore, we evaluated two aggregation methods for the Word2Vec vectors: summing and averaging the vectors. Thus, we combined the n resulting vectors into a one-dimensional vector of size (1Xn). In the seventh case study, we aimed to understand whether using the entire dataset instead of a balanced dataset could significantly affect the results. Finally, the eighth and ninth case studies were explored based on the baseline results and findings, as described in detail in the experimental results section.

We used 5-fold cross-validation on the training set for all case studies, with the F1-score as a scoring metric to record the results and choose the best value for parameters if needed. We also use a grid search to find the best values for our models. In the first, second, seventh, eighth, and ninth case studies, we experimented with different SVM parameters (kernel, regularization parameter ($\boldsymbol{C}$), and gamma), feature set size, and n-gram range. While in the third, fourth, fifth, and sixth case studies, we used the default parameters of the SVM of Adnan [19] because experimenting with Word2Vec or Doc2Vec consumes about 5-6 hours for a single experiment; hence, we were restricted by the time and available computational resources. However, for the two feature representation methods, Word2Vec and Doc2Vec, we evaluated two different feature vector sizes, as mentioned before, but we could not evaluate 500 for the vector size due to memory constraints.

b: DEEP LEARNING APPROACHES

As we want to experiment with legitimacy classification using different deep learning models, we evaluated five models: NNs, LSTM, CNN, BERT, and ALBERT. This section details the fine tuning of the hyperparameters of the models. Then, it presents the architectures of the used models,

[10]https://keras.io/

deployed case studies per model, and the tested and selected hyperparameters per case study.

Hyperparameter Tuning for Deep Learning Models: We used a grid search with 5-fold cross-validation and F1-score as a scoring metric on the training set for all the evaluated models to find the best hyperparameters. However, BERT is already trained and requires simple fine-tuning; hence, we did not use 5-fold cross-validation. The number of epochs was set to 20, and the early stopping technique provided by Keras was used. We selected 20 because we wanted to select the hyperparameters that would allow the model to learn quickly. After the model hyperparameters were selected, we increased the number of epochs to 100, along with early stopping, and allowed the model to train on the training set. We used binary cross-entropy as a loss function. However, owing to the limitations of time and computational resources, some of our model-specific hypermeters were set to default values, while some parameters related to the size of the data, such as input size, were set based on the memory size limit of 50 GB offered by Colab Pro +.

In all our deep learning experiments, we used the Hyperas[11] library to help us apply grid search and choose the best hyperparameters. In the Hyperas library, we provided hyperparameters space that we wanted to test, and it ran different experiments, and provided us with the best hyperparameters choices based on the F1-scoring function.

NNs Model: In this model, we evaluated different feature set sizes (500,700,1000,7000) for the TF-IDF, where we stopped at 7000 because half of the journals had approximately 7000 tokens. In addition, we tested two vector sizes for the Doc2Vec model: 100 and 300. We evaluated six case studies using this model:1) NNs using 500 as a feature set size for TF-IDF, 2) NNs using 700 as a feature set size for TF-IDF, 3) NNs using 1000 as a feature set size for TF-IDF, 4) NNs using 7000 as a feature set size for TF-IDF, 5) NNs using 100 as a vector size for Doc2Vec, and 6) NNs using 300 as a vector size for Doc2Vec.

For the NNs model, we evaluated different numbers of layers, different numbers of neurons, and different activation functions per layer. For the number of layers, we mean the intermediate layers, which are the hidden layer in NNs. In addition, we optimized the training hyperparameters, including the optimization algorithm, learning rate, and batch size. All these parameters were experimented with by grid-search through 5-fold cross-validation on the training set for every case study individually because every case study has a different input format or size; hence, it needs a specific hyperparameters configuration. Table 10 lists these ranges of tested hyperparameters and the selected value per case study for the NNs model.

LSTM Model: Six case studies were evaluated:1) LSTM using 500 as a feature set size for TF-IDF, 2) LSTM using 700 as a feature set size for TF-IDF, 3) LSTM using 1000 as a feature set size for TF-IDF, 4) LSTM using 7000 as a feature set size for TF-IDF, 5) LSTM using 100 as a vector size for Doc2Vec, and 6) LSTM using 300 as a vector size for Doc2Vec. Table 11 presents the ranges of the tested hyperparameters and the selected value per case study for the LSTM model.

CNN Model First Architecture: This model architecture is slightly different from that of previous models. This model was inspired by Li et al. in 2019 [47] for website classification. In this model, we evaluated six case studies:1) CNN using 500 as a feature set size for TF-IDF, 2) CNN using 700 as a feature set size for TF-IDF, 3) CNN using 1000 as a feature set size for TF-IDF, 4) CNN using 7000 as a feature set size for TF-IDF, 5) CNN using 100 as vector size for Doc2Vec, and 6) CNN using 300 as a vector size for Doc2Vec. Table 12 presents the ranges of the tested hyperparameters and the selected value per case study for the CNN model.

CNN Model Second Architecture: We used another CNN architecture inspired by Kim's research on text classification [45]. In this model, we have five layers: the input, convolutional, max-pooling, concatenation, and output layers. In this architecture, we have three parallel convolutional layers and three parallel max-pooling layers, where each convolutional layer captures different patterns from text.

Specifically, each convolutional layer has a different kernel size. Kim suggested using 3, 4, and 5 layers' kernel sizes and 100 filters per layer for text classification. In Kim's work, a layer with a kernel size of three captures patterns in sequential groups of three words, and the other layers work similarly. He selected five words as a cutoff because words farther away than five were generally less useful for identifying phrase patterns. We used an architecture similar to Kim's work [45], where we used three parallel convolutional and max-pooling layers. However, the length of our input data is different from that of Kim's work, where the average length of the sentences used was 23. Therefore, we experimented with different kernel and filter sizes per layer. In this model, we evaluated four case studies:1) CNN using Word2Vec with 3500 sequence lengths, 2) CNN using Word2Vec with 7000 sequence lengths, 3) CNN using 100 as a vector size for Doc2Vec, and 4) CNN using 300 as a vector size for Doc2Vec. For the first two cases, we evaluated Word2Vec with a vector size of 100. Owing to computational and memory constraints, we could not use all the scraped content, and we could not use the first 19,048 words, which specify the average length of 75% of the dataset; thus, we used the average length of 50% of the data, which is 6926–7000 words. Table 13 presents the ranges of the tested hyperparameters and the selected values per case study for the CNN model.

BERT Language Model: BERT is a recent transformer-based approach that achieved state-of-the-art results for different NLP tasks [62]. By choosing this model, we aim to understand how much the pre-trained BERT model can perform on the legitimacy detection task, which was pre-trained on a large corpus of sentences collected from BooksCor-

[11] https://github.com/maxpumperla/hyperas

TABLE 10. NNs tested and selected hyperparameters' values.

Hyperparameter	# Middle layers	# Neurons per layer	# Activation functions	Dropout per layer	Optimization algorithm	Learning rate	Batch size
Grid search hyperparameters	1,3,5	128, 256, 512, 1024, 2048	ReLu, sigmoid	Range (0,1)	Adam, SGD, RMSprop	0.0001, 0.001, 0.01, 0.1	16,32,64, 128
Values for case study 1: (300 TF-IDF)	1 layer	Layer1:2084	Layer1: ReLu	Layer1:0.02	RMSprop	0.01	32
Values for case study 2: (500 TF-IDF)	1 layer	Layer1:512	Layer1: ReLu	ayer1:0.31	RMSprop	0.01	32
Values for case study 3: (1000 TF-IDF)	5 layers	Layer1:256 Layer2:512 Layer3:2048 Layer4:2048 Layer5:512	Layer1: ReLu Layer2: ReLu Layer3: sigmoid Layer4: ReLu Layer5: ReLu	Layer1:0.59 Layer2:0.89 Layer3:0.55 Layer4:0.06 Layer5:0.16	Adam	0.0001	32
Values for case study 4: (7000 TF-IDF)	1 layer	Layer1:256	Layer1: ReLu	Layer1:0.01	RMSprop	0.01	32
Values for case study 5: (100 Doc2Vec)	5 layers	Layer1:256 Layer2:1024 Layer3:512 Layer4:1024 Layer5:1024	Layer1: ReLu Layer2: ReLu Layer3: ReLu Layer4: sigmoid Layer5: ReLu	Layer1:0.64 Layer2:0.005 Layer3:0.61 Layer4:0.33 Layer5:0.006	Adam	0.001	16
Values for case study 6: (300 Doc2Vec)	1 layer	Layer1:1024	Layer1: ReLu	Layer1:0.01	RMSprop	0.0001	16
Values for case study 5: (100 Doc2Vec)	**1 layer**	**Layer1:1024**	**Layer1: ReLu**	**Layer1:0.09**	**Adam**	**0.0001**	**32**
Values for case study 6: (300 Doc2Vec)	1 layer	Layer1:256	Layer1: ReLu	Layer1:0.12	Adam	0.0001	32

pus (800 million words) and English Wikipedia (2,500 million words) [62]. In our experiments, we fine-tuned the uncased version of BERT by adding a simple sigmoid classification layer that classifies journals as legitimate or predatory.

Because BERT accepts a maximum of 512 tokens per journal, we experimented with two input approaches. We used the same pre-processing technique as mentioned in the pre-processing component in the first approach. In the second one, punctuations and numbers were removed to increase the number of words that the BERT model could see. In this model, we evaluated two case studies:1) BERT using the original dataset, and 2) BERT using the original dataset after removing punctuations and numbers.

ALBERT Language Model: In addition to BERT, we evaluated ALBERT [73], which is a similar version to BERT but with fewer parameters and trained on the same dataset, and achieved better results than BERT in some tasks. However, the selected ALBERT model was pretrained on the OSCAR dataset.[12] We chose this because the OSCAR[13] dataset is similar to ours, where it is built from huge multilingual crawled

[12]https://huggingface.co/XSY/albert-base-v2-scarcasm-discriminator
[13]https://huggingface.co/datasets/oscar

TABLE 11. LSTM tested and selected hyperparameters' values.

Hyperparameter	# Middle layers	# Neurons per layer	# Activation functions	Dropout per layer	Optimization algorithm	Learning rate	Batch size
Grid search hyperparameters	1,3,5	128, 256, 512, 1024, 2048	ReLu, sigmoid, TanH	Range (0,1)	Adam, SGD, RMSprop	0.0001, 0.001, 0.01, 0.1	16,32,64 ,128
Values for case study 1: (300 TF-IDF)	3 layers	Layer1:1024 Layer2:128 Layer3:128	Layer1: ReLu Layer2: sigmoid Layer3: TanH	Layer1:0.72 Layer2:0.23 Layer3:0.49	Adam	0.01	64
Values for case study 2: (500 TF-IDF)	3 layers	Layer1:1024 Layer2:128 Layer3:128	Layer1: ReLu Layer2: sigmoid Layer3: TanH	Layer1:0.72 Layer2:0.23 Layer3:0.49	Adam	0.01	64
Values for case study 3: (1000 TF-IDF)	3 layers	Layer1:1024 Layer2:128 Layer3:128	Layer1: ReLu Layer2: sigmoid Layer3: TanH	Layer1:0.72 Layer2:0.23 Layer3:0.49	Adam	0.01	64
Values for case study 4: (7000 TF-IDF)	1 layer	Layer1:1024	Layer1: ReLu	Layer1:0.01	Adam	0.0001	32

webpages using Common Crawl. We believe that this model could provide satisfactory results.

We used the default hyperparameters values for fine-tuning the BERT and ALBERT models, except for the learning rate, batch size, and number of training epochs. We selected the best-performing values using a grid search on the training set, with F1 as a scoring metric. For the number of epochs, it was suggested to use 2,3,4 epochs; however, we used early stopping for 10 epochs to let the model train with the best number of epochs. Table 14 presents the tested and selected values for BERT and ALBERT hyperparameters.

2) LEGITIMACY CRITERIA VIOLATION DETECTION

Legitimacy criteria violation detection is the process of automatically providing a list of legitimacy criteria that a given questionable journal does not apply. We defined this problem as a multi-label classification problem, in which each label corresponds to one legitimacy criterion. Figure 5 depicts the architecture of our CNN model for our legitimacy criteria violation detection. The model's workflow is similar to the workflow explained before in the legitimacy detection task, except the output layer is configured to work with multi-label classification. Thus, the output layer is a fully connected sigmoid layer of (n) neurons, where n represents the number of criteria, which is 39 in our case. Moreover, we evaluated the model using Word2Vec feature representation with a vector size of 100 and sequence length of 7000.

CNN Model Second Architecture Hyperparameters Settings: This task differs from the first task (legitimacy classification), where we want the CNN model to capture the best feature maps for legitimacy classification. Thus, we fine-tuned the same parameters and hyperparameters again to select the best values for our multilabel classification task. We experimented with different kernel and filter sizes per layer. In addition, we experimented with the dropout rate, optimization algorithm, learning rate, and batch size. Table 15 lists the ranges of the tested and selected hyperparameters.

F. BASELINES AND EVALUATION METRICS

1) BASELINES

We used SVM and KNN as our baselines for the legitimacy detection task because Adnan proved that these classifiers are effective in the legitimacy classification task [19].

In addition, SVM performs well in text classification and website classification [61], and deals with sparse features [58]. For feature selection, we used information gain and TF-IDF, because Adnan also used these feature selection methods and proved to be effective in our domain [19]. However, we used SVM with TF-IDF and n-gram features selection methods as our baseline for the legitimacy criteria violations detection task, where the choices were based on the

TABLE 12. CNN first architecture tested and selected hyperparameters' values.

Hyperparameter	# Middle layers	# Filters	Kernel size	# Activation function	Dropout per layer	Optimization algorithm	Learning rate	Batch size
Grid search hyperparameters	1	128, 256, 512, 1024	2, 3, 5, 10,20,100	ReLu, sigmoid, TanH	Range (0,1)	Adam, SGD, RMSprop	0.0001, 0.001, 0.01, 0.1	16, 32, 64, 128
Values for case study 1: (300 TF-IDF)	1 layer	256	20	TanH	0.03	RMSprop	0.001	16
Values for case study 2: (500 TF-IDF)	1 layer	256	20	TanH	0.06	RMSprop	0.001	16
Values for case study 3: (1000 TF-IDF)	1 layer	128	100	TanH	0.10	Adam	0.001	64
Values for case study 4: (7000 TF-IDF)	1 layer	256	100	TanH	0.24	Adam	0.001	16
Values for case study 5: (100 Doc2Vec)	1 layer	128	20	TanH	0.02	Adam	0.001	64
Values for case study 6: (300 Doc2Vec)	1 layer	128	20	TanH	0.10	Adam	0.001	64

experiments' results of the legitimacy detection task. We used the Scikit-learn library [71] to implement both models.

Legitimacy Detection Task Baseline: SVM Hyperparameter Settings and Feature Set Size. SVM classifier has several hyperparameters that need to be optimized: kernel, regularization parameter (***C***), and gamma. We used the same settings as Adnan's study, as shown in Table 16.

However, as our dataset source and size differ from Adnan's dataset, which contains only 200 journals, we experimented with different value ranges to select the feature set size. We used grid search for F1-score using 5-fold cross-validation on the training set to select the feature set size.

Legitimacy Detection Task Baseline KNN Hyperparameter Settings and Feature Set Size. In the KNN classifier, we optimized two parameters: K and the feature set size. We similarly used a grid search for the F1-score using 5-fold cross-validation on the training set to obtain the best value for K and feature set size. The tested and selected values are listed in Table 17.

Legitimacy Criteria Violations Detection Task Baseline SVM Hyperparameter Settings and Feature set Size.

For the SVM classifier in the legitimacy criteria detection task, we utilized the "MultiOutputClassifier"[14] provided by the Scikit-learn library [71] because the SVM classifier does not support multi-label classification. MultiOutputClassifier enables multilabel classification with SVM by fitting one classifier per label.

We optimized four parameters which are regularization parameter (***C***), gamma, feature set size for the TF-IDF vectors, and n-gram ranges. The experimented and selected values are presented in Table 18. We used grid search for

[14]https://scikit-learn.org/stable/modules/generated/sklearn.multioutput.MultiOutputClassifier.html

TABLE 13. CNN's second architecture tested and selected hyperparameters' values.

Hyperparameter	# Filters per layer	Kernel size per layer	Dropout per layer	Optimization algorithm	Learning rate	Batch size
Grid search hyperparameters	50, 100, 200	20, 30, 50, 100	Range (0,1)	Adam, SGD, RMSprop	0.0001, 0.001, 0.01, 0.1	16, 32, 64, 128
Values for case study 1: (TF-IDF and using 3500 as a sequence length)	Layer1: 100 Layer2: 100 Layer3: 200	Layer1: 50 Layer2: 100 Layer3: 30	Layer1: 0.52 Layer2: 0.74 Layer3: 0.13	SGD	0.01	32
Values for case study 2: (TF-IDF and using 7000 as a sequence length)	Layer1: 200 Layer2: 50 Layer3: 50	Layer1: 20 Layer2: 20 Layer3: 20	Layer1: 0.57 Layer2: 0.28 Layer3: 0.39	RMSprop	0.001	32
Values for case study 5: (100 Doc2Vec)	Layer1: 100 Layer2: 50 Layer3: 200	Layer1: 100 Layer2: 50 Layer3: 50	Layer1: 0.39 Layer2: 0.90 Layer3: 0.30	Adam	0.001	32
Values for case study 6: (300 Doc2Vec)	Layer1: 50 Layer2: 200 Layer3: 200	Layer1: 30 Layer2: 30 Layer3: 50	Layer1: 0.67 Layer2: 0.79 Layer3: 0.01	RMSprop	0.001	32

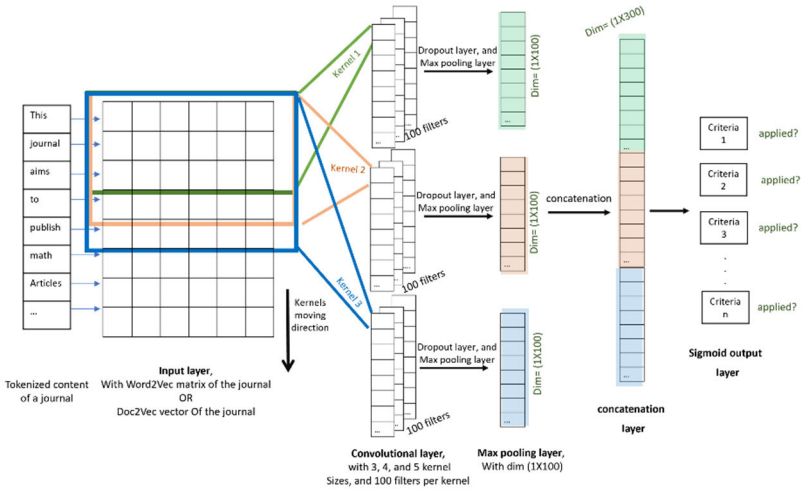

FIGURE 5. CNN model architecture 2 for legitimacy criteria detection.

F1-score using 5-fold cross-validation on the training set to fine-tune the parameters.

2) EVALUATION METRICS

Our research has two different classification problems: legitimacy classification (binary classification) and the detection of violated criteria (multi-label classification). Hence, we used two different evaluation metrics for each classification problem. However, to summarize and visualize the performance of our classification algorithms, we used a confusion matrix to report the results for the two classification problems.

a: METRICS OF LEGITIMACY CLASSIFICATION

To evaluate legitimacy classification, we calculated true negatives (TN) (predatory journals classified as predatory), true positives (TP) (legitimate journals classified as legitimate), false negatives (FN) (legitimate journals classified as

TABLE 14. BERT and ALBERT tested and selected hyperparameters' values.

Hyperparameter	Range	Selected value
Learning rate	5e-5, 3e-5, 2e-5	2e-5
Batch size	8, 16, 32	16
Number of epochs	Early stopping	Early stopping

TABLE 15. CNN architecture 2 tested and selected hyperparameters' ranges for legitimacy criteria violations detection task.

Hyperparameter	Range	Selected value
# Layers	3 parallel layers	3 parallel layers
# Filters	50, 100, 200	Layer1: 200 Layer2: 50 Layer3: 50
# Kernel size	20, 30, 50, 100	Layer1: 100 Layer2: 20 Layer3: 30
# Activation function	ReLu	ReLu
Dropout per layer	Range (0,1)	Layer1: 0.34 Layer2: 0.33 Layer3: 0.36
Optimization algorithm	Adam, SGD, RMSprop	RMSprop
Learning rate	0.0001, 0.001, 0.01, 0.1	0.0001
Batch size	16,32,64,128	16
Number of epochs	Early stopping	Early stopping

TABLE 16. Baselines: SVM Hyperparameter settings and feature set size for the legitimacy detection task.

Hyperparameter	Range	Selected value
kernel	rbf	rbf
regularization parameter (*C*)	1	1
gamma	1/num_features	1/num_features
Feature set size	5, 50, 100, 1000, 2000, 3000, …, 100000	5

TABLE XVII

TABLE 17. Baselines: KNN Hyperparameter settings and feature set size.

BASELINES: KNN HYPERPARAMETER SETTINGS AND FEATURE SET SIZE

Hyperparameter	Range	Selected value
K (*n_neighbors*)	[1,3,5,10,20,30,50]	10
Feature set size	5, 50, …, 100000	5

predatory), and false positives (FP) (predatory journals classified as legitimate) by comparing the predicted and actual labels. We then use the F1 measure, as in the following

TABLE 18. Baselines: SVM Hyperparameter settings and feature set size for legitimacy criteria violation detection task.

Hyperparameter	Range	Selected value
kernel	rbf	rbf
regularization parameter (*C*)	1, 100, 1000	1000
gamma	10,1,0.1, 0.01, 0.001	10
Feature set size	500, 3000, 10,000	10,000
n-gram range	(1,1), (1,3), (1,5)	(1,1)

equations:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{2}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{3}$$

$$\text{F1} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{4}$$

b: METRICS OF LEGITIMACY CRITERIA' VIOLATIONS DETECTION

Because our dataset is unbalanced and every journal has applied labels from 0 to 39 labels, we used the micro F1 score and weighted F1 score to evaluate the performance of this task.

For each sample-label pair, we calculated the following: true negatives (TN), true positives (TP), false negatives (FN), and false positives (FP) by comparing predicted and actual labels, where labels are binary (0 and 1) indicating whether or not a journal applies a specific criterion. Then, we calculated the micro-precision and micro-recall as in (2) and (3). Subsequently, we calculate micro-F1 from micro-precision and micro-recall, as shown in (1). Thus, we can see that micro-F1 gives each sample-label pair an equal contribution to the overall metric.

In contrast, the weighted-F1 score gives each label/criterion a weighted contribution to the overall metric, based on its support. The weighted-F1 score is calculated by first calculating F1 per label/class, as shown in (1). Then, we calculate the mean of all per-label F1 scores weighted by their label's support. Support refers to the number of actual label occurrences in the dataset.

VI. RESULTS AND DISCUSSION

This section presents the results of our baselines and the evaluated machine learning and deep learning models for legitimacy classification and legitimacy criteria violation detection tasks.

A. LEGITIMACY DETECTION

Table 19 and Table 20 summarize the case studies' best results achieved by our evaluated machine-learning and deep-learning models in terms of F1 score. Because neural networks are stochastic in nature and different results can be reported in different runs, we reported the results of the aver-

TABLE 19. Machine learning evaluation results of the evaluated models in the legitimacy detection task.

Model	Training data	Feature representation	Test set (F1)
SVM (baseline)	Full training set	Information gain + TF-IDF	0.93
KNN (baseline)	Full training set	Information gain + TF-IDF	0.93
SVM	Full training set	n-gram + TF-IDF	**0.95**
SVM	Full training set (First level of the dataset)	n-gram + TF-IDF	0.92
SVM	Full training set	Word2Vec with 100 as a vector size (sum of Word2Vec vectors)	0.00
SVM	Full training set	Word2Vec with 100 as a vector size (mean of Word2Vec vectors)	0.90
SVM	Full training set	Word2Vec with 300 as a vector size (sum of Word2Vec vectors)	0.00
SVM	Full training set	Word2Vec with 300 as a vector size (mean of Word2Vec vectors)	0.88
SVM	Full training set	Doc2Vec with 100 as a vector size	0.78
SVM	Full training set	Doc2Vec with 300 as a vector size	0.89
SVM	a subset of the training set contains 2730 journals per class	n-gram + TF-IDF	**0.95**
SVM	a subset of the training set contains 100 journals per class	n-gram + TF-IDF	**1.00**
SVM	Full training set	Information gain + TF-IDF (removing first five words from information gain list)	0.78

TABLE 20. Deep learning evaluation results of the evaluated models in the legitimacy detection task.

Model	Features	Test set (F1)
Neural Network	300 TF-IDF features	0.93
	500 TF-IDF features	0.94
	1000 TF-IDF features	**0.95**
	7000 TF-IDF features	0.93
	doc2vec, 100 dimensions	0.93
	doc2vec, 300 dimensions	0.94
LSTM	300 TF-IDF features	**0.95**
	500 TF-IDF features	**0.95**
	1000 TF-IDF features	**0.95**
	7000 TF-IDF features	0.93
	doc2vec, 100 dimensions	0.92
	doc2vec, 300 dimensions	0.94
CNN first architecture	300 TF-IDF features	**0.95**
	500 TF-IDF features	**0.95**
	1000 TF-IDF features	**0.95**
	7000 TF-IDF features	0.94
	Doc2vec 100 dimensions	0.90
	doc2vec, 300 dimensions	0.91
CNN second architecture	3500 Word2Vec	**0.96**
	7000 Word2Vec	**0.96**
	Doc2vec 100 dimensions	0.90
	Doc2vec 300 dimensions	0.91
BERT	First 512 tokens	0.91
	First 512 tokens after removing punctuations and numbers	0.92
ALBERT pre-trained on the OSCAR dataset	First 512 tokens	0.92

age of 5-fold cross-validation. We limited it to five because of the computational and time constraints of the project.

1) BASELINE PERFORMANCE COMPARED TO ADNAN's STUDY

As mentioned previously, we evaluated SVM and KNN on our dataset using the same settings as in Adnan's study. We did not perform parameter selection except for the feature set size extracted using information gain because our dataset is different from Adnan's dataset. In the experiments, the two models achieved a good and similar F1 score of 0.93 on our test dataset, as presented in Table 19. However, we noticed that the selected feature set size was only the first five words, and these five words are: '2021,' '2020,' '2019,' '1018,' and 'HTTPS.' We reviewed to the criteria to understand why just these five words are selected and why they give a high information gain for our legitimacy classification task. We found the following related criterion: ''The journal's website has no past or recent journal content, ''from the 'website' criterion. In other words, we can say that these selected words can refer to the last years of the dataset collection, where if a journal does not provide recent content, then it could be predatory. For the 'HTTPS' word, we can find that the website's security may affect the final classification results.

Comparing our results with those of Adnan, Adnan achieved better results for both SVM and KNN, where the SVM F1 score was 0.98, KNN F1 score was 0.94, and the feature set sizes were 300 for SVM and 4000 for KNN. However, we have several points according to Adnan's experiment. First, the reported results were based on 10-fold cross-validation of the entire dataset without performing a blind test, which could result in overfitting of the dataset. The second point is that the dataset source for predatory journals is Beall, while the source of legitimate journals is DOAJ; hence, two different dataset sources for predatory and legitimate journals are used. While we extracted our dataset from two different sources: Beall and DOAJ for the predatory journals and DOAJ for the legitimate journals; hence, the dataset sources for predatory and legitimate classes are partially similar. The third point is the size of the dataset, where Adnan used a total of 200 journals, which is a small sample, while we used a total of 6836 journals. Therefore, to explore the performance of our model on the same dataset size, we evaluated our SVM model (using TF-IDF and n-gram) using 200 journals randomly. The fourth and last point is the pre-processing of the dataset, where Adnan said that he had just performed stop words removal; however, we did not see any number in their list of the first 20 words that were extracted using information gain.Hence, we performed other experiments after removing the first five words as they were the most effective words in the classification, and we left the numbers as we wanted to know if they impacted the results. We summarize our findings for **Baselines performance compared to Adnan's study** in the following points.

1) Our baselines (SVM and KNN) achieved good and similar F1 scores of 0.93.
2) Adnan achieved better results than our baselines for both SVM and KNN, but we have the following considerations regarding his study:
3) Adnan did not perform a blind test,
4) Two different dataset sources for predatory and legitimate journals are used,
5) Adnan used a total of 200 journals, which is a small sample, while we used a total of 6836 journals, and
6) The pre-processing of the dataset is not clear.

2) SVM MACHINE LEARNING ALGORITHM RESULTS

As presented in Table 19, we evaluated different case studies using the SVM model, where each case study had a different feature representation or dataset. The worst result was the 0.0 F1 score, which was achieved using the summation method for Word2Vec vectors. As mentioned previously, we tried this approach because it yielded good results compared to TF-IDF [60]. However, we believe that it did not produce good results because our dataset and task are different from those in [60]. We had an average of 7000 words, where [60] had between 93 and 1263 words averaged by class, and some of our dataset's words come from very different fields in science as we have journals from different fields of science, and we had about 3,003,770 words, while [60] had 29,930 words. Hence, the meaning of a word can be lost when we use summation.

On the other hand, averaging the word vectors solved the problem and achieved better results (excluding the experiments on a subset of our dataset). The best-achieved F1 score was 0.95 with SVM with TF-IDF and bigrams. SVM achieved an F1 score of 0.90 using the 100 Word2Vec feature extraction method. This was followed by SVM with 300 Doc2Vec features extraction with an F1-score of 0.89. However, the two feature representations, 300 Word2Vec and 100 Doc2Vec, did not significantly differ from 100 Word2Vec and 300 Doc2Vec feature representations. In addition, both baseline models, which used information gain with TF-IDF feature selection, achieved 0.93, which is worse than our SVM model with TF-IDF and bi-gram.

One of our case studies was conducted using a small dataset size, where we used 200 journals like Adnan. The F1 score achieved using 200 journals was 1.0, whereas Adnan achieved 0.98 on the same portion of the dataset. This result shows that the dataset size affects our classification task, and using only 200 journals as a dataset can allow the model to overfit the dataset. Another case study was conducted after removing the first five words extracted from the information gain list. We conducted this experiment because only five words were selected. In this case study, 50 words were selected, with an F1 score of 0.78. The 0.78 F1 score is less than the baseline by 0.15 points, which emphasizes the importance of these words in the classification process, as long as the information gain and TF-IDF feature selection are used with the SVM model. We replicated the same experiment using Word2Vec, with a vector size of 100. The achieved classification F1-score was 0.88, which is less than that of

Word2Vec on the whole dataset by only 0.02 points. This indicates that Word2Vec does not rely too much on these five words in the classification process as the information gains, and it can extract helpful syntax and semantics from a journal's words.

Moreover, one of our case studies was conducted to explore whether we can perform the classification using only the first level of a journal website (Home Page). This experiment was performed because of the scraping cost in terms of time, complexity, and fees resulting from the scraping process. In this experiment, SVM achieved an F1 score of 0.92, which is close to the baseline result of 0.93, and the results may be better if different features or models are used.

We summarize our findings for **SVM machine learning algorithm results** in the following points:

1- The best-achieved F1 score was 0.95 with SVM with TF-IDF and bigrams.
2- Our SVM baseline with information gain and TF-IDF feature selection achieved 0.93, which is worse than that of our SVM model with TF-IDF and bi-gram.
3- Word2Vec is slightly better than Doc2Vec's feature representation.
4- The summation method for Word2Vec vectors had the worst result of 0.0 F1 score.
5- The meaning of a word could be lost when we used a summation if the length of the documents was very large.
6- The dataset's size affects our classification task.
7- Word2Vec can extract helpful syntax and semantics from a journal's words.
8- Using only the first level of a journal website to make the classification insufficient.

3) DEEP LEARNING RESULTS

As shown in Table 20, four different deep learning models were evaluated: NNs, LSTM, CNN first architecture, and CNN second architecture. The first three models were evaluated with two feature representations: TF-IDF feature selection and Doc2Vec feature extraction, while the fourth model was evaluated using two feature representations: Word2Vec and Doc2Vec feature extraction. The deep learning models achieved promising results, where the worst F1 score was 0.90, with CNN using Doc2Vec 100 dimension, and the best one was 0.96, which was achieved using the second architecture of the CNN model with Word2Vec feature representation for both 3,500 and 7,000 sequence lengths. The Word2Vec feature selection method was better than both TF-IDF and Doc2vec for the CNN model. This could be due to the ability of Word2Vec to capture the syntax and semantic features of words. Thus, the differences in the syntax between predatory and legitimate journals were captured, and the semantics of journals, such as policies, were also captured.

For the TF-IDF feature selection, we experimented with all the evaluated models with different feature set sizes of 300, 500, 1000, and 7000. LSTM and CNN first architecture achieved their best F1 score of 0.95 on all the sizes except with a feature set size of 7000, which had F1 scores of 0.93 and 0.94 for LSTM and CNN, respectively. While the NNs achieved the best F1 score of 0.59 with a feature set size of 1000, and it achieved 0.93 and 0.94 with the other sequences. However, for all models, the difference between the results of different feature set sizes differed slightly. Thus, we can conclude that the models can learn to classify using a small size for the TF-IDF feature set. In addition, we can conclude that the process of detecting predatory and legitimate journals can be achieved by using the journals' textual content with a small TF-IDF feature set of approximately 1000 words.

For Doc2Vec feature selection, NNs and LSTM achieved their best F1 score of 0.94 with 300 Doc2Vec. In contrast, the CNN model achieved the worst results of 0.91 F1 score, which was even worse than the baseline. This could be because CNN tries to learn relations between words, such as n-gram, but Doc2Vec will impact this feature by converting words into Doc2Vec vectors. Moreover, the TF-IDF feature selection was better than the Doc2Vec feature extraction for all models. We can conclude that because Doc2Vec attempts to extract the meaning of a given text, it loses the writing behaviors that we try to capture. In addition to writing behaviors, we try to find the details of the policies (every word could matter) to classify a journal; however, Doc2Vec can lose them when it attempts to convert all the journal content into one (100 or 300) vector.

We summarize our findings for **deep learning results** in the following points:

1- The best F1 score was 0.96 using the second architecture of the CNN model with Word2Vec feature representation for both 3,500 and 7,000 sequence lengths.
2- The worst F1 score was 0.90 with CNN using Doc2Vec 100 dimension.
3- The Word2Vec feature selection method was better than both TF-IDF and Doc2vec for the CNN model.
4- Word2Vec was able to capture the syntax and semantics features of words.
5- For all models, the results of different feature set sizes differed slightly.
6- The process of detecting predatory and legitimate journals could be achieved by using the journals' textual content with a small TF-IDF feature set of approximately 1000 words.
7- The TF-IDF feature selection was better than the Doc2Vec feature extraction for all models.
8- Doc2Vec loses the writing behaviors that we tried to capture and lost the details of the policies when it tries to convert all the journal content into one vector.

4) BERT AND ALBERT RESULTS

For the BERT model, we evaluated two different types of input features, as shown in Table 20. In the first one, we used the original text, while in the second one, we removed the punctuations and numbers. However, we have used ALBERT

along with the original text. As presented in Table 20, the second model of BERT and ALBERT achieved an F1 score of 0.92, whereas the first model achieved 0.91. The results of the evaluated models show a slight drop (approximately 1-2%) in terms of the F1 score compared to the baselines. Compared with other deep learning models, BERT and ALBERT failed to improve the classification results. The results were expected even they achieved state-of-the-art results in many NLP tasks [62], [73]. We reason that the small number of tokens that the models can use in the classification process cannot include all the required information (such as policies) to make an accurate prediction. However, BERT and ALBERT used the knowledge gained in the pre-training process and only 512 tokens to classify a journal with F1 scores of 0.91 and 0.92, which can be accepted as a first-time experiment in our domain.

We summarize our findings for **BERT and ALBERT results** in the following points:

1- The results of the evaluated models show a slight drop (approximately 1-2%) in terms of the F1 score compared to the baselines.
2- BERT and ALBERT failed to improve the classification results compared with the other deep learning models.
3- The small number of tokens that BERT and ALBERT can use in the classification process adversely affects the results.

5) MACHINE LEARNING AND DEEP LEARNING RESULTS

Based on the previous results, we can conclude that a journal's textual content can be used to diagnose predatory and legitimate journals. Moreover, the neural networks were able to deal with the same publisher problem, where they did not rely directly on the publisher to make their predictions, which we can see in the training and testing results. Additionally, the second CNN architecture with Word2Vec achieved 0.96, which is the best F1 score among all the evaluated machine learning and deep learning models. We believe this result is a good starting point in our legitimacy classification task. We believe that other models may achieve better results. However, we believe that we could not achieve a value higher than 0.96 because the text of the journals cannot detect behavioral criteria. An example of a behavioral criterion is whether editorial board members accept the research without performing a peer review. The journal content does not provide any information about this criterion, as it depends on a person's behavior. Moreover, some journals may be similar to legitimate ones, but they are not where they have many policies, but they do not apply them. In addition, we know that some journals provide their policies using attached files, but in our experiments, we excluded them because of computational limitations.

To summarize and visualize the results of our classifiers, we used a confusion matrix. In Table 21 and Table 22, we present the results of the best experiment per classification model based on F1 results. However, experiments performed on a part of our dataset were not included. As Table 21 shows, SVM using n-gram and TF-IDF feature representation was the best compared to other evaluated machine learning models, and was better than our baselines. For the deep learning models, the second architecture of CNN was better than other models including deep learning and machine learning models in terms of true positive, true negative, false positive, and false negative.

TABLE 21. Confusion matrix results of the evaluated machine learning models in the legitimacy detection task.

Model	True Positive	False Positive	True Negative	False Negative
SVM with Information gain + TF-IDF **(baseline)**	494	24	796	54
KNN with Information gain + TF-IDF **(baseline)**	499	27	**793**	49
SVM with n-gram + TF-IDF	**512**	**24**	796	**36**

B. LEGITIMACY CRITERIA VIOLATIONS DETECTION

Table 23 summarizes the results achieved by our evaluated SVM and CNN models in terms of the micro and weighted F1 scores. The SVM model was evaluated with TF-IDF and unigram, while the CNN's second architecture was evaluated along with the Word2Vec feature selection. Both models were selected because they were the best machine and deep learning models for legitimacy classification.

We reported the results of the average of 5-folds cross-validation for the CNN because neural networks are stochastic in nature, and different results can be reported in different runs. We limited the number to five because of the computational and time constraints of the project.

As Table 23 shows, in terms of micro-F1, our SVM baseline was better than that of the CNN model, with 6% better results. In terms of weighted-F1, both models achieved similar results, with a difference of 1% for the SVM model. The results show that the SVM as a machine learning model is better than the CNN model as a deep learning model for our multi-label classification problem. However, SVM could be better than CNN because we used SVM with the MultiOutputClassifier library for multi-label classification, where this library fits one classifier per label/criteria. Thus, the model learns the best parameters per label/criterion independently from other labels/criteria. In contrast, the CNN model tries to learn the best parameters while simultaneously observing all the labels/criteria. Generally, the results were poor, which could be due to the small sample size of some labels.

TABLE 22. Confusion matrix results of the evaluated Deep learning models in the legitimacy detection task.

Model	Features	True Positive	False Positive	True Negative	False Negative
Neural Network	1000 TF-IDF features	520	31	789	28
	doc2vec, 300 dimensions	516	36	784	32
LSTM	300 TF-IDF features	522	29	791	26
	500 TF-IDF features	526	37	783	22
	1000 TF-IDF features	527	46	774	21
CNN first architecture	300 TF-IDF features	522	34	786	26
	500 TF-IDF features	515	20	800	33
	1000 TF-IDF features	519	25	795	29
CNN second architectur	3500 Word2Vec	**527**	28	792	**21**
	7000 Word2Vec	513	**7**	**813**	35
BERT	First 512 tokens after removing punctuations and numbers	519	59	761	29

TABLE 23. Evaluation results of the evaluated models in the legitimacy criteria violations detection task.

Model	Micro-F1	Weighted-f1
SVM with TF-IDF and unigram (baseline)	0.67	0.61
CNN's second architecture with Word2Vec	0.61	0.60

We thoroughly investigated the score results per criterion by using a confusion matrix to understand how the classifiers worked in the legitimacy criteria violations detection task and why the classifiers misclassified the criteria and produced low results scores.

Table 24 presents the evaluation results for the SVM and CNN models in the legitimacy-criteria violation detection task. This table details the score results for true positives, false positives, true negatives, false negatives, precision, recall, and F1 score per criterion. A true positive means that the classifier correctly detected that the criterion is applied by the journal, whereas a true negative means that the classifier correctly detected that the criterion was not applied by the journal. False positive and false negative indicate that the classifier failed to detect whether the journal applied or did not apply a given criterion, respectively.

As we saw, the SVM outperformed the CNN model in terms of true positives and false negatives with scores of 4201 and 1717, respectively, which are better than the CNN model by 5%. Additionally, the SVM outperformed the CNN model in terms of micro F1 and weighted F1 with scores of 0.67 and 0.61, respectively, which are better than the CNN model by 6% and 1%, respectively. However, the results of both the SVM and CNN were considered low; thus, we performed this analysis to better understand the results.

The results presented in Table 24 show that the SVM model achieved low F1 scores (less than 0.23) on 13 criteria (highlighted in red), whereas the remaining criteria were above 0.63. Unlike the CNN classifier, which achieved low F1 scores (less than 0.47) on 13 criteria, the remaining criteria were above 0.52%. Moreover, the CNN model was slightly better than the SVM in the 13 highlighted criteria. The SVM model achieved better results than the CNN model for the remaining criteria (not highlighted in red). These results can be concluded from the macro F1 score, where the CNN achieved a better F1 score of 0.56, which is above the SVM by 2%. The macro F1 score was calculated by averaging the F1 score per label/criterion. However, the SVM model was similar to the CNN model in terms of weighted precision, and better in terms of the remaining measures.

These results led us to investigate why the SVM had low scores in some criteria (highlighted in red) and why the SVM model tended to predict the applied criteria precisely and recalled most of the applied criteria for a given journal.

To answer these questions, we first analyzed the reason for the low precision and recall scores for the criteria using the SVM model. From

Table 24, we can see that all 13 criteria (highlighted in red) with low scores (less than 0.23) have low support (few training examples), with 804 or fewer examples. We referred to the number of journals that applied the criteria, where the average number of journals that applied a specific criterion was 851, the minimum number was 460, and the maximum number was 1296. In addition to SVM, CNN achieved its lowest scores when there were few training examples for the criteria.

Additionally, we analyzed why the CNN model was slightly better than the SVM in the 13 highlighted criteria than the SVM model. We believe that this is because the CNN was not restricted by the available examples of the targeted criterion as the SVM model did, where the SVM model treated each criterion independently. Thus, the CNN model was better in the highlighted criteria because it was able to use other criteria and examples to make predictions for the targeted criteria. Ultimately, this could be due to the

TABLE 24. Confusion matrix results of the evaluated deep learning models in the legitimacy criteria detection task.

		CNN							SVM							Support
#	Criteria definition	p	R	f1	TP	FP	TN	FN	P	R	f1	TP	FP	TN	FN	
1	The journal website contains (Aims & Scope) statements th at are clearly defined	0.6	0.6	0.6	58	36	126	65	0.68	0.14	0.23	57	21	141	66	681
2	The journal website contains a peer- review model or process	0.54	0.56	0.55	91	57	60	77	0.60	0.90	0.72	111	64	53	57	871
3	The journal website contains a peer-review statement	0.76	0.85	0.8	157	42	37	49	0.72	0.96	0.82	201	68	11	5	1089
4	The journal guarantees manus cript acceptance or very short peer-review times	0.34	0.13	0.19	14	23	177	71	0.20	0.02	0.04	4	8	192	81	483
5	The journal has policies on jo urnal options for post- publication discussions and c orrections that are clearly visi ble on its website	0.55	0.56	0.56	83	59	79	64	0.51	0.93	0.66	112	97	41	35	867
6	The way(s) in which content i s available to readers is stated (written explicitly)	0.72	0.89	0.8	171	76	12	26	0.70	0.97	0.81	194	85	3	3	1120
7	The journal has policies on au thorship and contributorship t hat are clearly visible on its w ebsite	0.73	0.7	0.71	144	59	38	44	0.66	0.94	0.77	170	70	27	18	1010
8	The journal has policies on co nflicts of interest/competing i nterests that are visible on its website	0.5	0.43	0.46	47	55	110	73	0.48	0.09	0.15	21	28	137	99	628
9	The journal has policies on ho w the journal will handle com plaints and appeals that are cl early visible on its website	0.3	0.3	0.3	36	43	154	52	0.67	0.11	0.19	2	1	196	86	485
10	The publishing schedule is cle arly stated	0.81	0.95	0.88	219	51	2	13	0.81	0.96	0.88	231	51	2	1	1296
11	The available journal content appears as stated in the publis hing schedule	0.46	0.15	0.22	42	76	128	39	0.57	0.05	0.09	1	2	202	80	460
12	The journal has clearly indicat ed a long- term way for electronic backu p and preservation of access t o the journal content	0.5	0.38	0.43	117	105	28	35	0.55	0.95	0.70	117	107	26	35	853
13	The journal has a publishing l icense	0.6	0.75	0.67	95	74	48	68	0.56	0.94	0.70	153	116	6	10	976
14	The journal has policies and n otices of copyright	0.65	0.74	0.69	131	63	35	56	0.65	0.93	0.76	179	92	6	8	1005
15	The journal has a user license	0.62	0.67	0.64	128	90	29	38	0.57	0.85	0.68	102	69	50	64	832
16	The journal stated policies on posting final accepted version s or published articles on third -party repositories	0.69	0.6	0.64	132	61	29	63	0.69	0.95	0.80	191	88	2	4	1024
17	It is clear that If authors are al lowed to publish under a Crea tive Commons license, then a ny specific license requireme nts shall be noted	0.61	0.78	0.68	117	85	34	49	0.58	0.90	0.71	129	95	24	37	899
18	The journal displays the copyr ight holder (owner) on manus cripts	0.33	0.26	0.29	31	53	133	68	0.46	0.06	0.11	1	3	183	98	561
19	The journal is displaying adve rtisements on manuscripts	0.41	0.49	0.45	51	71	98	65	0.56	0.08	0.14	6	5	164	110	630
20	The journal has policies on da ta sharing and reproducibility that are clearly visible on its website	0.6	0.85	0.7	135	91	27	32	0.57	0.92	0.70	160	111	7	7	991
21	The manuscripts are available on the journal's website	0.71	0.9	0.79	171	73	12	29	0.71	0.97	0.82	198	85	0	2	1126
22	The editor-in-chief (highest- ranking editor) and the owner/ publisher are the same	0.44	0.34	0.38	63	67	99	56	0.62	0.13	0.21	24	26	140	95	699

TABLE 24. *(Continued.)* **Confusion matrix results of the evaluated deep learning models in the legitimacy criteria detection task.**

23	The journal has information about the ownership and/or management	0.65	0.36	0.46	148	82	25	30	0.63	0.96	0.76	168	96	11	10	995
24	The journal provides the full names and affiliations of its main editors (editor-in-chief, executive editor, managing editor, assistant editor…etc.)	0.68	0.68	0.68	128	61	45	51	0.63	0.96	0.76	170	94	12	9	1026
25	The journal provides the full names and affiliations of its general editors (editorial board)	0.66	0.72	0.69	134	68	27	56	0.67	0.96	0.79	182	86	9	8	1046
26	The journal provides full contact information for the editorial office (email, phone, address)	0.51	0.63	0.56	90	119	38	38	0.39	0.13	0.20	68	72	85	60	804
27	Policies on the ethical conduct of research using human subjects are clearly visible on the journal's website	0.38	0.35	0.37	22	42	140	81	0.35	0.07	0.11	7	12	170	96	588
28	Policies on the ethical conduct of research using animals are clearly visible on the journal's website	0.45	0.6	0.52	26	36	157	66	0.50	0.08	0.13	9	6	187	83	513
29	Policies on plagiarism are clearly visible on the journal's website	0.64	0.73	0.68	130	64	38	53	0.63	0.94	0.76	169	97	5	14	980
30	Policies on the citation are clearly visible on the journal's website	0.67	0.77	0.72	108	65	41	71	0.63	0.96	0.76	165	93	13	14	979
31	Policies on data falsification/fabrication are clearly visible on the journal's website	0.52	0.25	0.34	91	81	51	62	0.54	0.93	0.68	134	119	13	19	950
32	Policies on the consent of publication are clearly visible on the journal's website	0.42	0.91	0.57	56	86	84	59	0.35	0.08	0.13	43	51	119	72	750
33	The journal states its advertising policy, including what types of adverts will be considered	0.57	0.66	0.61	100	90	46	49	0.53	0.91	0.67	89	89	47	60	844
34	The journal states its advertising policy, including whether the adverts are linked to content or reader behavior or are displayed at random	0.52	0.62	0.57	68	62	71	84	0.54	0.89	0.67	111	99	34	41	857
35	The journal states its advertising policy, including who makes decisions regarding accepting adverts	0.52	0.44	0.47	90	88	53	54	0.50	0.87	0.63	51	49	92	93	817
36	The advertisements are not related in any way to editorial decision making	0.41	0.32	0.36	36	63	125	61	0.39	0.07	0.12	9	9	179	88	590
37	The journal website states business models, business partnerships/agreements, or revenue sources otherwise evident on the journal's website.	0.5	0.62	0.55	53	54	74	104	0.57	0.96	0.71	123	99	29	34	881
38	The journal website state how much fees or charges are required for manuscript processing and/or publishing materials in the journal	0.58	0.62	0.6	87	60	67	71	0.56	0.94	0.71	107	84	43	51	880
39	The manuscript's associated costs are stated on the journal's website	0.72	0.77	0.74	148	64	25	48	0.70	0.97	0.81	193	89	0	3	1110
	total	-	-	-	3709	2556	2641	2209	-	-	-	4201	2575	2622	1717	
	micro avg	0.6	0.63	0.61	-	-	-	-	**0.61**	**0.74**	**0.67**	-	-	-	-	33196
	macro avg	0.56	0.59	**0.56**	-	-	-	-	**0.57**	**0.65**	0.54	-	-	-	-	33196
	weighted avg	**0.59**	0.63	0.6	-	-	-	-	**0.59**	**0.74**	**0.61**	-	-	-	-	33196

TP = true positive, FP = false positive, TN = true negative, FN = false negative, P = precision, R = recall, support = number of training examples

low availability of examples or the relatedness between these criteria and other criteria.

Moreover, we analyzed the reason behind obtaining higher recall scores for the SVM classifier than for the CNN.

Recall is calculated by dividing the true positive by the total true positive and false negative values. Therefore, if the false negatives are zero, then the recall will be one. This means that the SVM recalled most of the journals that applied a specific criterion; however, this was done at the expense of its precision, where it sometimes predicted a journal to apply a specific criterion while it did not. The high recall compared to the CNN could be because the SVM classifier had one classifier per criterion; hence, it did not see other criteria while performing the classification. In addition, the number of examples was not low, which supports the classifier. In contrast, the CNN model did not recall all journals that applied the criteria, and it was not precise in its prediction. Thus, the CNN model traded between precision and recall.

We checked the criteria needed for the PDF files of the manuscripts to check if the manuscripts had a copyright holder and advertisement (criteria # 18 and 19). We found that they had low scores for SVM (0) and CNN (below 0.50). We think this low score occurred because we did not include PDF files in the learning process.

Additionally, we checked samples that scored less than 0.5. There are 17 criteria (colored or highlighted in red). However, 13 of these criteria have low sample sizes or the need for manuscript files to be evaluated, as mentioned before. The remaining criteria are as follows: 12, 23, 31, and 35. After checking some samples, we found that some journals did not present their criteria at the first two levels; thus, the classifier did not see it. In addition, there were some annotation errors.

Finally, we checked samples from the incorrectly predicted journals to understand the reasons behind obtaining low scores, and we found that most of the checked journals had their criteria in level 3, which was not included in the learning process because of limited computational resources. In contrast, the criteria that appear mostly on the first page (Home page), such as the peer review statement (criterion # 3), the ways in which the content is available to the readers (criteria # 6), and publishing schedule (criterion # 10) have high F1 scores (above 0.8), as presented in Table 28. Additionally, we believe that the fewer training examples and the large number of unique words in our dataset affected the results.

VII. CONCLUSION, LIMITATIONS, AND FUTURE WORK

Predatory journals comprise one of the risks that affect scholarly publishing, where these journals publish questionable articles and pose a global threat to the integrity and quality of scientific literature. Given their consequences and proliferation, several solutions have been developed; however, these solutions are manual and time-consuming, pressing the need for an effective automatic solution for legitimate journal detection. In this project, we aimed to build an intelligent framework that can automatically detect predatory venues with appropriate reasoning. To that end, we first discussed the origin of predatory venues and the theoretical background of the concept of predatory venues. In addition, we presented the available detection approaches for predatory venues. Moreover, a review of the predatory venues and website classification techniques is discussed.

Throughout this project, we constructed a dataset of 9,866 journals labeled as predatory and legitimate. Additionally, we scrapped approximately 6,836 journals. We annotated 1,945 predatory journals using our compiled legitimacy criteria. We discussed in detail the annotation process that we followed to obtain high-quality annotations for our journals' dataset. We also performed extensive experiments using seven different machine-learning and deep-learning models: SVM, KNN, NNs, LSTM, CNN, ALBERT, and BERT. The results obtained were promising, demonstrating the effectiveness of the proposed model in the legitimacy classification task. It also demonstrated the possibility of automating the process of journal legitimacy detection. The results showed that the CNN model outperformed the other models successfully, with an F1 score of 0.96. Additionally, we evaluated two machine and deep learning models, SVM and CNN, to provide appropriate reasoning regarding the violation. The results were not as good as those of the legitimacy detection task; however, we consider it a good starting point in this field. The SVM model achieved better micro F1 and weighted F1 of 0.67 and 0.61, respectively, whereas the CNN model achieved a better macro F1 of 0.56.

We believe that in our research, we answered our research questions and obtained satisfactory results; however, we believe that this work has several potential limitations, given the limited budget, time, and computational resources. For the dataset, we collected annotations for half of the predatory dataset, where we enforced stopping the collection process owing to time and budget constraints. Additionally, the study was limited to journals as venues because we did not find datasets of conferences' websites. In addition, the scrapping process took a long time, which led us to exclude some of the collected datasets because it required individual scrapers.

Besides to dataset construction limitations, tuning models' hyperparameters were limited in some models. The limited time led us to use the default values or previously reported values for some hyperparameters. Additionally, the tuned parameters in our experiments can be investigated with more fine-grained ranges. In addition, we could not perform some experiments because of computational resources (e.g., Doc2Vec with 500 as a vector size). In addition, the computational resources restricted us from using all the scrapped content, such as we excluded the scraped files and the third level of the websites, which affected the criteria violations detection task.

We believe that there are different ways to improve this study. As a learned lesson for future work from the previous rounds, we realized that we could decrease the annotation time if we provided the website as screenshots, because the Wayback Machine consumes a long time to serve clients.

In addition, we wanted to annotate the entire dataset of predatory and legitimate journals based on their violations. Specifically, we want to investigate the performance of data augmentation to label our dataset using legitimacy criteria.

Additionally, the experiments can be enhanced in different ways. First, apart from Word2Vec and Doc2Vec, different word embeddings can be performed, such as ELMo, BERT, and the Universal Sentence Encoder (USE) [65]. In addition, instead of training Word2Vec and Doc2Vec on our corpus, we can train them on a large corpus of different websites' structures. Moreover, we can incorporate journal annotation with textual content as features for legitimacy classification, as it could enhance the classification task. In addition, the collected 9,866 can be used to predict legitimacy using URL and other features such as the WHOIS domain and ISSN portal.

We intend to evaluate more machine learning and deep learning models and use different feature representation techniques to enhance the process of detecting the violations, where hybrid approaches can capture different features; thus, the results can be improved.

REFERENCES

[1] J. Olivarez, S. Bales, L. Sare, and W. vanDuinkerken, "Format aside: Applying Beall's criteria to assess the predatory nature of both OA and non-OA library and information science journals," *College Res. Libraries*, vol. 79, no. 1, p. 52, Jan. 2018, doi: 10.5860/crl.79.1.52.

[2] J. Beall, "Dangerous predatory publishers threaten medical research," *J. Korean Med. Sci.*, vol. 31, no. 10, pp. 1511–1513, Oct. 2016, doi: 10.3346/jkms.2016.31.10.1511.

[3] C. Shen and B.-C. Björk, "'Predatory' open access: A longitudinal study of article volumes and market characteristics," *BMC Med.*, vol. 13, no. 1, pp. 1–15, Oct. 2015, doi: 10.1186/s12916-015-0469-2.

[4] The InterAcademy Partnership (IAP). (Mar. 2022). *Combatting Predatory Academic Journals and Conferences*. Accessed: Jun. 16, 2022. [Online]. Available: https://www.interacademies.org/publication/predatory-practices-report-English

[5] D. A. Forero, M. H. Oermann, A. Manca, F. Deriu, H. Mendieta-Zerón, M. Dadkhah, R. Bhad, S. N. Deshpande, W. Wang, and M. P. Cifuente, "Negative effects of 'predatory' journals on global health research," *Ann. Global Health*, vol. 84, no. 4, pp. 584–589, May 2018.

[6] S. Eriksson and G. Helgesson, "Time to stop talking about 'predatory journals,'" *Learned Publishing*, vol. 31, no. 2, pp. 1–3, 2018.

[7] J. Beall, "Criteria for determining predatory open-access publishers," Scholarly Open Access, 2015. [Online]. Available: https://scholarlyoa.files.wordpress.com/2015/01/criteria-2015.pdf

[8] *Beall's List—Of Predatory Journals and Publishers*. Accessed: Mar. 10, 2020. [Online]. Available: https://beallslist.net/

[9] J. Beall, "Predatory publishing is just one of the consequences of gold open access," *Learned Publishing*, vol. 26, no. 2, pp. 79–84, 2013, doi: 10.1087/20130203.

[10] *Cabell's International—Homepage*. Accessed: Mar. 10, 2020. [Online]. Available: https://www2.cabells.com/

[11] *Principles of Transparency and Best Practice in Scholarly Publishing*, Committee on Publication Ethics, U.K., Jan. 2014, doi: 10.24318/cope.2019.1.12.

[12] L. Hoffecker, "Cabells scholarly analytics," *J. Med. Library Assoc.*, vol. 106, no. 2, pp. 270–272, Apr. 2018, doi: 10.5195/jmla.2018.403.

[13] *Promoting Integrity in Scholarly Research and Its Publication | Committee on Publication Ethics: COPE*. Accessed: Mar. 10, 2020. [Online]. Available: https://publicationethics.org/

[14] DOAJ. *Directory of Open Access Journals*. Accessed: Mar. 10, 2020. [Online]. Available: https://doaj.org

[15] J. Bohannon, "Who's afraid of peer review?" *Science*, vol. 342, no. 6154, pp. 60–65, Oct. 2013, doi: 10.1126/science.342.6154.60.

[16] J. A. Teixeira da Silva and P. Tsigaris, "Issues with criteria to create blacklists: An epidemiological approach," *J. Academic Librarianship*, vol. 46, no. 1, Jan. 2020, Art. no. 102070, doi: 10.1016/j.acalib.2019.102070.

[17] M. Baker, "Open-access index delists thousands of journals," Nature News, May 2016, doi: 10.1038/nature.2016.19871.

[18] M. Strinzel, A. Severin, K. Milzow, and M. Egger, "Blacklists and whitelists to tackle predatory publishing: A cross-sectional comparison and thematic analysis," *mBio*, vol. 10, no. 3, Jun. 2019, Art. no. e00411, doi: 10.1128/mBio.00411-19.

[19] A. Adnan, S. Anwar, T. Zia, S. Razzaq, F. Maqbool, and Z. U. Rehman, "Beyond Beall's blacklist: Automatic detection of open access predatory research journals," in *Proc. IEEE 20th Int. Conf. High Perform. Comput. Commun., IEEE 16th Int. Conf. Smart City, IEEE 4th Int. Conf. Data Sci. Syst. (HPCC/SmartCity/DSS)*, Exeter, U.K., Jun. 2018, pp. 1692–1697.

[20] J. Beall, "What I learned from predatory publishers," *Biochemia Medica*, vol. 27, no. 2, pp. 273–278, Jun. 2017, doi: 10.11613/BM.2017.029.

[21] T. V. McCann and M. Polacsek, "False gold: Safely navigating open access publishing to avoid predatory publishers and journals," *J. Adv. Nursing*, vol. 74, no. 4, pp. 809–817, Apr. 2018.

[22] D. Butler, "Investigating journals: The dark side of publishing," *Current Med. Res. Opinion*, vol. 30, no. 1, pp. S9–S23, 2014.

[23] D. F. Polit and C. T. Beck, *Nursing Research: Generating and Assessing Evidence for Nursing Practice*, 10th ed. Philadelphia, PA, USA: Wolters Kluwer Health, 2016.

[24] J. Revés, B. M. D. Silva, J. Durão, N. V. Ribeiro, S. Lemos, and P. Escada, "Predatory publishing: An industry that is threatening science," *Acta Médica Portuguesa*, vol. 31, no. 3, pp. 141–143, Mar. 2018.

[25] G. Strong, "Understanding quality in research: Avoiding predatory journals," *J. Human Lactation*, vol. 35, no. 4, pp. 661–664, Nov. 2019.

[26] G. Richtig, M. Berger, B. Lange-Asschenfeldt, W. Aberer, and E. Richtig, "Problems and challenges of predatory journals," *J. Eur. Acad. Dermatol. Venereol.*, vol. 32, no. 9, pp. 1441–1449, Sep. 2018, doi: 10.1111/jdv.15039.

[27] The Scholarly Kitchen. (May 11, 2015) *Should We Retire the Term 'Predatory Publishing?* Accessed: Mar. 11, 2020. [Online]. Available: https://scholarlykitchen.sspnet.org/2015/05/11/should-we-retire-the-term-predatory-publishing/

[28] A. Tosti and A. J. Maddy, "Ranking predatory journals in dermatology: Distinguishing the bad from the ugly," *Int. J. Dermatol.*, vol. 56, no. 7, pp. 718–720, Jul. 2017.

[29] A. R. Memon, "Predatory journals spamming for publications: What should researchers do?" *Sci. Eng. Ethics*, vol. 24, no. 5, pp. 1617–1639, Oct. 2018.

[30] T. F. Frandsen, "How can a questionable journal be identified: Frameworks and checklists," *Learned Publishing*, vol. 32, pp. 221–226, Mar. 2019.

[31] L. Toutloff. (Mar. 20, 2019). *Cabells Blacklist Criteria V1.1*. The Source. Accessed: Mar. 10, 2020. [Online]. Available: https://blog.cabells.com/2019/03/20/blacklist-criteria-v1-1/

[32] M. Dadkhah and G. Bianciardi, "Ranking predatory journals: solve the problem instead of removing it!" *Adv. Pharmaceutical Bull.*, vol. 6, no. 1, pp. 1–4, Mar. 2016, doi: 10.15171/apb.2016.001.

[33] R. Lang, K. Porter, H. B. Krentz, and M. J. Gill, "Evaluating medical conferences: The emerging need for a quality metric," *Scientometrics*, vol. 122, no. 1, pp. 759–764, Jan. 2020.

[34] C. Laine and M. A. Winker, "Identifying predatory or pseudo-journals," *Biochemia Medica*, vol. 27, no. 2, pp. 285–291, Jun. 2017, doi: 10.11613/BM.2017.031.

[35] M. A. Shahri, M. D. Jazi, G. Borchardt, and M. Dadkhah, "Detecting hijacked journals by using classification algorithms," *Sci. Eng. Ethics*, vol. 25, pp. 655–668, Apr. 2017, doi: 10.1007/s11948-017-9914-2.

[36] M. Dadkhah, T. Maliszewski, and V. V. Lyashenko, "An approach for preventing the indexing of hijacked journal articles in scientific databases," *Behav. Inf. Technol.*, vol. 35, no. 4, pp. 298–303, Apr. 2016, doi: 10.1080/0144929X.2015.1128975.

[37] *Official List of JPPS-Assessed Journals—JPPS (EN-GB)*. Accessed: Mar. 10, 2020. [Online]. Available: https://www.journalquality.info/en/journals-all/

[38] OASPA. *Open Access Scholarly Publishers Association*. Accessed: Mar. 10, 2020. [Online]. Available: https://oaspa.org/

[39] *Stop Predatory Journals*. Accessed: Mar. 10, 2020. [Online]. Available: https://predatoryjournals.com/

[40] M. Jalalian, "The story of fake impact factor companies and how we detected them," *Electron. Physician*, vol. 7, no. 2, pp. 1069–1072, Jun. 2015, doi: 10.14661/2015.1069-1072.
[41] M. Hashemi, "Web page classification: A survey of perspectives, gaps, and future directions," *Multimedia Tools Appl.*, vol. 79, pp. 11921–11945, Jan. 2020, doi: 10.1007/s11042-019-08373-8.
[42] M. Maktabar, A. Zainal, M. A. Maarof, and M. N. Kassim, "Content based fraudulent website detection using supervised machine learning techniques," in *Hybrid Intelligent Systems*, vol. 734. Cham, Switzerland, 2018, pp. 294–304, doi: 10.1007/978-3-319-76351-4_30.
[43] L. Beltzung, A. Lindley, O. Dinica, N. Hermann, and R. Lindner, "Real-time detection of fake-shops through machine learning," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, Dec. 2020, pp. 2254–2263, doi: 10.1109/BigData50022.2020.9378204.
[44] C. Carpineto and G. Romano, "An experimental study of automatic detection and measurement of counterfeit in brand search results," *ACM Trans. Web*, vol. 14, no. 2, pp. 1–35, Feb. 2020, doi: 10.1145/3378443.
[45] Y. Kim, "Convolutional neural networks for sentence classification," in *Proc. EMNLP*, 2014, pp. 1746–1751, doi: 10.3115/v1/D14-1181.
[46] W. Ali and A. A. Ahmed, "Hybrid intelligent phishing website prediction using deep neural networks with genetic algorithm-based feature selection and weighting," *IET Inf. Secur.*, vol. 13, no. 6, pp. 659–669, Nov. 2019, doi: 10.1049/iet-ifs.2019.0006.
[47] H. Li, Z. Zhang, and Y. Xu, "Web page classification method based on semantics and structure," in *Proc. 2nd Int. Conf. Artif. Intell. Big Data (ICAIBD)*, Chengdu, China, May 2019, pp. 238–243, doi: 10.1109/ICAIBD.2019.8837027.
[48] A. Gupta and R. Bhatia, "Ensemble approach for web page classification," *Multimedia Tools Appl.*, vol. 80, no. 16, pp. 25219–25240, Jul. 2021, doi: 10.1007/s11042-021-10891-3.
[49] L. Deng, X. Du, and J.-Z. Shen, "Web page classification based on heterogeneous features and a combination of multiple classifiers," *Frontiers Inf. Technol. Electron. Eng.*, vol. 21, no. 7, pp. 995–1004, Jul. 2020.
[50] T. N. P. Vinh and H. H. Kha, "Vietnamese news articles classification using neural networks," *J. Adv. Inf. Technol.*, vol. 12, no. 4, pp. 1–7, 2021, doi: 10.12720/jait.12.4.363-369.
[51] C.-G. Artene, D.-D. Vecliuc, M. N. Tibeică, and F. Leon, "An experimental study of convolutional neural networks for functional and subject classification of web pages," *Vietnam J. Comput. Sci.*, vol. 9, no. 4, pp. 435–453, Nov. 2022, doi: 10.1142/S2196888822500245.
[52] C.-G. Artene, M. N. Tibeică, D. D. Vecliuc, and F. Leon, "Convolutional neural networks for web documents classification," in *Proc. 13th Asian Conf. Intell. Inf. Database Syst.*, Berlin, Germany, Apr. 2021, pp. 289–302, doi: 10.1007/978-3-030-73280-6_23.
[53] C.-G. Artene, M. N. Tibeica, and F. Leon, "Using BERT for multi-label multi-language web page classification," in *Proc. IEEE 17th Int. Conf. Intell. Comput. Commun. Process. (ICCP)*, Oct. 2021, pp. 307–312, doi: 10.1109/ICCP53602.2021.9733492.
[54] *Selenium With Python—Selenium Python Bindings 2 Documentation*. Accessed: Jun. 18, 2022. [Online]. Available: https://selenium-python.readthedocs.io/
[55] *Zyte (Formerly Scrapinghub) #1 Web Scraping Service*. Accessed: Apr. 18, 2022. [Online]. Available: https://www.zyte.com/
[56] Data Solutions to Drive AI. *Practical Crowdsourcing for Efficient ML*. Accessed: Mar. 18, 2022. [Online]. Available: https://toloka.ai/academy/y-data
[57] D. Kouzis-Loukas, *Learning Scrapy: Learn the Art of Efficient Web Scraping and Crawling With Python*. Birmingham, U.K.: Packt Publishing, 2016.
[58] X. Deng, Y. Li, J. Weng, and J. Zhang, "Feature selection for text classification: A review," *Multimedia Tools Appl.*, vol. 78, no. 3, pp. 3797–3816, 2019.
[59] K. Kowsari, K. J. Meimandi, M. Heidarysafa, S. Mendu, L. Barnes, and D. Brown, "Text classification algorithms: A survey," *Information*, vol. 10, no. 4, p. 150, Apr. 2019.
[60] M. M. Truşcă, "Efficiency of SVM classifier with Word2Vec and Doc2Vec models," in *Proc. Int. Conf. Appl. Statist.*, vol. 2019, vol. 1, no. 1, pp. 496–503.
[61] T. Joachims, "Text categorization with support vector machines: Learning with many relevant features," in *Machine Learning*. Berlin, Germany: Springer, 1998, pp. 137–142, doi: 10.1007/BFb0026683.
[62] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," 2018, *arXiv:1810.04805*.
[63] DOAJ News Service. (May 22, 2014). *DOAJ Publishes Lists of Journals Removed and Added*. Accessed: May 16, 2020. [Online]. Available: https://blog.doaj.org/2014/05/22/doaj-publishes-lists-of-journals-removed-and-added/
[64] Committee Publication Ethics. *Principles of Transparency and Best Practice in Scholarly Publishing*. Accessed: May 18, 2022. [Online]. Available: https://publicationethics.org/resources/guidelines-new/principles-transparency-and-best-practice-scholarly-publishing
[65] Committee Publication Ethics. *Predatory Publishing*. Accessed: May 18, 2022. [Online]. Available: https://publicationethics.org/resources/discussion-documents/predatory-publishing
[66] *COPE: Committee on Publication Ethics*. Accessed: May 18, 2022. [Online]. Available: https://publicationethics.org/
[67] D. Ariely, A. Bracha, and S. Meier, "Doing good or doing well? Image motivation and monetary incentives in behaving prosocially," *Amer. Econ. Rev.*, vol. 99, no. 1, pp. 544–555, Feb. 2009, doi: 10.1257/aer.99.1.544.
[68] D. Karger, S. Oh, and D. Shah, "Iterative learning for reliable crowdsourcing systems," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 24, 2011, pp. 1–15. Accessed: May 19, 2022. [Online]. Available: https://proceedings.neurips.cc/paper/2011/hash/c667d53acd899a97a85de0c201ba99be-Abstract.html
[69] Appen Success Center. *Test Question Best Practices*. Accessed: May 19, 2022. [Online]. Available: https://success.appen.com/hc/en-us/articles/213078963-Test-Question-Best-Practices
[70] K. Sechidis, G. Tsoumakas, and I. Vlahavas, "On the stratification of multi-label data," in *Machine Learning and Knowledge Discovery in Databases*. Berlin, Germany: Springer, 2011, pp. 145–158, doi: 10.1007/978-3-642-23808-6_10.
[71] L. Buitinck, G. Louppe, M. Blondel, F. Pedregosa, A. Mueller, O. Grisel, V. Niculae, P. Prettenhofer, A. Gramfort, J. Grobler, R. Layton, J. Vanderplas, A. Joly, B. Holt, and G. Varoquaux, "API design for machine learning software: Experiences from the scikit-learn project," 2013, *arXiv:1309.0238*.
[72] Q. Le and T. Mikolov, "Distributed representations of sentences and documents," in *Proc. ICML*, 2014, pp. 1188–1196.
[73] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut, "ALBERT: A lite BERT for self-supervised learning of language representations," 2019, *arXiv:1909.11942*.

WED MAJED BIN ATEEQ is from Riyadh, Saudi Arabia. She received the B.S. and M.S. degrees (Hons.) in information technology from King Saud University, Riyadh, in 2016 and 2022, respectively. She is currently a Teaching Assistant with the Information Technology Department, Imam Mohammad Ibn Saud Islamic University. Her research interests include natural language processing, machine learning, deep learning, and detection of fraudulent publishing websites. Her awards include Dean's Award for Scientific Excellence, the third, fourth, and tenth places at the Scientific Meetings with King Saud University.

HEND S. AL-KHALIFA is currently a Professor with the Information Technology Department, King Saud University. She has contributed more than 180 research papers to symposiums, workshops, international conferences, and journals. Moreover, she has served as a program committee member at many national and international conferences and as a reviewer for several journals. Her research interests include semantic web technologies, computers for people with special needs, and Arabic NLP.