https://doi.org/10.5755/j02.eie.34743 ELEKTRONIKA IR ELEKTROTECHNIKA, ISSN 1392-1215, VOL. 29, NO. 5, 2023

Hybrid Technique for Detecting Extremism in Arabic Social Media Texts

Israa Akram Alzuabidi[1], Layla Safwat Jamil[2], Amjed Abbas Ahmed[3,4,*], Shahrul Azman Mohd Noah[3], Mohammad Kamrul Hasan[3]

[1]Continuing Education Unit, College of Arts, University of Baghdad, Baghdad, Iraq

[2]College of Agricultural Engineering Sciences, University of Baghdad, Baghdad, Iraq

[3]Center for Cyber Security, Faculty of Information Science and Technology, Universiti Kebangsaan Malaysia (UKM), Bangi 43600, Malaysia

[4]Department of Computer Techniques Engineering, Imam Al-Kadhum College (IKC), Baghdad, Iraq

*israa.jeyad@coart.uobaghdad.edu.iq; layla.s@coagri.uobaghdad.edu.iq; *amjedabbas@alkadhum-col.edu.iq; shahrul@ukm.edu.my; mkhasan@ukm.edu.my*

***Abstract*—Today, social media sites like Twitter provide effective platforms to share opinions and thoughts in public with millions of other users. These opinions shared on such sites influence a large number of people who may easily retweet them and accelerate their spread. Unfortunately, some of these opinions were expressed by extremists who promoted hateful content. Since Arabic is one of the most spoken languages, it is crucial to automate the process of monitoring Arabic content published on social sites. Therefore, this study aims to propose a hybrid technique to detect extremism in Arabic social media texts and articles to monitor the situation of published extremist content. The proposed technique combines the lexicon-based approach with the rough set theory approach. The rough set theory is employed with two approximation strategies: lower approximation and accuracy approximation. The hybrid technique used the rough set theory as a classifier and the lexicon-based as a vector. Furthermore, this study built three types of corpuses (V1, V2, and V3) collected from Twitter. The experimental findings show that among the proposed hybrid methods, the accuracy approximation was superior to the lower approximation with seed vector. It was also revealed that hybrid methods outperformed machine learning techniques in terms of efficiency. Moreover, the study recommends using an accuracy approximation method with seed vector to identify the polarity of the text.**

***Index Terms*—Accuracy approximation; Corpus; Extremism; Lexicon; Lower approximation; Rough set theory.**

I. INTRODUCTION

Extremism is the promotion of extreme methods or viewpoints. The term is most frequently used in a political or religious context to describe an ideology that is thought to be very different from the norms of society. The simplest definition of it is the actions (beliefs, feelings, attitudes, methods, etc.) of a person who differs significantly from the norm. Nowadays, many people may easily publish numerous postings online, making it impossible to manually code their contributions. Knowing who wrote the post will help the extremism analyst efficiently and precisely classify it (i.e., user or publisher). For decision-making purposes, however, it is important to automatically categorise these posts according to extremism detection of unstructured online content (or unstructured textual data). The decision-making process is incomplete without incorporating the knowledge gained from these online sources. In particular, public opinion surveys have always played an important role in policymaking at all levels.

Rapid system development has a direct impact on people's lives. Therefore, it is essential to give such systems the ability to assess data in real-time and make wise decisions to address certain challenges. People from all walks of life can read what is put on public websites and the information they find there can aid them in making crucial life decisions. In the field of identifying extremism, it requires a lot of time and effort to make a complete list of all topics or situations [1]. It is impossible to manually process the billions of articles produced by people each month by conducting public opinion surveys. Understanding the extremism and nonextremism of Arabic postings requires automated ideological text analysis techniques that can process massive amounts of data rapidly. The most crucial and challenging aspect of automated processing is determining whether an Arabic post is extremist or not [2], [3].

Researchers and academics have already benefited from the use of opinion mining and intelligent technologies to automate the content analysis process, notably in the areas of data collection, preparation, management, and visualisation. These modifications have allowed us it to conduct extensive research and to monitor websites in real time. Recent text mining studies have shown that when a feature set is found and weighted, the texts are then frequently divided into three categories rather than two using a traditional binary classifier [4], [5]. A traditional binary classifier is unable to reclassify training documents back into their original categories,

Manuscript received 30 June, 2023; accepted 25 September, 2023.

whether they were initially identified as relevant or irrelevant. The idea that documents may be neatly separated into two categories is a common misunderstanding. However, a traditional text classifier cannot handle this assumption because it is too powerful. This makes it difficult for any classical classifier to do binary classification in a single pass. There are some objects whose polarity is ambiguous, and this group of objects, known as the boundary region, is assumed to be real. The rough set theory (RST) has demonstrated the possibility of defining the boundary and the viability of area division [6], [7]. To arrive at the final result, which will include two unique zones, one with only relevant items and the other with only irrelevant ones, a binary classifier is required. Because of this, it is hard to determine which way all documents are going at the border point, which makes it hard to process the border area [8].

This research addresses the problem of Arabic extremism rather than focussing on customer reviews, which have been the subject of several earlier studies [9], [10]. Opinion mining has already attracted the attention of researchers studying extremism, but they have largely focused on the analysis of specific phrases or statements. In this study, we believe that the focus of various previous studies on using short texts such as tweets to analyse opinions is insufficient to provide a comprehensive understanding of opinion mining in the context of Arabic extremism [2].

We focus on extremism in Arabic due to the influence of the Arab Spring, which featured several extremist activities and events, the majority of which were covered online [11]. Politicians need to evaluate these publications so they can make judgments that are in the best interests of the state, as well as the security and academic establishments. Researchers were asked to investigate these incidents to determine the impact of extremism on the general public.

This research tries to fill the gap caused by the lack of publicly available and easily accessible Arabic extremism in the extremist opinion mining sector (there are no corpora for Arabic extremism available). It aims to propose a hybrid technique for detecting extremism in Arabic social media texts and articles. The technique has two tasks: detecting extremism in Arabic posts and mining opinions that are not-extremist. The technique is a combination of the lexicon-based approach (LA) with the RST approach. The RST is used with two approximation strategies: the lower approximation (LA) and the accuracy approximation (AA). Figure 1 shows an example of a text on social media with its Arabic translation, and the proposed technique is intended to identify this post as extremist.

We must stand against Muslims in Sweden and demolish and burn all mosques and kill Muslims	يجب الوقوف ضد المسلمون في السويد وهدم جميع المساجد وحرقها وقتل جميع المسلمون

Fig. 1. Example of social media text with its Arabic translation.

The rest of the paper is organised as follows. Section II introduces related work in the area, while Section III presents the methodology adopted in developing the proposed technique. Section IV provides the experimental results of the proposed technique. Finally, we present the conclusions that can be drawn from this research work in Section V.

II. RELATED WORK

The hybrid approach combines the lower, upper, and accuracy approximations defined by Pawlak with lexicon-based techniques based on statistical-based and human-based input to divide the text classification problem into two distinct decision-making actions based on statistical attributes [12]–[15]. But there are not enough training examples for text classification tasks to make the usual three-way choice based on probability.

However, note that no actual proof has been shown, and the analysis described here is purely speculative. Some analysts have tried to find a way out of this sticky predicament by relying less on probability and more on its close relative, odds, which is the ratio between the chances of something happening and the chances of it not happening. In place of the traditional method of using a pair of boundary values of the region-division [16], a pair of centroid vectors is proposed independently of the relevant and irrelevant training subsets. This is because the distance between pairs of related documents in the vector space of the document closely correlates with their degree of similarity. To improve the overall performance of traditional binary classifiers, it is suggested that a set of decision rules be made based on the pair of centroids, in addition to the specific criteria and Euclidean relations of the document vectors. This would help divide the documents into three regions and give more information about the undetermined objects in the boundary region.

Arabic is spoken in more than 30 countries and territories and is the fifth most spoken language in the world. It is the native tongue of about 422 million people and the second language of another 250 million [17]–[19]. There are 28 different symbols that make up the Arabic alphabet. Like English, Arabic does not have a system of uppercase and lowercase letters. The Arabic script reads from right to left [20]. Arabic, a Semitic language, has morphological grounds that are more complicated and numerous than those of English [21]. It has a complicated morphology due to the way words in it change form as they are inflected [22], [23].

A word in Arabic can be feminine or masculine, singular, dual, or multiple, and can also take on one of three grammatical cases: nominative, accusative, or genitive [24]. The nominative case is used for subjects, the accusative case for objects of verbs, and the genitive case for prepositional phrases. There are three primary types of words: nouns (including adjectives and adverbs), verbs, and particles. Some nouns and all verbs have a common set of morphological roots. Affixes are predetermined patterns used to create new words. The numerical value, gender, and tense can all be indicated by adding an appropriate affix to a word. Learning Arabic is difficult for a variety of reasons [22], [25]:

– Sentence order, e.g., (“ التعليم يحتاج الى اصلاح”) can be replaced with (“يحتاج التعليم الى اصلاح”) to express the same idea by changing the sentence order. As a result, there are a large number of free orders in Arabic;

– In the Arabic language, there is a level of complication with expressions such as: (اعلم من السائل على المسؤول ان يكون).

Because of these problems, the Arabic language needs a set of preprocessing methods before it can be used for any pocess.

There are few studies on identifying extremism in Arabic, and those that are available focus primarily on the English language. Al-Hassan and Al-Dossari [26] conducted a survey on the detection of hate speech on multilingual social networks. Aljarah *et al.* [27] proposed an approach to detect hate speech in an Arabic social network. They applied natural language processing (NLP) techniques and machine learning methods. They collected a data set from Twitter using the Twitter streaming application programming interface (API) and then deployed it into four machine learning algorithms: support vector machine (SVM), Naive Bayes (NB), decision tree (DT), and random forest (RF). Their results showed that the RF classifier performed the best over the other classifiers used. Johnston and Weiss [28] proposed an approach that can automatically identify a subset of web pages and text on social media that contains extremist content. The approach uses deep learning algorithm to classify text as extremist or nonextremist. Ahmad, Asghar, Alotaibi, and Awan [29] suggested a terrorism-related content analysis framework with the goal of categorising tweets into extremist and nonextremist classes employing deep learning-based sentiment analysis techniques. They claimed that their outcomes of their experiments are positive and open doors for future studies.

Mursi, Alahmadi, Alsubaei, and Alghamdi [30] provided a manually labelled data set of 3,000 Arabic Islamic tweets that contain hateful and nonhateful tweets. They utilised advanced machine learning techniques and performed sentiment analysis to capture the meaning of Arabic words in a proper word embedding (Word2Vec). They also used their model to classify 100,000 tweets. Sofat and Bansal [31] proposed an algorithm to detect online radicalised accounts on the Internet and quantify the degree to which these user accounts propagate radical content. They used three features: similarity to domain, presence of radical content, and sentiment to calculate the radicalness score for each online user. Their algorithm used a deep learning technique to accurately differentiate between radical/nonradical content. Sanoussi, Xiaohua, Agordzo, Guindo, Al Omari, and Issa [32] aimed to detect hate speech for French texts. They collected 14,000 comments on Facebook and labelled them in four categories (hate, offence, insult, and neutral). NLP is used to clean the data set and then three word embedding methods are applied: Word2Vec, Doc2Vec, and Fasttext. Then, four classifiers are used to classify the comments collected. The classifiers are logistic regression (LR), SVM, RF, and k-nearest neighbours (KNN). The results showed that the SVM classifier gives the best results.

To summarise, there is a lack of studies that focus on detecting extremism for Arabic language, and the available approaches use the traditional classifiers for classifying texts as extremist or nonextremist. This encourages us to propose a hybrid technique to solve the problem of detecting extremism. The proposed technique is described in detail in the next section.

III. PROPOSED TECHNIQUE

Machine learning algorithms struggle with one of the most specialised problems because there are no shared properties between the article and the corpus. It takes a long time to use this procedure, which is problematic when working with only three grams. Thus, a classifier based on a lexicon is presented, and the rough set theory (RST) technique is suggested as a possible vector. The suggested technique uses terms (relevant words) rather than numerical vectors, so it can quickly categorise the article. Although machine learning is faster at solving differential equations, RST uses set theory to improve accuracy.

RST is used in our study to categorise the data. It uses two approximation strategies: lower approximation (LA) and accuracy approximation (AA). As it requires only intersection operations, the lower approximation may be computed relatively quickly. There are, however, drawbacks, such as its high value and uniform class. Here, an accuracy approximation is employed to improve the procedure by overcoming the restrictions of the lower approximation.

For lexicon-based systems, the suggested vector consists of two primary components: the lexicon vector and the seed vector. In such systems, the article is parsed into individual tokens using the three grammatical components. This creates a lexicon vector. Second, terms from a certain category, such as “extremism” and “nonextremism”, are extracted for their frequencies to build the seed vector. Thus, threshold values are employed to pick words with frequencies below or equal to the respective threshold values. Figure 2 shows the overall hybrid process, which is followed by human-based selection to remove unnecessary terms.

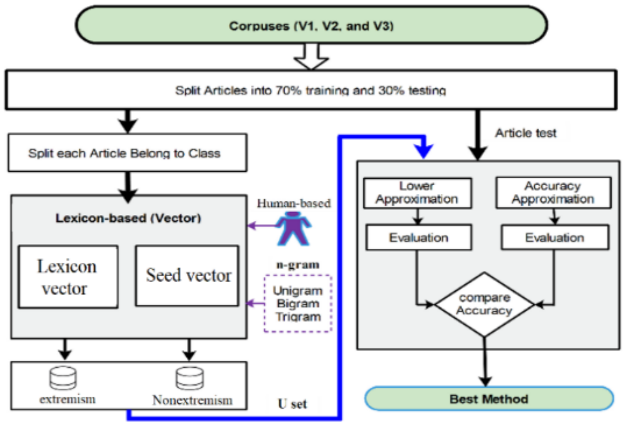

Fig. 2. General hybrid technique.

Corpora V1, V2, and V3 are shown in Fig. 2. The 70/30 ratio between training and testing utilises 70 % of the corpora. There are two primary components to the proposed hybrid technique. The first is the lexicon-based (vector) approach, which is used to find instances of words appearing in two separate vectors. In contrast to the focus of the seed human-based vector and the unigram, the focus of the lexicon-sole vector is three grams. In the second component, lower estimates and precision approximations are put to the test. Finally, we evaluate each technique side by side to find the most effective one. The following is a description of the lexicon-based (vector).

A. *Lexicon-Based (Vector)*

Since machine learning may have its limits with only three grams of data, a lexicon-based approach was developed to overcome this obstacle. The lexicon approach offers several benefits. This approach has a few advantages over others, including the fact that it is quick and can generate a vector for each class [33]. Furthermore, it can handle both narrow and broad topics with the same ease. The following is an explanation of how the lexicon vector and the seed vector were constructed. The lexicon vector stands for dictionaries, whereas the seed vector combines statistical corpora and human input. In what follows, we examine the context of these three vectors and discuss their practical applications. The division of the corpus into these three categories is seen in Fig. 3. The items that make up each category are included in their respective classes. These pieces are broken down into words, and the choice of words is made in accordance with the vector employed.

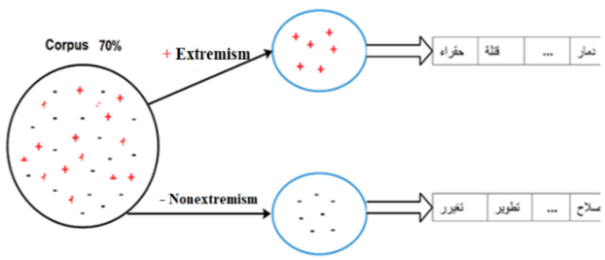

Fig. 3. Lexicon-based vector.

Assume $P = \{A_1, A_2, A_3 \ldots A_n\}$, where n represents number of the articles where every one of the articles belongs to the label $L = \{extremisim, nonextremism\}$. L makes a partition on P such that $A_i \in l_j$ for a value of j, in the case where $A_i \in l_j, A_i A_i$ is referred to by $A_i^j A_i^j$. Equation (1) is used to save and classify into classes:

$$\begin{cases} P_1 = \bigcup A_i^1 \middle| A_i^1 \in l_1, \\ P_2 = \bigcup A_i^2 \middle| A_i^2 \in l_2, \end{cases} \tag{1}$$

where l_1 and l_2 stands for extremism and nonextremism, respectively. Equation (1) makes partition such that every one of the articles must be part of one partition precisely, where P partition is either extremism and nonextremism.

1. *Lexicon Vector*

First, the lexicon vector is proposed for use in this research. Its capabilities are identical to those of dictionary-based vectors. Our corpus is divided into training and testing sets, and the resulting lexicon vector is illustrated in Fig. 3. In machine learning, training entails constructing a vector with a split size similar to the traditional 70:30 split. This vector was constructed using (1); each of the three partitions is from the category of our extremism data sets and is composed of words rather than numbers. The construction time will be reduced as a result of this. We used five grams to generate the vector; articles were tokenised by weight. The vector is then constructed after this step. The lexicon vector is constructed using the following recommended equation.

Equation (1) creates portions for each essay in the class. For each category in L, a U set is built by (2)

$$U_j = \left\{ W_{k,r}^i \mid W_{k,r}^i \in P_j, i = 1, 2, 3, \ldots, n, i = 1, 2 \right\}. \tag{2}$$

Ultimately, in the classification models, lower estimation and consistency inference, the U_j set is being used.

2. *Seed Vector*

Since there are not as many operations as with BOW, this vector is a viable alternative to the lexicon vector that reduces construction time without sacrificing accuracy. Even if good findings are produced, the issue of low precision persists. The seed vector, a proposed new vector, is proposed as a possible solution. Unigrams are the only building blocks of this vector. This vector is based on the corpus-based approach, which employs statistical and human-based methodologies to determine which words are most successful. Figure 4 depicts the process by which these powerful words are formed.

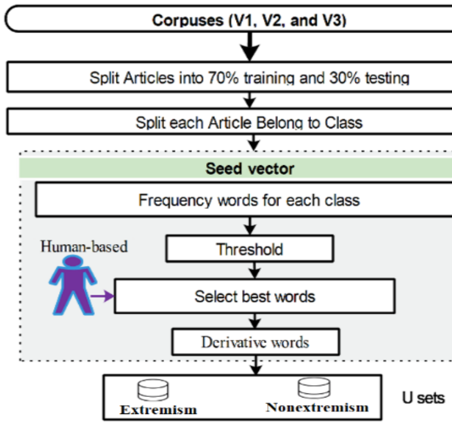

Fig. 4. Seed vector.

Figure 4 depicts how seed vector would be created by calculating frequency of the words belonging to some certain class, such as extremism and nonextremism. The frequencies of words in each partition are calculated using (3)

$$fre = \left(W_{k,r}^i \in A_i^j \right) = number\ of\ words\ frequancy\, W_{k,r}^i\ in\ A_i^j. \tag{3}$$

Since a large number of words can be generated from (3), threshold values are used to filter out words with frequencies lower than or equal to the set threshold (number 30 is used as a threshold). Thirty words were used in this study to indicate how quickly and effortlessly an expert could solve that problem. As shown in (4), a U set can be made by first making a list of the most common words in each class and then choosing the one with the highest frequency, as shown in (3)

$$U_j = \left\{ \begin{matrix} W_{k,r}^i \mid W_{k,r}^i \in A_i^j\ Such that\ frequancy \\ W_{k,r}^i\ in threshold\ fre\left(W_{k,r}^i \right) \end{matrix} \right\}. \tag{4}$$

The most frequent words are then presented to human specialists, who select the most functional (unigram) terms from the set. Words with the same meaning as those used by human experts are extracted from the corpus V1 database. Because the original corpus was not stemmed, the suggested method for making seed vectors uses terms from Table I that are related to the original terms.

TABLE I. SAMPLE LIST OF WORDS TO BUILD SEED VECTOR.

Classes	Words	Derivative words
extremism	داعش، ايران، مقتدى، القتل، التنظيم، الارهاب، القاعدة، متشدد، تجاوز، طائفي، فساد	قتل، قاعدة، تنظيم، ارهاب
nonextremism	الاصلاح، ابطال، حرر	اصلاح، بطل

B. *Hybrid Method Classifier*

The RST is used to categorise the article here according to its orientation. A table was required to display the information in the first RST. There are some drawbacks to reusing the table from the first RST of this work. For one thing, we cannot construct a table without resorting to techniques like term frequency (TF) or term frequency-inverse document frequency (TF-IDF). The second factor is the time required to conduct the test due to indiscernibility (IND), which will be enormous. And finally, it would be hard to figure out the value of rare words using TF or TF-IDF if they had to be added to the table.

Hybrid approaches are thus defined as those that utilise both rough set theory and lexicon-based techniques. Now that three vectors have been constructed, they may be used as feature extraction tools. In the case of the four parameters denoted by $PM = U, A, V, > f$, we employ and apply our polarity approach as the original. Table II explains these factors.

TABLE II. PARAMETERS OF POLARITY METHOD.

Parameter	Description
U	N objectives are a finite and nonempty set. In the case of this study, the goals are to tweet. The total number of twitter comments $\langle a_1, a_{2,\dots,} a_n \rangle$
A	Nonempty and finite set of the features. We will need words of at least three grams in weight, preferably of human origin. As a result, the A-frame structure relies on a large vocabulary to be constructed. In this work, we make A by combining two different kinds of vectors, just as we did before when we talked about how vectors are made. The words in vector $\langle w_1, w_{2,\dots,} w_m \rangle$
V	Attributes are classified V_l whereas l into two categories: extremism and nonextremism
F	$f; A \to V$ $f; A \to V$ information or description function $f\ (x, a)\ \in\ V_l$

Any corpus should be divided into training and testing sets, as previously described. Here, we train to create the vectors. "*U* Set" is shorthand for the collection of all training materials; in this book [14], there are two categories of items in the *U* Set: extremism and nonextremism. Words from each article are culled using a three-gram or human-based approach. The extracted words should neatly fall into one of three categories. These groups, also called "domains", are represented by the letter *V*, and when a word is taken from set A and mapped to set B of the test words of the article, it is put into one of the three classes *V*.

To determine which class an article belongs to, the IND (IND = set of words dependent on three grams) is constructed for each article that undergoes the three-gram test and tokenisation and then mapped to the *V* domain. In the next sections, we will show how much weight to give to lower approximation vs. precision approximation in this context.

Lower approximation method. The primary strategy used in this research is a classification system to determine the category of the article (the orientation of the article). IND testing will be used to determine the quality of the product. In the study, three vectors (lexicon vector, seed vector, and a third unspecified vector) and two partitions (*V*) per vector were used to determine the domain to which the item belongs. The lower approximation is illustrated in Fig. 5.

If X represents an article, we will use the proposed approach to determine the predicted class for the article. Article X includes a group of words. The length of those words is determined by two grams. $X = w_1, w_2, w_3, \dots w_n$, where n represents number of the words in an article that has been tested with the use of (5). This equation represents a lower approximation known as $w \in X$, every one of the words belongs to some article in X, and the number of the matches is going to be backed up

$$\underline{B}(X)_J = \{\# w \mid w \in X \text{ and } w \in U_j\}, \tag{5}$$

where there are # elements in the set. For the application of the test to such article X, it is necessary to perform a test on every one of the classes in U_j, then compare the numbers of the classes in the article. In such a case, (6) is used to determine maximum value in $\underline{B}(X)_J$

$$\Pr(X) = argmax\left(\underline{B}(X)_J\right), \tag{6}$$

where Pr is the expected category, and the highest value achieved from all classes is chosen. When getting close to n, the output of (6) varies from $0 \leq \underline{B}(X)_J \leq n$ when it becomes close to n, then $\underline{B}(X)_J$ has numerous words in U_j with X test article.

Accuracy approximation method. Two issues can be addressed using this approach. Dependence on the maximum value is the primary issue. The second is when there is little differentiation between the classes and picking the right one would be tough. The accuracy estimate is depicted in Fig. 5.

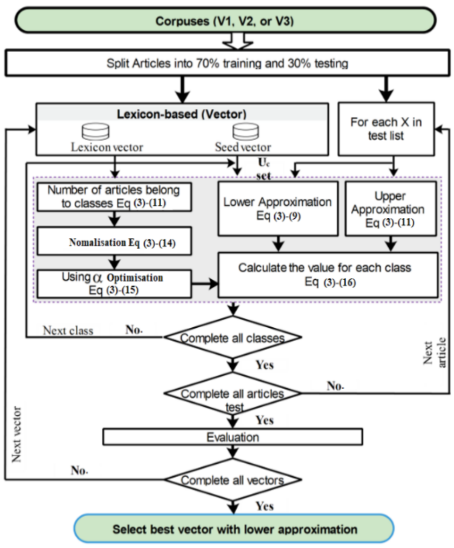

Fig. 5. Accuracy approximation method.

The issues mentioned above can be addressed by employing a lower approximation, higher approximation, and normalisation, all of which are illustrated in Fig. 5. In this approach, a more conservative approximation is produced directly from (6). Due to this, we must resort to the more precise upper estimate given by (7)

$$\overline{B}(X) = number\ of\ words\ w\ in\ article\ X. \tag{7}$$

The upper and lower approximations of the training set's article count will be completed and then, based on the P1 and P2 partitions, the training set's article count will be determined. After putting the articles in groups using (4), (8) can be used to figure out how many of each group there are in the training set:

$$\begin{bmatrix} \delta_{ij} = \begin{cases} 1, & A_i \in p_j, \\ 0, & otherwise, \end{cases} \\ N_j = \sum_{i=1}^{n} \delta_{ij}, \end{bmatrix}, \tag{8}$$

where δ_{ij} is used to collect 1 in the case where the article belongs to the P_j class, and N represents number of the articles in P_j training set. When calculating N_j value, multiply N_j value by the lower value of the approximation, and the equation will become as follows

$$ACC(X,N) = \frac{\underline{B}(X)_j \, N_j}{\overline{B}(X)}. \tag{9}$$

The N_j value should be normalised, as shown by the equation above, because the result of multiplying it by lower value of the approximation and dividing it by upper approximation value will potentially be $1 \leq N_j \leq 3$ one. Therefore, the accuracy obtained becomes very low. Normalisation has been shown in (10) below, where N_j ranges between 0 and $0 \leq N_j \leq 1$

$$N_j = \frac{N_j - argmin(N_j)}{argmax(N_j) - argmin(N_j)}. \tag{10}$$

An issue was revealed by (10). The issue is that the value of N_j will either be 0 or 1. In the case where N_j returns a value of 0, then the result of (9) is 0. However, if N_j returns a value of one, the result of (9) will be used in the original form, i.e., without normalisation. The solution to that problem is to use optimisation, as shown in (11)

$$N_j = \frac{N_j - argmin(N_j)}{argmax(N_j) - argmin(N_j)} \pm \alpha. \tag{11}$$

The range of the optimisation between 0 and 1 such as $0 \leq \alpha < 1$ and they will be dependent on whether $N_j - argmin(N_j)$ equals 0, then utilise plus (+), otherwise use minus (-). The final equation for the approximation of the accuracy will be as follows

$$ACC(X,N) = \frac{\underline{B}(X)_j \dfrac{N_j - argmin(N_j)}{argmax(N_j) - argmin(N_j)} \pm \alpha}{\overline{B}(X)}. \tag{12}$$

IV. EXPERIMENTS

In this part, we introduce two classes of approximated characteristics. This lower approximation was used for a system containing two vectors, such as a lexicon vector and a seed vector. The identical two vectors were also used in an accuracy estimate. In this investigation, two techniques were refined. In the first case, we have a lower approximation (LA), while in the second, we have an accuracy approximation (AA). The rough set theory underpins both approaches in our study. In terms of vectors, two were chosen using a lexicon-based approach. The lexicon vectors (L) and the seed vectors (S) are examples of such vectors (S). Combining the lexicon-based approach (LA) with the rough set theory (AA) approach (hybrid technique) yields a new approach. The speed of rough set comes from its use of set theory, particularly linear algebra.

The parameter and lower approximation from the rough set theory were utilised to enhance the AA and LA methods, respectively. Two lexicon-based vectors were used for both LA and AA to provide the most accurate results in the least time compared to future-state machine learning.

A. Lower Approximation

In this part, we introduce a crude set theory-based approximation technique for the next-to-best approximation. Two vectors, a lexicon vector and a seed vector, were employed in this process. The text from Twitter was used to train each vector. In Twitter parlance, orientation refers to the slant of a set of tweets tagged with a certain label. Predicting the polarity or label of a Twitter text was done using LA. To determine which classes the individual words in the text belong to, LA utilised the union between the text and the vector. We tested two possible orientations: radicalism and moderation.

Table III displays the application of LA with lexicon vector to three grammatical structures and three corpora to determine which structure and corpus the LA with lexicon vector performed best in. Table III shows that the unigram in the V2 corpus achieved high accuracy (90.853), while corpus V1 achieved (89.024) accuracy in the unigram, and corpus V3 achieved (89.024) accuracy in the unigram (83.536). Bigram in corpus V3 achieved (86.585); on the other hand, corpus V2 achieved the accuracy (81.707). Incorporating the accuracy of both corpuses V1 and V2, the trigram achieved (71.341) in corpus V3 (70.121). The vote for using LA with the lexicon vector was for corpus V3 three times, V2 once, and V1 not at all. LA with lexicon vector worked well in this case. V2 was the best in terms of Unigram, V2 and V3 were the best in terms of Bigram, and V3 was the best in terms of Trigram.

TABLE III. LEXICON VECTOR LOWER APPROXIMATION.

Number of grams	V1 %	V2 %	V3 %	Vote
Unigram	89.024	90.853	83.536	V2
Bigram	79.268	81.707	86.585	V3
Trigram	70.121	70.121	71.341	V3

Table IV is an illustration of how LA, seed vector, and second vector interact. This vector significantly improved accuracy. The lexicon vector was shown to work well with corpus V2 based on the results tabulated in Table III. However, when using the seed vector, the accuracy decreased from bigram to trigram. The unigram accuracy of the corpus V1 was 92.073. Even corpus V2 showed accuracy with unigram, it achieved 93.292, and 86.585 for corpus V3 respectively. The lowest value was achieved in corpus V1 with (73.780) in trigram. In contrast to the lexicon vector, which is constructed from a sequence of words, the seed vector was robust since it only involves single words. Because of this, the seed vector can pick out individual words, giving it a high level of accuracy in all three grammatical LOWER structures.

TABLE IV. LEXICON VECTOR LOWER APPROXIMATION.

Number of grams	V1 %	V2 %	V3 %	Vote
Unigram	92.073	93.292	86.585	V2
Bigram	82.926	85.975	88.414	V3
Trigram	73.780	77.439	77.439	V2 & V3

Table IV shows that in all corpora, the accuracy increased from unigram to trigram. The accuracy of the lexicon vector in Table III shows that there is no repeating of vectors during training for bigrams and trigrams. This means that the accuracy is stable from unigrams to bigrams. If there is even one word in the gram that is different between the test set of three grams and the vector set of three grams, then LA is not presented, making the lexicon vector the low vector with LA. The seed vector is constructed using unigrams and because the words are chosen by a human expert, it can be useful even if the entire text is provided as three grams and only one word in each gram comes from the seed vector. The LA selection process is distinct from the lexicon vector method. On the basis of testing data, the seed vector appears to be more accurate than the lexicon vector. However, the vote was low for the lexicon vector and the seed vector V1, and it was the same for the seed vector V2, and V3 corpora.

B. Accuracy Approximation

Here, we put the AA technique to derive α as shown in Table V. It demonstrates the AA approach. Table V shows the alpha parameters for a sample of training articles (70 %) that are used to illustrate the training process. The number of texts are assigned to each category number that can be used to improve vectors to find which one performed best. Parameter values for the AA technique are equalised.

TABLE V. ACCURACY APPROXIMATION PARAMETERS WITH OUTPUT VALUE.

Class	Normalisation	+	α	Value
Extremism	0	+	0.1	1.1
Nonextremism	1	+	1.1	1.1

Three vectors are used with the AA method when it receives the alpha parameter, as shown in Table V. The lexicon vector is the first vector used in AA. Table VI shows that the unigram scored 93.902 with corpus V2, the bigram scored 89.634 with corpus V3, and the trigram scored 81.097 with corpus V2. Here, corpus V2 has achieved 93.902 accuracy in unigram and 81.097 accuracy in trigram. Corpus V3 achieved an accuracy of (86.585) in bigram. In the lexicon vector with the AA method, the vote went to V2 twice and V3 once. However, the three grams achieved high accuracy in V1, V2, and V3 in general.

TABLE VI. LEXICON VECTOR ACCURACY APPROXIMATION.

Number of grams	V1 %	V2 %	V3 %	Vote
Unigram	92.682	93.902	87.195	V2
Bigram	83.536	86.585	89.634	V3
Trigram	75	81.097	78.048	V2

The usage of a seed vector by the AA technique is shown in Table VII. As can be seen in Table VI, when AA is combined with LA, the vector becomes extremely stable. Corpus V1's improvement reached 92.682, and corpus V2's reached 93.902. Corpora V2 and V3 were improved with the AA technique to address the issue of equal class value, as shown in Table VII. Corpus V2 also won the popularity poll for its unigram accuracy (93.902) and trigram accuracy (81.097).

TABLE VII. ACCURACY APPROXIMATION WITH THE SEED VECTOR.

Number of grams	V1 %	V2 %	V3 %	Vote
Unigram	94.512	94.512	89.024	V1 & V2
Bigram	86.585	88.414	90.243	V3
Trigram	76.829	83.536	80.487	V2

As shown in Tables VI to VII. The use of two vectors of the AA technique and seed vector, yielded satisfactory results. After bigram, the lexicon vector fell on hard times, whereas the seed vector overcame this challenge and rose in popularity. The lexicon vector performed well in experiments, but the seed vector, which makes use of light stemming, outperformed it on the V2 corpus in terms of speed and accuracy.

Both the lexicon vector and the seed vector from the LA and AA procedures were employed in this investigation. According to the data shown in Tables III through VII the AA approach was superior to LA because it was able to address the issue of equal class value. In comparison to the AA technique that uses an identical lexicon vector, the accuracy of the LA method was lower. It was also shown that the accuracy of LA-trained seed vectors was lower than that of identical vectors trained with the AA approach but that the AA method ultimately achieved better accuracy.

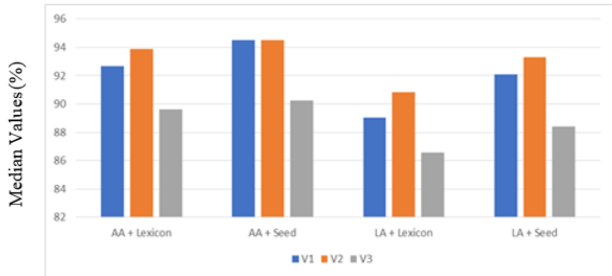

Fig. 6. Comparison between LA and AA methods with two vectors.

The median of the three grams of each corpus is shown in Fig. 6. which represents the two vectors for every procedure. Both approaches produced a lexicon vector with poorer precision than average. The seed vectors appeared similar for both approaches, but the AA output was better. Compared to other corpora, Corpus V3 appeared to perform worse in both approaches. When using the AA technique, the seed vector outperformed LA. Furthermore, the performance of corpus V3 was poorer than that of other corpuses when using either of the two approaches. AA outperformed LA here and may be implemented in either seed vector or lexicon vector settings.

C. The Hybrid Method for the Best Vector

Here, we implemented two vectors, a lexicon vector and a seed vector. The AA and LA procedures relied both on these vectors to determine polarity and make their best vector selections. Table VIII shows the two approaches and two vectors utilised with the three-gram types (unigram, bigram, and trigram). To draw parallels with machine learning, we settled on the three-gram level. The results of the vote might range from 0 to 3, with 3 being the most popular.

Both approaches resulted in a lexicon vector that was less than the seed vector, as shown in Table VIII. The LA procedure seed vector received 9 points and no votes. For LA, the best results were obtained using the V1, V2, and V3 corpora, all of which scored zero in the lexicon vector.

TABLE VIII. ACCURACY APPROXIMATION PARAMETERS WITH OUTPUT VALUE.

Corpus	LA		Vote	AA		Vote
	Lexicon vector (L)	**Seed vector (S)**		**Lexicon vector**	**Seed vector**	
V1	0	3	S	0	3	S
V2	0	3	S	0	3	S
V3	0	3	S	0	3	S
Total	0	9	S	0	9	S

On the contrary, the hybrid approach received 9 points overall but no support from the community. With no votes cast, the AA approach using a seed vector managed to get 9 points. Although the seed vector received no votes at all, it nevertheless managed to get a total of 9. In general, the seed vector is clearly superior to the lexicon vector, but in this situation, we can see that both are good.

D. Select the Best Hybrid Method

In this part, we compare LA and AA to machine learning to determine which is superior. The superior vector, the seed vector, was used for this purpose. Grams, unigrams, bigrams, and trigrams are selected using a two-vector approach, just like in machine learning. Although root stemming made corpus V3 the best for machine learning, we used all the corpuses in Table IX because our suggested approaches worked very well with all of them.

TABLE IX. SELECT OPTIMAL HYBRID METHOD.

Corpus	Vector	LA	AA	Vote
V1	Lexicon vector	0	3	AA
	Seed vector	0	3	AA
V2	Lexicon vector	0	3	AA
	Seed vector	0	3	AA
V3	Lexicon vector	0	3	AA
	Seed vector	0	3	AA
Total		0	18	AA

E. Applying the Proposed Method on another Corpus Benchmark

In this part, we put the suggested technique through its paces using a different corpus to get the best possible answer. The BBC News Data Set was utilised for this evaluation. Information about University College Dublin (UCD) is available for further exploration. Since the data was culled from BBC news websites in 2004, it has been published in English. In Table X, we can see that our data set consisted of 5 distinct categories.

TABLE X. THE TRAINING AND TESTING FOR BBC DATA SET.

Class	Training %70	Testing %30	Total
Business	357	153	510
Entertainment	235	101	336
Politics	292	125	417
Sport	358	153	511
Tech	281	120	401
Total	1523	652	2175

With a total of 2175 entries, this data set contains more records than any of our Political Arabic Articles Dataset (PAAD) data sets. To improve the performance of the hybrid technique suggested in this data set, we combined the lower approximation method with the integration of the lexicon vector.

The output of using LA with a lexicon based on the raw data set is displayed in Table XI. A total of 96.706 % accuracy was found in all categories throughout the study. We can see that there is a separate F-score, accuracy, and recall for each category. In the Tech category, the recall scored 0.99, in the Entertainment category, it scored 1.00, and in the Sports category, it scored 0.98. Overall, the level of accuracy was rather high. Using unigram, we were able to evaluate how well our algorithm performed with a new corpus, a new language, a large number of articles, and a significant corpus size (2225).

TABLE XI. THE ACCURACY OF THE PROPOSED METHOD FOR UNIGRAM.

Class	Precision	Recall	F-score	Accuracy %
Business	0.95	0.95	0.95	96.706
Entertainment	1	0.94	0.97	
Politics	0.96	0.97	0.96	
Sport	0.99	0.98	0.99	
Tech	0.94	0.99	0.97	

V. CONCLUSIONS

Methods for dealing with the lower approximation (LA) and the accuracy approximation (AA) were addressed using a crude set theory-based approach. The lexicon vector and the seed vector were utilised in this study. Three grams were employed in the lexicon vector, seed vector, and a human-based. Additionally, our approach was tested on three different corpora. Based on the above comparison, it is clear that the AA technique performed well with the lexicon vector but performed much better with the seed vector.

Our study found that political discourse tends to fall into one of two categories: extremism or nonextremism. One portion of this data was emotion-tagged so that it could be studied in its entirety by analysing user posting patterns in distinct cohorts. Then a procedure was developed to determine the orientations of the Tweet texts.

In contrast to machine learning, which operates with numbers, we used a vector in the form of words. Lower approximation and greater accuracy approximation were found to be best achieved by using the lexicon vector and the seed vector. Application to the corpus confirmed the usefulness of the proposed technique. The following are some inferences that may be drawn from the findings of this study:

– Compared to traditional methods, the hybrid approach performed better across the board, but particularly well with the V1 and V2 corpuses;

– The problems that were seen with machine learning (zero correlation and low accuracy) were solved by the suggested hybrid technique, which uses both rough set theory and lexicon-based methods;

– Researchers found that the zero-relation problem of TF and TF-IDF feature extraction may be overcome by using two vectors (lexicon vector and seed vector) in a lexicon-based approach;

– The study also showed that the ensemble vector and the seed vector were superior to the lexicon vector in terms of accuracy and precision;

– Using precision approximation with an alpha parameter helped to get around the equal value and high value limits of the lower approximation method;

– Recent research improved the value selection method for future polarity work. Several other techniques, such as cuckoo search, particle swarm optimisation, and the firefly algorithm, may also automatically choose the value, but their slow pace makes them cumbersome to work with.

CONFLICTS OF INTEREST

The authors declare that they have no conflicts of interest.

REFERENCES

[1] K. M. Quinn, B. L. Monroe, M. Colaresi, M. H. Crespin, and D. R. Radev, "How to analyze political attention with minimal assumptions and costs", *American Journal of Political Science*, vol. 54, no. 1, pp. 209–228, 2010. DOI: 10.1111/j.1540-5907.2009.00427.x.

[2] S. Wu, F. Wu, Y. Chang, C. Wu, and Y. Huang, "Automatic construction of target-specific sentiment lexicon", *Expert Systems with Applications*, vol. 116, pp. 285–298, 2019. DOI: 10.1016/j.eswa.2018.09.024.

[3] D. H. Abd, A. T. Sadiq, and A. R. Abbas, "Classifying political Arabic articles using support vector machine with different feature extraction", in *Applied Computing to Support Industry: Innovation and Technology. ACRIT 2019. Communications in Computer and Information Science*, vol. 1174. Springer, Cham, 2020, pp. 79–94. DOI: 10.1007/978-3-030-38752-5_7.

[4] S.-t. Wu, Y. Li, and Y. Xu, "Deploying approaches for pattern refinement in text mining", in *Proc. of Sixth International Conference on Data Mining (ICDM'06)*, 2006, pp. 1157–1161. DOI: 10.1109/ICDM.2006.50.

[5] L. Zhang, Y. Li, C. Sun, and W. Nadee, "Rough set based approach to text classification", in *Proc. of 2013 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT)*, 2013, pp. 245–252. DOI: 10.1109/WI-IAT.2013.190.

[6] Y. Y. Yao and S. K. M. Wong, "A decision theoretic framework for approximating concepts", *International Journal of Man-Machine Studies*, vol. 37, no. 6, pp. 793–809, 1992. DOI: 10.1016/0020-7373(92)90069-W.

[7] Y. Li, C. Zhang, and J. R. Swan, "An information filtering model on the Web and its application in JobAgent", *Knowledge-Based Systems*, vol. 13, no. 5, pp. 285–296, 2000. DOI: 10.1016/S0950-7051(00)00088-5.

[8] L. Zhang, "Modelling uncertain decision boundary for text classification", PhD dissertation, Queensland University of Technology, 2016.

[9] S. M. Al-Ghuribi, S. A. M. Noah, and S. Tiun, "Unsupervised semantic approach of aspect-based sentiment analysis for large-scale user reviews", *IEEE Access*, vol. 8, pp. 218592–218613, 2020. DOI: 10.1109/ACCESS.2020.3042312.

[10] B. A. H. Murshed, J. Abawajy, S. Mallappa, M. A. N. Saif, S. M. Al-Ghuribi, and F. A. Ghanem, "Enhancing big social media data quality for use in short-text topic modeling", *IEEE Access*, vol. 10, pp. 105328–105351, 2022. DOI: 10.1109/ACCESS.2022.3211396.

[11] G. Wolfsfeld, E. Segev, and T. Sheafer, "Social media and the Arab Spring: Politics comes first", *The International Journal of Press/Politics*, vol. 18, no. 2, pp. 115–137, 2013. DOI: 10.1177/1940161212471716.

[12] Z. Pawlak, "Rough sets, decision algorithms and Bayes' theorem", *European Journal of Operational Research*, vol. 136, no. 1, pp. 181–189, 2002. DOI: 10.1016/S0377-2217(01)00029-7.

[13] Z. Pawlak and A. Skowron, "Rudiments of rough sets", *Information Sciences*, vol. 177, no. 1, pp. 3–27, 2007. DOI: 10.1016/j.ins.2006.06.003.

[14] Y. Yao, "Three-way decision: An interpretation of rules in rough set theory", in *Rough Sets and Knowledge Technology. RSKT 2009. Lecture Notes in Computer Science()*, vol. 5589. Springer, Berlin, Heidelberg, 2009, pp. 642–649. DOI: 10.1007/978-3-642-02962-2_81.

[15] Y. Yao, "Three-way decisions with probabilistic rough sets", *Information Sciences*, vol. 180, no. 3, pp. 341–353, 2010. DOI: 10.1016/j.ins.2009.09.021.

[16] A. Schwering and M. Raubal, "Spatial relations for semantic similarity measurement", in *Perspectives in Conceptual Modeling. ER 2005. Lecture Notes in Computer Science*, vol. 3770. Springer, Berlin, Heidelberg, 2005, pp. 259–269. DOI: 10.1007/11568346_28.

[17] M. S. Khorsheed, "Off-line Arabic character recognition – A review", *Pattern Analysis Applications*, vol. 5, pp. 31–45, 2002. DOI: 10.1007/s100440200004.

[18] A. S. Fadel, M. E. Saleh, and O. A. Abulnaja, "Arabic aspect extraction based on stacked contextualized embedding with deep learning", *IEEE Access*, vol. 10, pp. 30526–30535, 2022. DOI: 10.1109/ACCESS.2022.3159252.

[19] S. M. Al-Ghuribi and S. Alshomrani, "A simple study of webpage text classification algorithms for Arabic and English languages", in *Proc. of 2013 International Conference on IT Convergence and Security (ICITCS)*, 2013, pp. 1–5. DOI: 10.1109/ICITCS.2013.6717784.

[20] B. Brahimi, M. Touahria, and A. Tari, "Data and text mining techniques for classifying Arabic tweet polarity", *Journal of Digital Information Management*, vol. 14, no. 1, pp. 15–25, 2016.

[21] M. Galal, M. M. Madbouly, and A. El-Zoghby, "Classifying Arabic text using deep learning", *Journal of Theoretical Applied Information Technology*, vol. 97, no. 23, pp. 3412–3422, 2019.

[22] M. Saad and W. Ashour, "Arabic text classification using decision trees", in *Proc. of the 12th International Workshop on Computer Science and Information Technologies CSIT'2010*, 2010, pp. 75–79.

[23] S. H. Ghwanmeh, "Applying clustering of hierarchical K-means-like algorithm on Arabic language", *International Journal of Computer Information Engineering*, vol. 1, no. 8, pp. 2396–2400, 2007. DOI: 10.5281/zenodo.1327445.

[24] S. M. Al-Ghuribi and S. Alshomrani, "BI-languages mining algorithm for classifying text documents (BILTc)", *International Journal of Academic Research*, vol. 6, no. 5, pp. 16–25, 2014. DOI: 10.7813/2075-4124.2014/6-5/A.3.

[25] E. Abuelyaman, L. Rahmatallah, W. Mukhtar, and M. Elagabani, "Machine translation of Arabic language: Challenges and keys", in *Proc. of 2014 5th International Conference on Intelligent Systems, Modelling and Simulation*, 2014, pp. 111–116. DOI: 10.1109/ISMS.2014.25.

[26] A. Al-Hassan and H. Al-Dossari, "Detection of hate speech in social networks: A survey on multilingual corpus", in *Proc. of 6th International Conference on Computer Science and Information Technology*, 2019, pp. 83–100. DOI: 10.5121/csit.2019.90208.

[27] I. Aljarah *et al.*, "Intelligent detection of hate speech in Arabic social network: A machine learning approach", *Journal of Information Science*, vol. 47, no. 4, pp. 483–501, 2021. DOI: 10.1177/0165551520917651.

[28] A. H. Johnston and G. M. Weiss, "Identifying sunni extremist propaganda with deep learning", in *Proc. of 2017 IEEE Symposium Series on Computational Intelligence (SSCI)*, 2017, pp. 1–6. DOI: 10.1109/SSCI.2017.8280944.

[29] S. Ahmad, M. Z. Asghar, F. M. Alotaibi, and I. Awan, "Detection and classification of social media-based extremist affiliations using sentiment analysis techniques", *Human-centric Computing Information Sciences*, vol. 9, art. no. 24, pp. 1–23, 2019. DOI: 10.1186/s13673-019-0185-6.

[30] K. T. Mursi, M. D. Alahmadi, F. S. Alsubaei, and A. S. Alghamdi, "Detecting Islamic radicalism arabic tweets using natural language processing", *IEEE Access*, vol. 10, pp. 72526–72534, 2022. DOI: 10.1109/ACCESS.2022.3188688.

[31] C. Sofat and D. Bansal, "Radscore: An automated technique to measure radicalness score of online social media users", *Cybernetics and Systems*, vol. 54, no. 4, pp. 406–431, 2023. DOI: 10.1080/01969722.2022.2059134.

[32] M. S. A. Sanoussi, C. Xiaohua, G. K. Agordzo, M. L. Guindo, A. M. Al Omari, and B. M. Issa, "Detection of hate speech texts using machine learning algorithm", in *Proc. of 2022 IEEE 12th Annual Computing and Communication Workshop and Conference (CCWC)*, 2022, pp. 0266–0273. DOI: 10.1109/CCWC54503.2022.9720792.

[33] S. M. Al-Ghuribi, S. A. Noah, and S. Tiun, "Various pre-processing strategies for domain-based sentiment analysis of unbalanced large-scale reviews", in *Proceedings of the International Conference on Advanced Intelligent Systems and Informatics 2020. AISI 2020. Advances in Intelligent Systems and Computing*, vol. 1261. Springer, Cham, 2021, pp. 204–214. DOI: 10.1007/978-3-030-58669-0_19.