Ahmed, AA, Hasan, MK, Jaber, MM, Al-ghuribi, SM, Abd, DH, Khan, W, Sadiq, AT and Hussain, A

Extremism Arabic Text Detection using Rough Set Theory: Designing a Novel Approach

http://researchonline.ljmu.ac.uk/id/eprint/19599/

Article

Citation (please note it is advisable to refer to the publisher’s version if you intend to cite from this work)


LJMU has developed LJMU Research Online for users to access the research output of the University more effectively. Copyright © and Moral Rights for the papers on this site are retained by the individual authors and/or other copyright owners. Users may download and/or print one copy of any article(s) in LJMU Research Online to facilitate their private study or for non-commercial research. You may not engage in further distribution of the material or use it for any profit-making activities or any commercial gain.

The version presented here may differ from the published version or from the version of the record. Please see the repository URL above for details on accessing the published version and note that access may require a subscription.

For more information please contact researchonline@ljmu.ac.uk
Arabic Text Detection using Rough Set Theory: Designing a Novel Approach

Amjad Abbas Ahmed1,2, Mohammed Kamrul Hasan1, Senior, IEEE, Mustafa Musa Jaber3, Sumaia Mohammed Al-ghuribi1,4 Dhafar Hamed Abd5, Wasiq Khan6, Senior, IEEE, Ahmed Tareq Sadiq7, Abir Hussain8, Senior, IEEE

1Center for Artificial Intelligent Technology, Faculty of Information Science and Technology, Universiti Kebangsaan Malaysia, Bangi 43600, Malaysia
2Imam Al kadhum college (IKC), Department of Computer Science, University of Technology, Baghdad, Iraq
3Department of Medical Instruments Engineering techniques, Dijlah University College, Baghdad, 10021, Iraq
4Department of Computer Science, Faculty of Applied Sciences, Taiz University, Taiz 6803, Yemen
5Computer Science Department, College of Computer Science & Information Technology, University of Anbar, Anbar, Iraq
6Computer Science, Liverpool John Moores University, Liverpool, L33AF, UK
7Department of Computer Sciences, University of Technology, Baghdad, Iraq

Corresponding author: mustafa.musa@duc.edu.iq

ABSTRACT The linguistics related research and particularly, sentiment analysis using data-driven approaches, has been growing in recent years. However, the large number of users and excessive amount of information available on social media, make it difficult to detect extremism text on these platforms. The literature revealed a plethora of research studies focusing the sentiment analysis primarily, for English texts, however, very limited studies are available concerning the Arabic language which is the 4th mostly spoken language in the world. We first time in this study, propose a text detection mechanism for extremism orientations distinction in Arabic language, to improve the comprehension of subjective phrases. The study introduces a novel method based on Rough Set theory to enhance the accuracy of selected models and recognize text orientation reliably. Experimental outcomes indicate that the proposed method outperforms existing algorithms by contributing towards feature discriminations. Our method achieved 90.853%, 81.707% and 71.951% accuracies for unigram, bigram, and trigram representations, respectively. This study significantly contributes to the limited research in the field of machine learning and linguistics in Arabic language.

INDEX TERMS Sentiment analysis, Arabic language sentiment, extremism analysis, social media sentiment, Rough Set Theory, Arabic text mining

I. INTRODUCTION

Sentiment analysis, also known as sentiment orientation or opinion mining, is the process of determining the orientation of unstructured data. It is essentially a categorization exercise, where the mood or point of view expressed in a sentence or article is classified as negative, positive, or neutral. Because most of the research works in this topic focus on the English language, there is a research gap in the domain of sentiment analysis for Arabic language. Moreover, Arabic language processing and analysis significantly varies from English and has its own set of challenges and obstacles [1]. In this context, it is important to note that the Arabic language is ranked number four of the most widely spoken language globally. Arabic is spoken as a first language by over one billion of the population and as a second language by 250 million people. There are 28 alphabets in Arabic language that are unlike English letters, lack upper and lower cases, written reversely (i.e., from right to left), in terms of orientation. There have been very limited studies on Arabic language sentiments analysis (ALSA), attitudes, emotions, and opinions compared to the excessive works on the English text sentiment analysis [2]. Most of the prevailing works for the ALSA focus on specific material, such as review text or classifying positive and negative text contents. In contrast, proposed research aims to use extremism orientation recognition over a primary dataset gathered from public Tweets, potentially comprising Arabic text exhibiting extremism. Generally, sentiment analysis can be categorized into three levels: sentence, aspect, and document. We present sentence level ALSA, specifically to classify an opinion sentence as either extremist or non-extremist [3] where the entire sentence is regarded as a primary information unit in our framework. The detection is based on rough set theory and lexical analysis. Our core
method can be further divided into three components. The first component gathers the statistical aspects of the corpus, including the total number of texts, words, punctuation marks, and unique words. In the second component, distinct vector copies are created for the two lexicon-based and machine learning (ML) approaches. In the third component, detailed experiments are performed using a hybrid model, combining Rough Set (RS) theory [30], lexicon-based approaches [13-14], and ML algorithms for orientation and categorization [24-29]. To overcome the limitations of previous works particularly a) limited studies on extremism for ALSA, b) manual lexicon extraction, we propose automated lexicon construction, combined with the use of RS theory, to address the issue of low accuracy under sentimentality analysis. Major contributions for the proposed study include:

1- Primary dataset containing extremism in political and religion contexts. In total, 44007 texts were collected and annotated (21502 for extremism and 22505 for non-extremism).
2- Used lexicon to build vector which can be then used to perform the classification task.
3- Used RS theory and ML to classify the Arabic text into extremism or otherwise.
4- First time enhance the RS theory by using accuracy approximation that helps to enhance the ALSA performance.

The objective of our research is as follows:

- To develop new model for the detection of Arabic extremism in Twitter platform.
- To design new lexicon-based method for Arabic extremism using human-based method and word frequency.
- To design a new model based on the rough set theory and lexicon-based method to enhance the extremism detection.
- To verify and evaluate the performance of our proposed model.

The remainder of this manuscript is organized as follows. Section II presents the related works; while Section III shows basic concept of RS theory and. Section IV provides the details for our proposed ALS method, and Section V shows the statistical results from the experiments and discussions. Finally, Section VI presents the conclusion drawn from this study and recommendations for future direction.

II. RELATED WORKS

We are living in the age of social networking (SN) which is growing at a surpassing rate. SNs are digital environments where users can communicate, interact, and share information (e.g., beliefs, and ideologies etc.) [6]. One of the most popular, fast-spreading and micro-blogging services of SN is Twitter [7]. Apparently, SN results in a massive amount of user generated data that forms a rich resource for conducting research and building valuable knowledge. However, some users exploit the Twitter for propagating extremist and discrimination ideas which lead to the dissemination of hate speech or hate crime [8]. Extremism is a complex incident which can be applied in various scenarios e.g., hate group, racist communities, Jihad terrorism, and personal insult, use of abusive language, propagating obscene or extremist videos, and more[9]. Extremism groups communicate with each other on Twitter and sharing information to hire new members by gradually reaching a worldwide audience that helps persuade others to commit violence and terrorism [10]. There is limited number of studies on the problem of extremism [11]. Extremism has many types such as supremacism, sectarianism, nationalism, and etc. [12] however, there is no studies on religiously and political motivated extremism [12]. This is the lacuna which the proposed study seeks to fill, the exploration of extremism in religion and political.

Lexicon-based methods use several words or phrases, which are considered an important resource when dealing with sentiment analysis [13, 14]. There are various approaches for lexicon construction, including manually [15] and automatically [16]. Manual lexicon extraction is costly particularly, it is impractical for big data and lacks in terms of generalization to other domains, and therefore, automated lexicon extraction has become a popular research topic [17]. There are some works on sentiment orientation word detection which are dependent on lexicon construction [18]. Despite the substantial research advances in this field, several limitations are associated with state-of-the-art methods. Firstly, some methods utilize manually defined lexicons [19], making it inconvenient to transfer their techniques to other domains. Secondly, a lexicon does not provide high accuracy and performance by itself [20]. In this regard, Pawlak [21] used RS theory as a mathematical tool to deal with uncertain, vague, and inexact information, which led numerous researchers to pursue further theoretical developments and applications [22]. In data analysis, a major benefit of RS theory is the fact that there is no need for prior information regarding the data itself [23].

Rbooraig et al [3] proposed a new method for automatic categorization of Arabic articles based on political orientation. The method started with collecting texts for building a corpus, then studying the performance of various feature reductions. They utilized the two most popular feature extraction techniques; traditional text (TC) and stylometric (SF). The authors used six algorithms, which are: Naive Bayes (NB), Discriminative Multinomial Naïve Bayes (DMNB), Sparse generative model (SGM), Support Vector Machine (SVM), Random Forest (RF), and Mixed classifiers (VOTE). Their results indicate that the highest accuracy was obtained using TC.

Al-Radaideh et al. [24] proposed a new method for Arabic text categorization using term weighting and multiple reductions. This method uses term weight to extract the weight from the text; it then uses RS theory to reduce the number of terms that are used for generating classification rules. In their study, a quick reduction algorithm was proposed with multiple reductions to generate the set of
classification rules that represent the RS classifier. An Arabic corpus comprising 2700 documents with nine categories, was used to evaluate the classification algorithm. The experimental results revealed that the method achieved higher accuracy compared to K-nearest neighbor (KNN) and decision tree (DT) algorithms.

Waqs Sharif et al [25] proposed a new model that uses principal component analysis (PCA) for the dimensionality reduction and frequency-inverse document frequency (TF-IDF) for feature extraction. Different ML algorithms were used including KNN, SVM, RF, NB, and ensemble classification algorithms trained and evaluated over twitter text. The results showed that SVM achieved higher accuracy than other algorithms (84%).

### Table I

<table>
<thead>
<tr>
<th>Reference</th>
<th>Year</th>
<th>Feature</th>
<th>Data Set Source</th>
<th>Performance</th>
</tr>
</thead>
<tbody>
<tr>
<td>[3]</td>
<td>2018</td>
<td>Lexicon features word-based CHARACTER-BASED</td>
<td>DIFFERENT POLITICAL WEBSITES CONSIST OF 3000 DATA</td>
<td>ACC = 90.7% FOR SVM</td>
</tr>
<tr>
<td>[24]</td>
<td>2019</td>
<td>TF-IDF</td>
<td>DIFFERENT SOURCES CONSIST OF 2700 DATA</td>
<td>ACC = 94% FOR ROUGH SET THEORY</td>
</tr>
<tr>
<td>[25]</td>
<td>2019</td>
<td>TF-IDF</td>
<td>TWITTER</td>
<td>ACC = 84% FOR SVM</td>
</tr>
<tr>
<td>[26]</td>
<td>2020</td>
<td>TF-IDF</td>
<td>VKONTAKTE</td>
<td>ACC = 83% FOR RF</td>
</tr>
<tr>
<td>[27]</td>
<td>2020</td>
<td>TF-IDF</td>
<td>FACEBOOK</td>
<td>F-SCORE = 0.81% FOR SVC</td>
</tr>
<tr>
<td>[28]</td>
<td>2020</td>
<td>-</td>
<td>TWITTER</td>
<td>ACC = 82.6% FOR SVM</td>
</tr>
<tr>
<td>[29]</td>
<td>2021</td>
<td>TF-IDF n-gram word2vec</td>
<td>TWITTER</td>
<td>ACC = 97.29% FOR SVM</td>
</tr>
</tbody>
</table>

Mussiraliyeva et al [26] proposed detection of extremism in communication and monitoring the behavior and forecasting of threats emanating from individual users, groups, and network communities, that generate and distribute terrorists and extremists information on the internet. The study used four ML techniques which are SVM, multinomial naive bayes (MNB), RF, and logistic regression (LR) with two feature extraction TF-IDF and Word2Vec. The accuracy was higher with RF (83%).

Asif et al [27] proposed extremism classification approach into four categorize that include high extreme, low extreme, moderate, and neutral, based on their level of extremism. They created lexicon with the intensity weights validating from domain experts leading to attaining 88% accuracy for validation. Subsequently, MNB and SVM algorithms are employed for classification purposes. Overall, on the underlying multilingual dataset, SVM outperformed with an accuracy of 82%.

Fraiwan et al. [28], collected, analyzed and classified Twitter data from affiliated members of ISIS, as well as sympathizers. The authors used ML classification algorithms to categorize the tweets as terror-related, generic religious, and unrelated. The authors report the classification accuracy of KNN, Bernoulli Naive Bayes (BNB) and SVM [one-against-all (OAA) and all-against-all (AAA)] algorithms producing F1 score of 83%.

Aldera et al [29], used dataset that published between 2011 and 2021 and used different ML algorithms including LR, SVM, MNB, RF, and bidirectional encoder representations from transformers (BERT). Different feature extraction approaches were used including n-gram, word2vec, and term frequency-inverse document frequency. The study revealed that SVM with TF-IDF achieved highest accuracy (97.29%).

Table 1 presents some state-of-the-art works addressing ASLA using machine learning classifiers and rough set theory. The literature presents different approaches towards SA including machine learning and rough set theory; however, it can be noticed that limited works are available for extremism ALSA. Likewise, there is a lack of automatic lexicon-based methods.

Based on the literature review, we have identified the following limitations:

- Researchers are using term frequency and Word2vec for feature extractions, which is different in this study in which we are using lexicon vector for improved accuracy.
- The stat of the art utilized well know machine learning algorithms for their analysis. In this study, we are using updated concept in rough set theory.
- To the best knowledge of the authors, there is no research related to Arabic extremism detection for religion and political.
- Lack definition of extremist in internet activity.
- The constant evolution of behaviors associated with online extremism in order to avoid being detected by the developed algorithms (changes in terminology, creation of new accounts, etc.).
- The lack of data validation methods.

This research addresses these challenges and presents dataset with multi-ideology (religion and political) and binary-class (extremism and non-extremism).

### III. AN EXPOSITION OF ROUGH SET THEORY

Rough set theory was introduced by Pawlak (1982) [30] as an intelligent mathematical approach for handling uncertainty and incompleteness in data. It applies the concept of set representations, estimation space, and lower-case and upper-case estimates in a set. A crucial advantage of RS theory is the process of reducing the required number of features [31]. Some of the attributes are eliminated using the concept of dependence degree in RS theory, in which the
smaller set of attributes has a similar level of dependency as the original set of data.

An information system (IS) signifies understanding of RS, symbolized as 4-tuple, i.e., IS = <U, A, V, F>. In this notation, U denotes the sealed universe, a limited set about the number n items \{x_1, x_2, . . . , x_n\}. A represents a limited group of characteristics \{a_1, a_2, . . . , a_n\}, A = \{C \cup D\}, which might be divided into C and D, whereby C shows the condition attributes and D indicates a group of choice features [17]. Moreover, V = \bigcup_{a \in A} V_a, where V_a represents the domain of attribute a, and f: U \times A \rightarrow V symbolizes the overall choice function represented like information function in which f(x, a) \in V_a for individually a \in A, XU.

In RS theory, both upper and lower estimates are considered as primary operations in which X \subseteq U. In contrast to feature set R \subseteq A, X could further be determined using both upper and lower estimates. Besides, lower estimation of X is the group of items of U which are definitely in X

\[
x \in U: [x]_R \subseteq X = R(X)
\]  

(1)

The set of U articles that could be in X is the upper estimate of X

\[
\{x \in U: [x]_R \cap X \neq \emptyset\} = (X)R
\]  

(2)

The accuracy approximation X can be considered arithmetically as:

\[
R(X) = ((X) - R)/((X) - R)
\]  

(3)

This is used to estimate the quality of the approximation.

IV. PROPOSED METHOD

This section presents the overall ALSA methodology proposed in our work.

In this study, we compile a labelled corpus of extremism text in Arabic, where we define various forms of the vector (lexicon-vector and seed-vector) for lexicon-based and ML methodologies. We then conducted experiments using a hybrid approach utilizing ensemble lexicon-based and RS theory and machine knowledge approaches for orientation recognition of text divergence.

Figure 1 presents a summary of the hybrid system utilizing RS model and lexicon model in the proposed work.

![Figure 1. Sequential processing in the proposed ALSA approach comprising i) pre-processing, ii) dataset splitting, iii) lexicon-vector, iv) classification.](image_url)

Figure 1 displays the overall sequential processing of the proposed extremism classification, mainly including (a) data preparation, (b) feature extraction, and (c) RS theory. A comprehensive explanation of the components is presented in the following sections.

A. Dataset Explanation

This section provided details about our dataset for ALSA extremism classification.

One of the major challenges of online extremism search in Arabic posts, is the unavailability of a standard Arabic dataset. In this study, we collected a primary dataset from Twitter recourse, the Extremism Arabic Post Dataset (EAPD), consisting of 44007 texts, which is not available online. Texts in EAPD are classified into two categories, extremism with 21502 posts and the remaining are non-extremism. It should be noted that dataset entries were selected to avoid potential bias due to class imbalance. Extremism posts comprise any script in the procedure of articles on extremism occasions, printed by authors or essayists, frequently related to governmental orientation. The assemblage of articles includes articles printed in current normal Arabic and slang Arabic; however entire vernacular Arabic articles were removed. Approximately, 60% of the composed corpuses were uploaded on termination of the Arab Spring Revolt, but 30% of the information was gathered earlier. The remaining 10% of total articles were related to specific beliefs before the Arab Spring Revolution.

Initially, we used an Application Programming Interface (API) tool [47] to extract posts from the Twitter website. We manually excluded spam articles, with valid posts subsequently classified into extremism and non-extremism. Each category is represented by a folder containing the text files (i.e., post text), chronologically totaled. Entire documents are distinctive and reserved in raw arrangement,
i.e., restricted from washing, stemming, and other kinds of preprocessing. The labeling (annotation) process is then performed with careful consideration of any imprecision or error which can reduce the overall quality of the dataset, and may lead to misclassifications [48-50].

Table II (A) summarizes the training and testing datasets used in our experiments. The two classes, i.e., extremism and non-extremism are well balanced, and thus, we do not envisage any issues related to class bias. The original corpus was partitioned into 70% for training and 30% for testing [43]. While Table II (B) shows the properties of our dataset.

<table>
<thead>
<tr>
<th>Class</th>
<th>Training number</th>
<th>Testing number</th>
</tr>
</thead>
<tbody>
<tr>
<td>Extremism</td>
<td>15051</td>
<td>6451</td>
</tr>
<tr>
<td>Non-Extremism</td>
<td>15753</td>
<td>6752</td>
</tr>
<tr>
<td>Total</td>
<td>30804</td>
<td>13203</td>
</tr>
</tbody>
</table>

B. Dataset Details and Properties

Once the data is compiled to noise free and standardized form, we build the lexicon for each text class to be fed to the Lexicon and RS Theory (LRST) model. Consider \( n \) to be the number of posts, \( n \) represent the number of labels, and \( w \) is the number of words in the text. Let \( X \) represents the collection of articles where \( X = \{ A_i | A_i is an article, i \in Z \} \) and \( C \) is the collection of labels for each of the articles, \( C = \{ l_j | l_j is label of the article \} \). M. makes a partition on \( X \) such that \( A_i \in l_j \) for the same \( j \). When \( A_i \in l_j \), we refer to \( A_i \) by \( A_{ij} \).

Assume \( V \) to be the lexicon, which will be constructed for each article \( X \) in the label set \( C \) as shown in Equation (4):

\[
P_{ext} = \bigcup_{i} A_{i}^{ext} | A_{i}^{ext} \in l_{ext}
\]

\[
P_{non-ext} = \bigcup_{i} A_{i}^{non-ext} | A_{i}^{non-ext} \in l_{non-ext}
\] (4)

Equation (4) makes the partition such that every article must exactly belong to partition. In general, \( P_j \cup A_{ij} \). Where the number of articles belong to \( j \) [4]. We construct vector \( V \) for each class in \( l \) as shown in Equation (5) which will build the lexicon for each class.

\[
V_j = \{w | w \in P_j, 1 \leq j \leq n\}
\] (5)

D. Proposed Accuracy Approximation (AA) Method

This section presents the proposed model for accuracy approximation.

Figure 2 summarizes the proposed model for accuracy approximation.

By removing unnecessary error which can reduce the overall quality of the dataset, and may lead to misclassifications [48-50].

Table II (A) summarizes the training and testing datasets used in our experiments. The two classes, i.e., extremism and non-extremism are well balanced, and thus, we do not envisage any issues related to class bias. The original corpus was partitioned into 70% for training and 30% for testing [43]. While Table II (B) shows the properties of our dataset.

<table>
<thead>
<tr>
<th>Class</th>
<th>Training number</th>
<th>Testing number</th>
</tr>
</thead>
<tbody>
<tr>
<td>Extremism</td>
<td>15051</td>
<td>6451</td>
</tr>
<tr>
<td>Non-Extremism</td>
<td>15753</td>
<td>6752</td>
</tr>
<tr>
<td>Total</td>
<td>30804</td>
<td>13203</td>
</tr>
</tbody>
</table>

B. Pre-processing

This section provided details about the pre-processing method utilized in our proposed methodology.

Before lexicon generation (as shown in Fig. 1), it is important to perform the preprocessing on raw text data, to improve classification accuracy of the sentiment text based on the presented information while reducing the time taken to detect its orientation. This includes several operations comprising handling the punctuations errors, tokenization, stop-words, normalization, and curtailing [34]. Tokenization of text to words is the initial stage, followed by removing Arabic punctuations, using specific constraints, for instance, removing punctuations with a token dimension of less than 2.

Standardization in SA is important for removing unnecessary information, resulting in a standardized writing style in the Arabic language; in this research, we engaged in a variety of normalization procedures. We removed diacritics such as \( \{\vec{\text{،}}\vec{\text{،}}\vec{\text{،}}\vec{\text{،}}\vec{\text{،}}\vec{\text{،}}\vec{\text{،}}\vec{\text{،}}\vec{\text{،}}\vec{\text{،}}\vec{\text{،}}\} \). The “tawwif”; a form of Arabic writing; was removed since it creates issues with the length of the word, which could appear to be longer by four times its standard length due to the inclusion of flat line parts, i.e., \( \{\vec{\text{،}}\vec{\text{،}}\vec{\text{،}}\vec{\text{，}}\vec{\text{，}}\vec{\text{،}}\vec{\text{،}}\vec{\text{،}}\vec{\text{،}}\vec{\text{،}}\vec{\text{،}}\} \). This aspect in the text data may be frequent, thus complicating processing and potentially impacting the accuracy of the classification [35, 36]. The final step was to substitute some letters with their universal (standard) form. For instance, in Arabic, the letter ‘Alif’, which may be found in various forms, e.g., \( \{\vec{\text{،}}\vec{\text{،}}\vec{\text{،}}\vec{\text{،}}\vec{\text{،}}\vec{\text{，}}\vec{\text{،}}\vec{\text{،}}\vec{\text{،}}\vec{\text{،}}\vec{\text{،}}\} \), is normalized to Alif (‘Alif’). On the other hand, “Ya” (‘Ya’) is the standard for the letter ‘Alif-Maqsura’ (‘Ya’), while the Arabic letter (‘Ya’ or ‘Maqsura’) becomes (‘Ya’), and the letter (‘Ya’) is normalized to (‘Ya’).

Removing stop-words helps in reduction of unnecessary words while reducing the length and improving the compactness and efficiency of the feature vector. For instance, we removed sundered Arabic stop words defined in Python’s Natural Language Toolkit (NLTK) library. Stemming is further vital in decreasing the length of the article vector [18, 37]. There are two types of stemming methods: root and light stem. We used light stem by utilizing the Info Science Research Institute’s (ISRI) stemmer tool [38].

C. Lexicon Generation

In this section, lexicon generation will be shown.

Once the data is compiled to noise free and standardized form, we build the lexicon for each text class to be fed to the Lexicon and RS Theory (LRST) model. Consider \( n \) to be the number of posts, \( n \) represent the number of labels, and \( w \) is the number of words in the text. Let \( X \) represents the collection of articles where \( X = \{ A_i | A_i is an article, i \in Z \} \) and \( C \) is the collection of labels for each of the articles, \( C = \{ l_j | l_j is label of the article \} \). M. makes a partition on \( X \) such that \( A_i \in l_j \) for the same \( j \). When \( A_i \in l_j \), we refer to \( A_i \) by \( A_{ij} \).

Assume \( V \) to be the lexicon, which will be constructed for each article \( X \) in the label set \( C \) as shown in Equation (4):

\[
P_{ext} = \bigcup_{i} A_{i}^{ext} | A_{i}^{ext} \in l_{ext}
\]

\[
P_{non-ext} = \bigcup_{i} A_{i}^{non-ext} | A_{i}^{non-ext} \in l_{non-ext}
\] (4)

Equation (4) makes the partition such that every article must exactly belong to partition. In general, \( P_j \cup A_{ij} \). Where the number of articles belong to \( j \) [4]. We construct vector \( V \) for each class in \( l \) as shown in Equation (5) which will build the lexicon for each class.

\[
V_j = \{w | w \in P_j, 1 \leq j \leq n\}
\] (5)

VOLUME XX, 2017

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4.0/
be in the range of $1 \leq N_j \leq 2$. Thus, the obtained accuracy is very weak [40]. A popular normalization approach is shown in Equation (9), where the value of $N_j$ is between $0 \leq N_j \leq 1$:

$$N_j = \frac{N_j - \arg\min(N_j)}{\arg\max(N_j) - \arg\min(N_j)} \pm \alpha$$  

(9)

It should be noted that an $N_j$ value of 0 is problematic since, based on Equation (9), the accuracy approximation will also be equal to 0 [41]. To avoid these issues, the use of $\alpha$ optimization is proposed as shown in Equation (10):

$$N_j = \frac{N_j - \arg\min(N_j)}{\arg\max(N_j) - \arg\min(N_j)} \pm \alpha$$  

(10)

Where $\alpha$ is the optimization parameter with $0 \leq \alpha < 1$ and the $\pm$ will be based on outcomes of the numerator in Equation (11). If $N_j - \arg\min(N_j)$ is equal to 0, then the plus term of alpha is used; otherwise, the negative of the alpha term is used. By substituting Equation (9) into Equation (10), we get:

$$\text{Acc}(X, N_j) = \frac{B(X)_j \times (\frac{N_j - \arg\min(N_j)}{\arg\max(N_j) - \arg\min(N_j)} \pm \alpha)}{B(X)}$$  

(11)

V. RESULTS AND DISCUSSIONS

This section presents the detailed experimental outcomes and analysis of the statistical results. Precision, recall, F1-score, and accuracy are popular performance evaluation metrics in the context of classification problems. Table III presents the evaluation metrics where recall and precision are calculated using true negatives (TN), false negatives (FN), and false positives (FP). Further information about the use of these performance metrics can be found in study [42].

The proposed method is useful for selecting the main distinguishing subgroups (extreme and non-extreme). Table IV shows the value of the separate class available at the feature vector, using n-gram-based approaches.

<table>
<thead>
<tr>
<th>Metric name</th>
<th>Equation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy</td>
<td>$\frac{TN + TP}{TN + TP + FN + FP}$</td>
</tr>
</tbody>
</table>
Using n-gram representation, text segments consisting of a sequence of n words are formed. Then the similarity, i.e., the occurrence of these segments is measured in Table IV, considering their repetition, that is, with similarity and uniqueness. It is noted that when using trigrams, the total number of text segments with and without similarity is approximately the same. The frequency term (TF), as well as the Inverse Document Frequency-Term Frequency, are two statistical measures based on the presence of these properties (TF-IDF), which are subsequently processed by a ML classifier.

Tables V and VI present the performance of proposed ASLA approach while evaluating multiple ML methods using TF and TF-IDF, respectively. The RST method proposed in this work uses a lexicon vector. The proposed ALSA approach does not work with numerical data, and thus the n-gram representation is directly used, while the ML methods are based on numerical features (i.e., TF and TF-IDF). The performance of the RST technique is benchmarked with those of the ML techniques in Table VI.

Next, we considered which of the two feature extraction methods, i.e., TF and TF-IDF, led to higher performance with n-gram representation and ML algorithm. The performance results of Tables V and VI are consolidated in Table VI. Here, the first three columns correspond to the ML method used, whereas the first three rows belong to the ML method used for unigram, bigram, and trigram representation methods, respectively. Each of the three columns is further subdivided representing the use of TF and TF-IDF feature extraction with the specific n-gram representation. An entry of 1 for TF and TF-IDF corresponds to higher accuracy for the specific ML method, in Table VI, respectively. When the choice of feature extraction method does not influence the performance of the specific ML technique, the symbol ("-") appears in the corresponding columns. The last column indicates which of the two methods, TF and TF-IDF, resulted more frequently in higher performance in terms of accuracy. The last row simply sums up the number of times each of the two feature extraction methods resulted in higher performance under each of the three n-gram representations.

It can be noticed in Table VII, that the proposed feature extraction method has a substantial influence on the performance of NB. However, in the case of SVM and KNN, TF-IDF improved the performance. Thus, we can conclude that using TF-IDF features leads to improved performance of ASLA.

The next set of experiments in our study aim to select the best performing ML method to benchmark with our proposed method. By examining Tables V and VI, we observe that the greatest recital in terms of ML methods and the proposed approach is obtained when using the unigram representation. Moreover, the performance overview shown in Table VII concluded that TF-IDF leads to higher accuracy. Table VIII evaluates the performance of each ML method against the others, subject to the usage of unigram and TF-IDF. For selecting the best procedures, vote figures from zero to three were engaged. We started appraising the representations of the algorithms as of the row since the number that is absent from the brackets in the row indicates...
that the algorithm scored higher. The three grams as a bigram, trigram, and unigram, with TF-IDF, were used in the experiment, and the row was more expressive than the column, according to the outcomes.

**TABLE VIII**

<table>
<thead>
<tr>
<th>Alg.</th>
<th>NB</th>
<th>SVM</th>
<th>KNN</th>
<th>DT</th>
<th>RF</th>
<th>ANN</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>2(1)</td>
<td>1(2)</td>
<td>2(1)</td>
<td>1(2)</td>
<td>2(1)</td>
<td>2(1)</td>
</tr>
<tr>
<td>NB</td>
<td>0</td>
<td>2(1)</td>
<td>1(2)</td>
<td>2(1)</td>
<td>1(2)</td>
<td>2(1)</td>
</tr>
<tr>
<td>SVM</td>
<td>1(2)</td>
<td>0</td>
<td>2(1)</td>
<td>2(1)</td>
<td>3(0)</td>
<td>2(1)</td>
</tr>
<tr>
<td>KNN</td>
<td>2(1)</td>
<td>1(2)</td>
<td>0</td>
<td>2(1)</td>
<td>1(1)</td>
<td>2(1)</td>
</tr>
<tr>
<td>DT</td>
<td>1(2)</td>
<td>1(2)</td>
<td>1(2)</td>
<td>0</td>
<td>2(1)</td>
<td>2(1)</td>
</tr>
<tr>
<td>RF</td>
<td>1(2)</td>
<td>0(3)</td>
<td>1(1)</td>
<td>2(1)</td>
<td>0</td>
<td>2(1)</td>
</tr>
<tr>
<td>ANN</td>
<td>2(1)</td>
<td>1(2)</td>
<td>1(2)</td>
<td>2(1)</td>
<td>1(2)</td>
<td>0</td>
</tr>
</tbody>
</table>

In Table VIII, if the column and the row are similar, a value of zero is assigned (because there are no preferences between them). The addition and comparison of other algorithms are unaffected by a value of zero. Table IX shows the comparison between different algorithms in terms of received points. The algorithms were compared at the three-gram level, which was nearly the best level with ML. Each algorithm wins or loses based on the number it gets out of a possible 9, which is calculated by multiplying three algorithms by three grams.

**TABLE IX: WIN AND LOSE ALGORITHM**

<table>
<thead>
<tr>
<th>Algorithms</th>
<th>Win</th>
<th>Lose</th>
</tr>
</thead>
<tbody>
<tr>
<td>NB</td>
<td>8</td>
<td>7</td>
</tr>
<tr>
<td>SVM</td>
<td>10</td>
<td>5</td>
</tr>
<tr>
<td>KNN</td>
<td>8</td>
<td>6</td>
</tr>
<tr>
<td>DT</td>
<td>6</td>
<td>9</td>
</tr>
<tr>
<td>RF</td>
<td>6</td>
<td>8</td>
</tr>
<tr>
<td>ANN</td>
<td>7</td>
<td>8</td>
</tr>
</tbody>
</table>

Table IX shows that all algorithms have a value of 3 for both lost and win. In this case, all three algorithms were selected for benchmark purposes with the proposed algorithm. We applied the proposed process with two vectors to identify that functioned well [45]. The proposed process requires a value of \( \alpha \) parameter, as demonstrated in Table X. It displays how to find a value to develop the proposed technique, on the parameters of applying accuracy approximation. The choice of the alpha parameter is demonstrated in Table X, in which several training texts are applied; namely, 70% are chosen for training.

In every class, the total value of texts is regulated by Equation (10).

**TABLE X**

<table>
<thead>
<tr>
<th>Class</th>
<th>Normalization</th>
<th>( \alpha )</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Extremism</td>
<td>0</td>
<td>+</td>
<td>0.1</td>
</tr>
<tr>
<td>Non-extremism</td>
<td>1</td>
<td>+</td>
<td>1.1</td>
</tr>
</tbody>
</table>

The accuracy approximation gets the alpha parameter, as presented in Table X, therefore the proposed technique is combined with lexicon vectors. Table XI shows that the unigram, bigram, and trigram representations achieved an accuracy of 90.853%, 81.707%, and 70.121%, respectively. In lexicon-vector with the proposed method, the three grams achieved higher accuracy than ML.

**TABLE XI**

<table>
<thead>
<tr>
<th>Method</th>
<th>Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unigram</td>
<td>Bigram</td>
</tr>
<tr>
<td>NB</td>
<td>71.951</td>
</tr>
<tr>
<td>SVM</td>
<td>81.707</td>
</tr>
<tr>
<td>KNN</td>
<td>78.048</td>
</tr>
<tr>
<td>DT</td>
<td>75.609</td>
</tr>
<tr>
<td>RF</td>
<td>78.048</td>
</tr>
<tr>
<td>Proposed method</td>
<td>90.853</td>
</tr>
</tbody>
</table>

Figure 3 demonstrates the superiority of proposed ALSA approach when benchmarked together with SVM, KNN, and NB for each n-gram demonstration. It can be noticed that the proposed approach achieves higher accuracy than ML techniques.

**FIGURE 3.** Comparison between the proposed method and machine learning approaches using TF-IDF.

We have used a variety of evaluation metrics (as illustrated in Table II) on the unigram, bigram, and trigram representations. The simulation results are shown in Tables XII to XIV.
Experiment was conducted to benchmark the proposed method with the SVM, NB, and KNN algorithms. Tables XII - XIV show the precision, recall, and f-score of this experiment when using the different n-grams. From Figure 3, it can be observed that the proposed method has higher accuracy compared to SVM, NB, and KNN. As shown in Tables V and VI, the accuracy of the proposed method using trigram representation was lower when using unigram and bigram. To verify the accuracy of the proposed model, we benchmark with NB, SVM, and KNN of the corpus [46]. As illustrated in Table XI, the accuracy of the proposed method is better than the predicted accuracy of the benchmark methods. Generally, compared with the SVM, NB, and KNN, the proposed algorithm achieved improved classification error rates. By analyzing the results presented in the previous Tables, it is observed that the recommended method attained the optimal result (90.853%), which is better than the results obtained through other techniques. This remains unaffected by the type or size of the feature extraction method. Furthermore, in unigram, bigram, and trigram, the proposed method indicated better performance as compared to SVM, NB, and KNN based methods. (see Tables XII to Table XIV).

<table>
<thead>
<tr>
<th>Method</th>
<th>Class</th>
<th>P</th>
<th>R</th>
<th>F</th>
<th>Acc</th>
</tr>
</thead>
<tbody>
<tr>
<td>DT</td>
<td>Extremism</td>
<td>0.73</td>
<td>0.85</td>
<td>0.79</td>
<td>75.609</td>
</tr>
<tr>
<td></td>
<td>Non-extremism</td>
<td>0.79</td>
<td>0.65</td>
<td>0.71</td>
<td></td>
</tr>
<tr>
<td>RF</td>
<td>Extremism</td>
<td>0.72</td>
<td>0.95</td>
<td>0.82</td>
<td>78.048</td>
</tr>
<tr>
<td></td>
<td>Non-extremism</td>
<td>0.92</td>
<td>0.58</td>
<td>0.71</td>
<td></td>
</tr>
<tr>
<td>ANN</td>
<td>Extremism</td>
<td>0.78</td>
<td>0.78</td>
<td>0.78</td>
<td>76.829</td>
</tr>
<tr>
<td></td>
<td>Non-extremism</td>
<td>0.75</td>
<td>0.75</td>
<td>0.75</td>
<td></td>
</tr>
<tr>
<td>Proposed method</td>
<td>Extremism</td>
<td>0.86</td>
<td>0.96</td>
<td>0.91</td>
<td>90.853</td>
</tr>
<tr>
<td></td>
<td>Non-extremism</td>
<td>0.96</td>
<td>0.86</td>
<td>0.91</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table XII</th>
<th>Overview of performance metrics centered on bigram representation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>Class</td>
</tr>
<tr>
<td>NB</td>
<td>Extremism</td>
</tr>
<tr>
<td></td>
<td>Non-extremism</td>
</tr>
<tr>
<td>SVM</td>
<td>Extremism</td>
</tr>
<tr>
<td></td>
<td>Non-extremism</td>
</tr>
<tr>
<td>KNN</td>
<td>Extremism</td>
</tr>
<tr>
<td></td>
<td>Non-extremism</td>
</tr>
<tr>
<td>DT</td>
<td>Extremism</td>
</tr>
<tr>
<td></td>
<td>Non-extremism</td>
</tr>
<tr>
<td>RF</td>
<td>Extremism</td>
</tr>
<tr>
<td></td>
<td>Non-extremism</td>
</tr>
<tr>
<td>ANN</td>
<td>Extremism</td>
</tr>
<tr>
<td></td>
<td>Non-extremism</td>
</tr>
<tr>
<td>Proposed method</td>
<td>Extremism</td>
</tr>
<tr>
<td></td>
<td>Non-extremism</td>
</tr>
</tbody>
</table>

To summarize, the study used three grams for two feature extractions (TF and TF-IDF). ML methods performed well with unigram; however, most ML algorithms performed poorly with bigram and trigram; and most algorithms performed poorly with trigram. Because the relationship between the training and the test was zero, these algorithms did not perform well with bigram and trigram representations. This is known as the zero-relation limitation.

It should be noted that the main limitation of our proposed work is features selection should be performed manually which could consume considerable time but the advantage in relation to other work is the high accurate. It should also be noted that there is a the is a lack of emphasis on classifying extremism text into religion and political classes in which this paper is trying to address.

**VI. CONCLUSIONS AND FUTURE WORKS**

Sentiment analysis is a hot research topic with several challenges associated with natural language processing. It has a wide range of applications including marketing, news analytics, security, business information gathering, and many more. In this paper, the sentiment of twitter text in Arabic is presented in relation to detect the extremism in a primary dataset collected as part of this work. To reduce the amount of noise in the text, a variety of pre-processing techniques was applied. In addition, lexicon-based feature extraction is used to indicate the degree of negativity or positivity of each term in the lexicon. Sets of documents are used as input and output in the proposed algorithm, which utilizes the rough set theory. The proposed model is evaluated with machine learning based ALSA techniques as well as existing similar works, to investigate its effectiveness. Experimental results indicated that the
proposed technique outperformed the existing works when benchmarked with NB, SVM, and KNN, producing the accuracies of 90.853%, 81.707%, and 71.951% for unigram, bigram, and trigram representations, respectively. Three grams are utilized for two feature extractions which are TF and TF-IDF. ML methods performed well with unigram; however, most ML algorithms performed poorly with bigram and trigram.

Future works can use swarm optimization such as cuckoo search, genetic algorithm, etc., to select the optimal words for building a lexicon. Another direction for future work will involve the enhancement of accuracy approximation based on simulated annealing.

REFERENCES


This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2023.3278272
Amjad Abbass Ahmed received the B.Sc. degree in computer science from the University of Baghdad and the M.Sc. degree in computer science from Binary University, Kuala Lumpur, Malaysia, in 2012. From July 2013 to July 2022, he was a Lecturer in the Imam Al-Kadhum College, Baghdad, Iraq. He is currently pursuing the Ph.D. degree in University of Kebangsaan Malaysia, with a focus on Artificial Intelligence.

Mohammad Kamrul Hasan (Senior Member, IEEE) received the Doctor of Philosophy (Ph.D.) degree in electrical and communication engineering from the Faculty of Engineering, International Islamic University, Malaysia, in 2016. He is currently working with the Center for Cyber Security, Faculty of Information Science and Technology, Universiti Kebangsaan Malaysia (UKM), as a Senior Lecturer.

Mustafa Musa Jaber is a PhD holder from technical University of Malaysia and he received a postdoctoral from University Tun Hussein Onn Malaysia, his interest in telemedicine, machine learning, and human factor.

Sumaia Mohammed AL-Ghuribi received the BSc with honors in Computer Science from Taiz University, Yemen in 2008. She received the M.S. degree in Computer Science from King Abdulaziz University, Jeddah, Saudi Arabia in 2014, and the PhD degree from the Universiti Kebangsaan Malaysia (UKM), in 2021. Her range of research interests includes natural language processing, web mining, sentiment analysis, and recommender systems.

Dhafar Hamed Abd is a PhD holder from university of technology Iraq, his interest in NLP, machine learning, deep learning, optimization, and sentiment analysis.
**Wasiq Khan** is a Senior Academic in Artificial Intelligence & Data Sciences within the Department of Computer Science at Liverpool John Moores University, UK. He is also serving as a Visiting Professor of Artificial Intelligence (University of Anbar, Iraq). Wasiq received his B.Sc. in Mathematics, Physics, and Geography following an M.Sc. in Computer Science from Pakistan. At Bradford University UK, He received an M.Sc. in Artificial Intelligence for Board Games following by a Ph.D. in Speech Analysis & Intelligent Reasoning and a Post Graduate Certificate in Teaching & Learning in Higher Education (PGCHEP). Wasiq has been publishing the research outcomes in high impact Journals, peer reviewed conferences, News blogs & Media, Scientific festivals and public events. He is an active reviewer for top ranked Journals (e.g., Transactions) and government funding bodies.

**Ahmed Tareq Sadiq** is a professor at the University of Technology in Baghdad, Iraq. Prof. Sadiq received a B.Sc., M.Sc. & Ph. D. degree in Computer Science from the University of Technology, Computer Science Department, Iraq, 1993, 1996 and 2000 respectively. He is Professor in A.I. since 2014. His research interests in artificial intelligence, data security, patterns recognition and data mining. He has more than 100 published papers in several scientific journals and conferences.

**Abir Hussain** is a visiting professor of Machine Learning at Liverpool John Moores University, UK. She completed her PhD study at The University of Manchester (UMIST), UK in 2000 with a thesis title Polynomial Neural Networks for Image and Signal Processing. She has published numerous referred research papers in conferences and Journal in the research areas of Neural Networks, Signal Prediction, Telecommunication Fraud Detection and Image Compression. She has worked with higher order and recurrent neural networks and their applications to e-health and medical image compression techniques. She has developed with her research students several recurrent neural network architectures. She is a PhD supervisor and an external examiner for research degrees including PhD and MPhil. She is one of the initiators and chairs of the Development in e-Systems Engineering (DeSE) series.