Mustansiriyah Journal of Pure and Applied Sciences, Print ISSN: 2957-9910. Online ISSN: 2957-9929, Vol. 3, No. 4, 2025, Pages 173-189
DOI: 10.47831/mjpas.v3i4.292

MJPAS

MUSTANSIRIYAH JOURNAL OF PURE AND APPLIED SCIENCES

Journal homepage:
https://mjpas.uomustansiriyah.edu.iq/index.php/mjpas

RESEARCH ARTICLE – COMPUTER SCIENCE

Extremism Detection Using Hybrid Deep Learning Models

Hind Ali Suleiman[1*], Zuhair Hussein Ali [2]

[1,2] Department of Computer Science, College of Education, Mustansiriyah University, Iraq

[*] Corresponding author E-mail: hindali@uomustansiriyah.edu.iq

Article Info.

Article history:

Received
16 August 2024

Accepted
3 October 2024

Publishing
30 September 2025

Abstract

Extremist groups often utilize social media to disseminate their ideas and beliefs in order to compile more new members and thereby help them spread violent content and extremist ideologies that threaten social cohesion. The goal of detecting extremism on social media is to identify and prevent the spread of these ideas through the use of artificial intelligence technologies that will help us detect extremist texts in social media. This paper proposes a methodological approach based on the use of deep learning algorithms for the effective cessation of extremist texts spread. The dataset we used is called ISIS radical annotated tweets; it consists of 24078 tweets from 174 accounts related to the extremist organization known as the Islamic State. Initial preparation and cleaning of the dataset are applied to ensure the accuracy of the dataset. Feature extraction techniques using (Term Frequency (TF), Inverse Document Frequency (IDF), Word2vec) are applied to extract most important features. To identify extremist texts, this paper chooses two state-of-the-art deep learning algorithms Long Short-Term Memory (LSTM) and combined Convolutional Neural Network CNN- LSTM.LSTMs have a lengthier term of memory and thus work well with text content, whereas CNNs are designed to extract features from the data. The final CNN-LSTM model combines both algorithms' strengths. In this study we used Deep learning techniques to classify extremist texts, two different deep learning approaches were used (LSTM and Hybrid CNN-LSTM). The CNN-LSTM combination produced the highest accuracy reached to (98.20%).

The official journal published by the College of Education at Mustansiriya University

Keywords: Extremism, Long Short-Term Memory (LISTM), Convolutional Neural Networks (CNN).

1. Introduction

Compared to earlier times, more people can now access the Internet, and more people are using social networking sites. The Internet is full of programs, like Facebook, Instagram, Twitter, and many more, where a vast number of people may share their thoughts and feelings about a wide range of topics and objects. And disseminating their ideas through these platforms; however, despite the benefits of these applications, there is a drawback some extremist individuals and groups attempt to use these platforms propagate hate speech and sectarian ideas throughout

society. The terrorist group ISIS is one example of an extremist group that does this [1]. Several extreme groups are attempting to use social media to disseminate hate speech and their ideas. Put differently, these organizations are recruiting new members and disseminating harmful beliefs and plans via social media. To stop these groups from propagating their dangerous beliefs and inciting violence and hatred on social media platforms, which could lead to war and conflicts in peaceful communities, it is crucial to identify their members[2]. The most worrying thing about this very important issue is how challenging it is to manually monitor and review every daily publication because users publish hundreds of thousands of publications on social media platforms in a matter of minutes [3]. The aim of this paper is to identify and prevent the spread of extremist texts and the dissemination of their dangerous beliefs that may cause the spread of violence, hate speech and sectarian ideas throughout society, which may lead to war and conflicts in peaceful societies. For this reason, we develop techniques that automatically detect content Hate and extremism without human intervention.

The identification of extremism on social networking sites is a complex problem that calls for interdisciplinary solutions that bring together sociology, ethics, technology, and law. In order to develop useful instruments that can reduce the dissemination of dangerous ideas while upholding individual rights, these issues must be addressed [4]. key Aspects of this Problem[5] :

- Content Diversity: It is difficult to create a detection algorithm that works for all types of extremist content because it might appear as text, photos, videos, or memes.
- Changing Language: Extremist groups frequently employ slang, symbols, and coded language that changes over time, making identification more difficult.
- Volume of Data: Real-time monitoring and analysis are severely hampered by the massive amount of user-generated content on social networking sites.
- Contextual Understanding: Automated systems may find it difficult to understand subtleties and sarcasm; context is essential in deciding if a statement is radical or innocuous.
- Privacy Issues: Ethical conundrums arise when attempting to strike a balance between the necessity to protect user privacy and freedom of speech and effective detection.
- False Positives/Negatives: High percentages of both (extremist content not discovered) and false positives (innocuous content labeled as extremist) can erode confidence in detection systems.

2. Related Work

This section compiles the most important research on the subject of extremism detection in Arabic literature.

Nuha Al Badi et al. in 2018 [6], conducted the first study on the detection of Arabic extremism. They also produced the first Arabic dictionary containing terms that are hateful toward religion, which they made publicly available in an effort to promote further research on this subject. Additionally, they created the first Arabic dataset that was used to identify hate speech. Using various classification models as the basis for their methodology, it was discovered that the Arahate-PMI performed best in terms of F1, recall, and accuracy, while the RNN based on the GRU produced the best results, averaging 0.84 on some measures. Arahata-BNS was the best

performance in terms of accuracy than AUCROC. While the n-gram-based models, logistic regression, and the support vector machine (SVM) performed similarly, they outperformed the lexicon-based models, especially in terms of accuracy. The drawback of this study Look at religious hate speech only this might miss other types. Might not catch new ways people express hate over time. Manual labelling for hate speech can be subjective, causing errors.

Nuha Albadi et. al. in 2019 [7], presented a study that was a continuation paper from a conference published in ASONAM 2018(Albadi et al. 2018 [6]). They employed four methods for detecting religious hate speech which include a lexicon-based approach, an Ngram-based approach, GRU+ word embedding, and four manual embedding features of GRU+. They concluded They concluded GRU-based RNNs with word embedding pre-trained models outperform other lexicon-based and n-gram classifiers. Their training of the GRU model was on some features such as user, content, and temporal. As well as including pre-trained words for tweets and user descriptions, resulting in speech recall performance (0.84).

Ahmed I. A. AbdElaal et. al. in 2020 [1], introduced a new architecture with an advanced algorithm that finds Pro-ISIS Twitter accounts on its own. The system involves two subsystems: the crawling system and the query system. The two kernel subsystems are smart detectors. Which have characteristics such linguistic and conduct characteristics. Supervised machine learning techniques were utilized in the development of the Smart Detector kernel for the crawl and query subsystems. The results were as follows, linear SVM algorithm with TF-IDF embedding got the best accuracy of 89% for ISIS content detector. also showed that the ISIS computation detector provides 94% pest accuracy based on the f1 score using the Skip-gram linear-modulated SVM algorithm. The drawbacks of this study assuming that certain words and actions are only used by radical groups could be wrong because these groups might change their ways to avoid getting caught. Finding accounts automatically could lead to mistakes and invade privacy.

Mohammed A. Al Ghamdi et. al. in 2020 [8], introduced a system that uses datasets that were tweeted to train a classifier to detect suspicious activity using supervised machine learning algorithms. During the testing phase, the system evaluated the unlabeled twitter data to assess if the content was suspicious or not. They use six supervised machine learning algorithms to test the system: decision tree (DT), k-nearest neighbors (KNN), linear discrimination algorithm (LDA), SVM, artificial neural networks (ANN), and long short-term memory networks. ANN has the slowest execution speed while SVM outperforms all other classifiers in predicting correct results, with an average accuracy of 86.72%.the drawbacks of this study the tool's performance could change with different data. It doesn't say how well the tool works with new tweets.

Saja Aldara et al. In 2021 [9] gathered a dataset to address the classification techniques that can be applied to identify radicalization. 89,816 Arabic-language tweets published between 2011 and 2021 made up the data collection. Experts evaluated the tweets according to predetermined standards to determine whether or not they were radical. an investigational study of the data that was conducted to comprehend the characteristics of the data collection. They then employed methods for classification, including RF, BERT, naive Bayes polynomial, logistic regression, and support vector machines. Of the standard machine learning models, the SVM TF-IDF feature achieved the highest accuracy (0.9729). BERT, however, fared better with 0.9749 accuracy than the traditional models. The drawbacks of this study experts might disagree on what's extremist,

and it misses other types of extremist content. One method is complex and needs lots of resources.

Mohammad Fraiwan in 2022 [2], The study was based on classifying tweets as terrorist-related, generally religious, or unrelated using artificial intelligence (AI) and ML classification algorithms. The obtained results achieved an accuracy of K-nearest neighbors (KNN), Bernoulli Naive Bayes (NN), and SVM [one-against-all (OAA) and all-against-all (AAA) algorithms. At SVM-OAA, it has a highly rated F1 score of 83%. The drawbacks of this study Limited to analyzing Twitter data from ISIS members this might not capture all nuances of terror-related content. Doesn't account for potential evasion techniques used by extremists.

A summary and a brief comparison between the studies reviewed above in terms of data sets, preprocessing methods, text representation, feature extraction methods, classification models, as well as the highest accuracy obtained by the model. As illustrated in Table 1.

TABLE 1. Comparison of Arabic Extremism Detection Researches

Researchers and year	Datasets	Dialect/MSA	Pre-processing	Text representation	Machine Learning model	Accuracy
Nuha Albadi et al (2018) [6]	6000 Arabic tweets in 2017 and 600 tweets in 2018	Dialect and MSA	Clean data Remove stopwords Tokenize Stemming Normalizing	chi-square, PMI, and BNS web_CBOW Wikipedia_CBOW	Approaches based on lexicons, N-grams, and deep learning	GRU-based RNN performs best, with 0.79 accuracy and 0.84 AUROC.
Nuha Albadi et al (2019) [7]	6000 Arabic tweets	Dialect and MSA	Clean data Remove stopwords Tokenize Stemming Normalizing	Chi-square, PMI, and BNS web_CBOW Wikipedia_CBOW	AraHate-PMI AraHate-Chi AraHateBNS logistic regression SVM GRU + word embeddings GRU + word embeddings + handcrafted features	Training a GRU and pre-trained word embeddings performs in terms of recall (0.84)
Ahmed I. A. Abd-Elaal et al (2020) [1]	21,000 tweets and three datasets in Kaggle "How ISIS Uses Twitter", "Religious Texts Used By ISIS",	Dialect and MSA	Remove URL links and mentions, Discarding non-alpha letters removal, Normalization, Stop words removal, Tashkeel removal, Prefix/suffix removal	TF-IDF and Skip-gram "Mazajak"	BNN, DT C, K-NN, SVM, LR and RF Classifiers	best accuracy 94% by linear SVM with Skip-gram word embedding

	"Tweets Targeting ISIS"					
Mohammed A. AlGhamdi et al (2020) [8]	1555 tweets	MSA	Clean data, Stemming, and Lemmatization	Bag-of-words (BoW) model and word embedding	DT , k-NN , LDA, SVM , ANN, and Long short-term memory networks (LSMN)	SVM was the best performance with 86.72% mean accuracy.
Saja Aldera et al (2021) [9]	89,816 tweets published between 2011 and 2021	Dialect and MSA	Lemmatization, Stop-words removal, Tokenization	TF-IDF and Word2Vec	LR, MNB, SVM, RF, and BERT	SVM using TF-IDF achieved accuracy (0.9729) , while BERT model outperformed SVM, achieve 0.9749.
Mohammad Fraiwan (2022) [2]	24,078 tweets	Dialect and MSA	Filtering the duplicate tweets, Tokenizing, Removing diacritic marks, Normalization, Lemmatizing	Word embedding	KNN, BNB and SVM linear Kernel OAO and OAA classifiers	achieved F1 score of 83\% in SVM-OAA

3. Methodology for Extremism Detection

The architecture for the suggested extremism detection module is described in this section, which has a four-part architecture, as shown in Fig. 1.

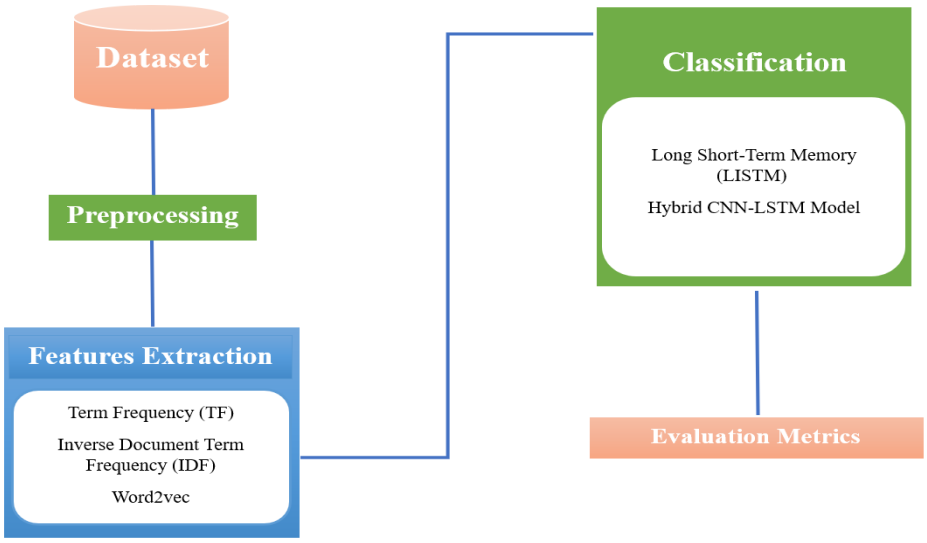

Fig.1. Proposed architecture.

3.1 Data collection

As a result of the increase in the number of Arab users on social media platforms, it has become important to monitor the content of social media sites to detect extremist texts and offensive speech that may affect our society and incite violence and terrorism in this paper we used data set called (Annotated ISIS radical tweets) to detect extremism. this dataset is available online was by (Mohammad Fraiwan, 2022) [2].

It consists of a set of 24078 tweets from 174 accounts in related to the extremist organization known as the Islamic State. The annotation determines if it is radical and associated to terrorism (marked in the excel file with the letter T) or religious but unrelated to terrorism (marked with the letter F).

3.2 Data preprocessing

Before heading to analysis data, it must be preprocessed to remove noise. This step is very important and must be done because of the insufficient, inconsistent of some databases and consist noise in it. It is necessary to clean and get ready for analysis the gathered data. This entails taking away unwanted characters, normalizing texts, removing stop words, and other common preprocessing techniques to enhance data quality [10].

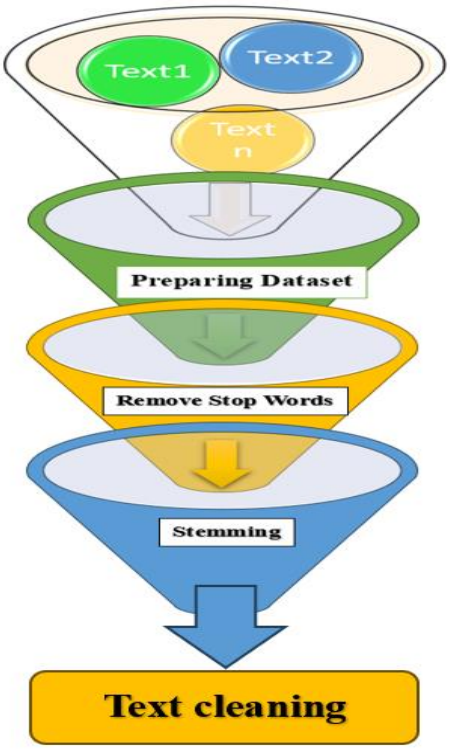

Fig.2. Phase of Data preprocessing.

3.2.1 Data Preparing:

The data cleaning step consists of many important steps such as:

- URL Removal.
- Eliminating English words.
- Eliminating Punctuation.
- Eliminating Usernames.
- Text tokenization.
- Eliminating single-letter words.
- Eliminating special characters [11].

3.2.2 Remove Stop Words:

The goal of this phase is to get rid of any unnecessary words that don't help make a distinction between categories. It entails getting rid of frequently used terms that don't help with class differentiation, like prepositions, articles, single letters, auxiliary words, and formatting tags. The elimination of these terms has no negative effects on categorization accuracy because they are broad and not particular to any text category. As a matter of fact, their inclusion may reduce categorization accuracy [12]. There are many strategies utilized for indicating such stop words list. Now, various Arabic stop word list is generally utilized to preprocess the dataset .

3.2.3 Stemming:

The linguistic process of stemming reduces words to their base, or stem form, regardless of whether this stem matches the word's morphological basis. Assuring that all related word variations are combined under a single stem makes this an essential text preparation step, especially prior to document indexing. By treating diverse grammatical forms or variations of a verb as the same word, it improves text mining algorithms' accuracy and efficiency. For instance, stemming ensures words like " ,"المسافرون","المسافران and " مسافر" are recognized as the same term by the system [13].

3.3 Feature Extraction

The text must then be changed to a format that the Deep learning algorithm can comprehend after the data has undergone preprocessing. We call this method feature extraction. Several methods are available for feature extraction, including:

3.3.1 Term Frequency (TF): In document d, the TF shows the number of times a certain word, t, appears. It follows that a term gets more relevant the more times it appears in the text. Since the term ordering is irrelevant, we can use a vector to represent the text in the bag of term models. For every distinct term in the document, there is an entry with the TF as the value. We must calculate the following [14] : (Total number of terms in the document) / (number of times a term appears in a document) is the value of TF(t).

3.3.2 Inverse Document Frequency (IDF): It evaluates the word's relevance first and foremost. The primary objective of the search is to locate the pertinent records that satisfy the demand. Furthermore, because TF views every phrase as equally relevant, it is not able to determine a term's weight in the article solely by looking at its term frequencies [14]. It weights the observed numbers through the IDF. This model is an extension of the TF model, but in TF only words are used [15] .To find a phrases document frequency, count the number of documents that include it. First, we must compute the following: IDF(t) is equal to loge (total documents / documents containing the term t).

3.3.3 Word2vec: The majority of NLP models make extensive use of Word2Vec. It transforms the word into vectors. A two-layer net called Word2vec uses words to parse text. The text corpus serves as the input, and feature vectors which represent the words in the corpus serve as the output. Word2vec transforms text into a clear computational format for deep neural networks, despite not being a deep neural network itself. Gathering vectors of the same words collectively in vector space is Word2vec's goal and advantage. In other words, it looks for mathematical parallels. Word2vec generates vectors based on numerical representations of word constituents, including contextual information about individual words. It achieves this without assistance from humans [16].

3.4 Classification Models

The efficiency of the various classification models that are available varies depending on the problem domain.

3.4.1 Convolutional Neural Networks (CNN)

ConvNets, another name for CNN, are a subset of artificial neural networks (ANNs). They have a deep feed-forward architecture and, in comparison to other networks with front-end (FC) layers, an amazing capacity for generalization. Specifically, they are able to identify objects more accurately and become familiar with highly abstracted object properties, especially spatial data. A finite number of A deep CNN model is composed of processing layers., which can learn different input data features (such images) at several levels of abstraction. Higher abstraction is used by the deeper layers to learn and extract low level features, whereas lower abstraction is used by the initiatory levels to learn and extract high level features. Convolutional, pooling, and fully connected layers are among the layers that make up CNN [17]. Fig.3. depicts CNN architectural layout.

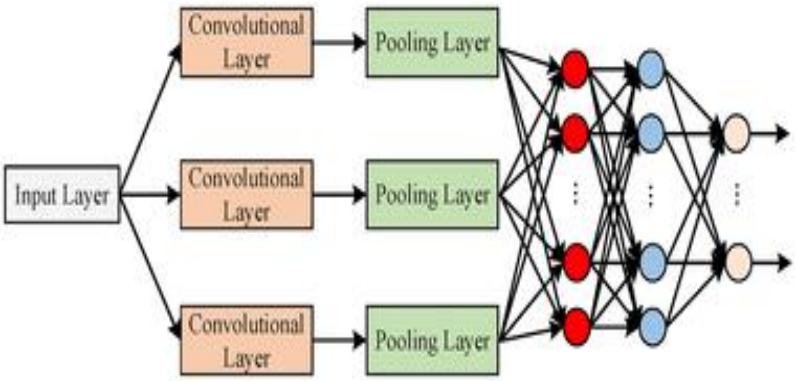

Fig.3. CNN architecture [18].

3.4.2 Long Short-Term Memory (LSTM)

LSTM is considered as a development of RNN and was presented via Schmidhuber and Hochreiter to address the issues of the disadvantages of RNN via adding more interactions for each one of the modules (or cells). LSTM is considered a distinctive RNN type to learn long-term dependencies and remember information for prolonged periods as a default. As can be seen in Fig.4. the basic idea of an LSTM model is the control of cell states through the employment of three gates: the input, forget, and output gates [19].

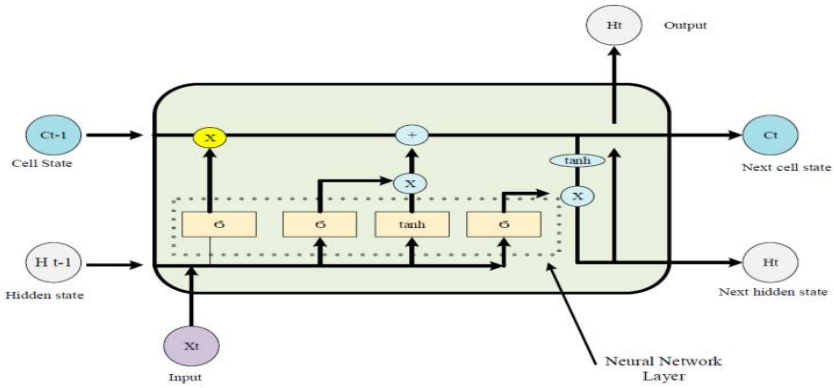

Fig.4. Diagram of the LSTM network's block [20].

4. Model Construction

The Proposed Hybrid CNN-LSTM Model The proposed system depends on Two algorithms of deep learning A CNN type and an LSTM model were integrated. One CNN layer and one LSTM layer make up this model. 80% of the dataset are using for training while testing uses the remaining 20%. The process of recognizing extremism is addressed as a binary classification problem, where class "0" represents non-extremism and class "1" represents extremism. The suggested combination hybrid CNN-LSTM for extremism identification is shown in Fig.5. Layers (Embedding layer, Convolution layer, Max pooling layer, LSTM layer, fully connected layer) make up this model.

- **Embedding Layer:** First, pre-trained and prepared weights will be employed as vectors to represent the words within the embedding matrix using the vectors that were made clear during the feature extraction stage. As a result, in addition to the length of words in the dataset, the size of the embedding matrix will vary depending on the length and kind of vector employed.

- **Convolution Layer:** CNN layer uses a fixed-size filter to extract the n-grams (n consecutive words) of features by scanning a series of vectors used in the input layer. Since text analysis involves one-dimensional feature matrices, convolutional layer one dimension, or Conv1D, is employed. It functions essentially like a user-selectable sliding window in terms of movement. Each filter used the ReLU activation function to find several features in a sentence and represent them in the feature map.

- **Max pooling Layer:** Following the feed of the embedding layer to the network, the pooling layer uses the embedding layer features to down sample them, thereby reducing the size of the feature set and identifying the optimal feature relationships for classification. Max pooling, a nonlinear down sampling technique, is used to help choose the best-performing terms or elements. where the down sampling is applied by computing the maximum activation of predefined subregions inside the features set.

- **LSTM Layer:** In order to give memory to the output of the preceding layer and make it an input for it to determine the long-term associations between feature sequences, The model is extended with one layer of LSTM, each with a distinct number of units.

- **Fully connected Layer:** The input vector from the preceding layer is transformed into a single output in the final layer, depending on how many classes need to be classified— four or six for multiclass classification, and two for binary classification. The activation function in this layer is the sigmoid function as shown in equation (1) with binary classification and the SoftMax function as shown in equation (2) with multi classification [21]. When compiling the model two types of loss functions were utilized to calculate the error to measure the distinction of the actual distributions. Binary loss entropy function with the binary classification model and categorical loss entropy function with the multi classification model. Finally, in the back propagation phase, the errors are calculated between the target and predicted outputs, and it is checked whether these errors are acceptable or not. Then the Adam optimization algorithm is used to update the weight values.

$$f(x) = \frac{1}{1 + \exp^{-x}} \qquad (1)$$

Where x is the input to calculate sigmoid. If the logit is tiny, that means the logistic neuron output is so near to 0. Otherwise, the logit is very large and means the logistic neuron output is nearest to 1.

$$f(x) = \frac{\exp(x_i)}{\Sigma_j \exp(x_i)} \qquad (2)$$

where all the x values are the elements of the input vector and can take any real value. exponential function is applied to each element of the input vector. It ensures that all the output values of the function will sum to 1 and each be in the range (0, 1).

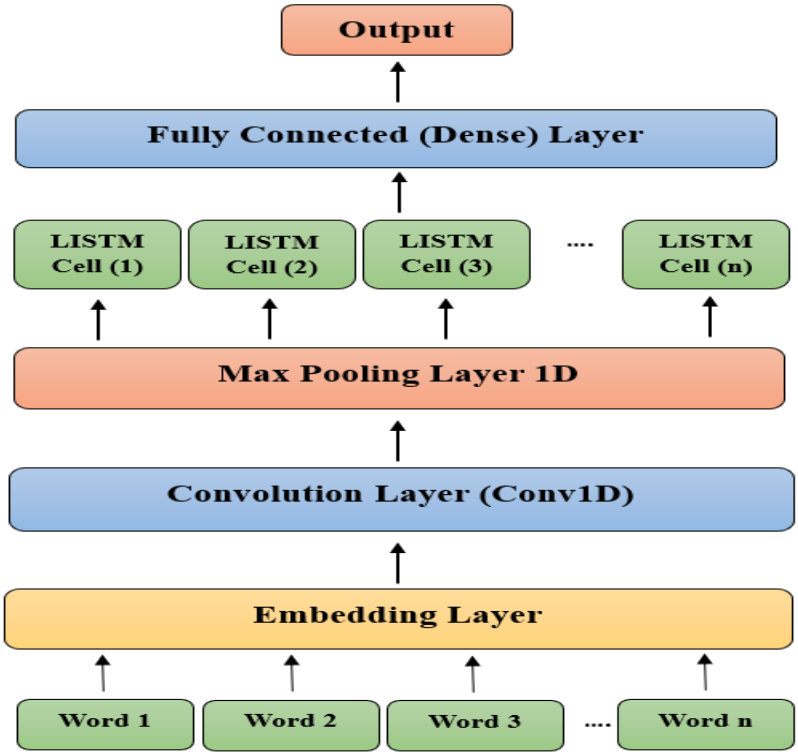

Fig.5. The proposed model of CNN-LSTM.

5. Performance Evaluation

Analyzing a Deep Learning model's performance is an important step in creating a competent model. Evaluation metrics, also known as performance metrics, are a collection of measurements used to rate a model's quality. These assessment metrics make it easier to comprehend how well the prediction models perform in respect to the given dataset. Additionally, the assessment measures are used to improve the model's performance through parameter adjustments.

5.1 Accuracy:

Accuracy is the proportion of correct predictions the model produces. The calculation involves dividing the entire number of forecasts by the amount of the predictions that are true positives (TP) and true negatives (TN). At this point, it is important to never ignore the number of false positive (FP) and false negative (FN) predictions the model produces. These are the cases where the model prediction and the actual category deviate [22].

$$\text{Accuracy} = \frac{\text{Tp+TN}}{\text{Tp+FN+TN+Fp}} \quad (3)$$

5.2 Precision and Recall:

The two metrics that were used to evaluate the algorithms' efficacy were recall and precision. The algorithm's performance in correctly recognizing documents was evaluated using the Recall metric, and the percentage of correctly retrieved documents was measured using the Precision meter [23].

$$\text{Precision} = \frac{\text{TP}}{\text{TP+FP}} \quad (4)$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP+FN}} \quad (5)$$

5.3 F1-score:

The F-Measure provides a way to integrate recall and precision into a single metric that incorporates both characteristics. Since a model may have great precision but low memory, or vice versa, it is impossible to fully understand the evaluation of a model by looking at precision and recall separately. It is possible to combine both metrics into a single score using the F-measure [24].

$$\text{F1} - \text{score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{precison+ Recall}} \quad (6)$$

The receiver operating characteristic curve (ROC) is created by plotting the true positive rate with the false positive rate. Zero and one define the bounds of the area under the curve (AUC), which typically exceeds 0.5.

6. Experimental Results and Discussion

These are the findings from our models' performance assessments; The experimental results, precision, recall, accuracy, F1-score, and AUC values obtained for (models) are shown in Table 2. When the models' findings were analyzed, the CNN-LSTM hybrid model performed better in terms of classification accuracy than the LSTM model. With an accuracy of 98.20, recall of 95.64, precision of 97.53, and F1 measure of 96.92, the CNN-LSTM achieved the top results. While LISTM model accuracy was 92.79, recall was 92.10, precision was 94.33, and F1 measure was 92.62.

TABLE 2. Results of proposed system

MODEL	ACCURACY %	RECALL %	PRECISION %	F1-SCORE %
LISTM	92.79	92.10	94.33	92.62
CNN-LISTM	98.20	95.64	97.53	96.92

Fig (6) and (7) below show the Receiver Operating Characteristic (ROC), with the true positive rate (TPR) on the Y-axis and the false positive rate (FPR) on the X-axis. The term false positive rate (FPR) describes the number of false positives that were mistakenly categorized as positives, while the term true positive rate (TPR) describes the number of positives that were correctly classified as positives. The trade-off between TPR and FPR at different categorization criteria is shown by the ROC curve.

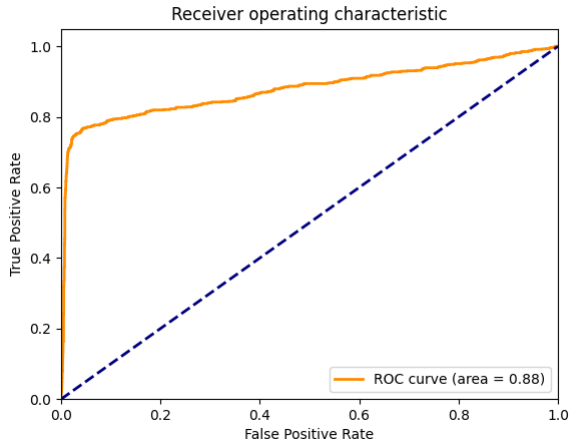

Fig.6. ROC curve for the model LISTM.

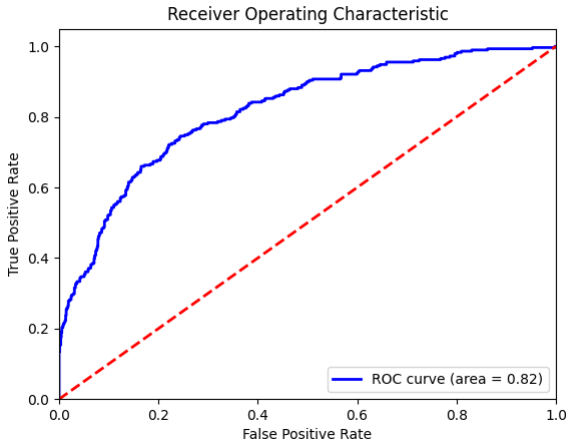

Fig.7. ROC curve for the model CNN-

We also made a comparison of our proposed models with the relevant works as shown in Table 3. Which displays the results of our models (LSTM, CNN-LSTM) and other classifiers Our proposed models outperform the other classifiers with an accuracy of 98.20 for CNN-LSTM, 92.79 for LSTM. Our models also achieved higher scores in terms of macro-average F1-score, recall, and precision compared to the other classifiers.

TABLE 3. Comparing the Proposed Models Results with other Related Work

RELATED WORK	CLASSIFIER	ACCURACY	F1- SCORE	RECALL	PRECISION
OUR PROPOSED MODELS	LISTM	92.79	92.62	92.10	94.33
	CNN-LISTM	**98.20**	96.92	95.64	97.53
NUHA ALBADI ET AL (2018) [6]	AraHate-PMI	0.71	0.69	0.72	0.66
	AraHate-Chi	0.69	0.65	0.66	0.65
	AraHate-BNS	0.70	0.64	0.62	0.67
	logistic regression	0.74	0.71	0.70	0.71
	SVM	0.75	0.72	0.72	0.72

	GRU-based RNN	**0.79**	0.77	0.78	0.76
MOHAMMED A. ALGHAMDI ET AL (2020) [8]	DT KNN LDA SVM ANN LSTM	**SVM was the best performance with 86.72% mean accuracy.**			
SAJA ALDERA ET AL (2021) [9]	LR	0.9723	0.9724		
	MNB	0.9046	0.9032		
	SVM	**0.9729**	0.9730		
	RF	0.9671	0.9653		
	BERT	**0.9749**	0.9749		
AHMED I. A. ABD-ELAAL ET AL (2020) [1]	Bernoulli NB Decision Tree Classifier K Neighbors Classifier Linear SVC Logistic Regression Random Forest Classifier	**best accuracy 94% by linear SVM with Skip-gram word embedding**			
MOHAMMAD FRAIWAN (2022) [2]	KNN	**achieved F1 score of 83% in SVM-OAA**	69.6	76.6	64.2
	BNB		77.3	80.1	74.7
	SVM-OAA		83.2	84.3	82.2
	SVM-OAO		82.5	83.5	80.7

7. Conclusions

The social media extensive availability and user-friendliness have made it simple for radical individuals, groups, and organizations to disseminate false information, draw sizable audiences, and enlist new members. We have examined thousands of tweets endorsing and advancing ISIS in this research. We have applied text processing (URL Removal, Remove English words, Eliminating Punctuation, Eliminating Usernames, etc.) and AI Using deep learning algorithms, extremist tweets are categorized. Two distinct deep learning techniques—LSTM and CNN-LSTM—were employed in this study. The CNN-LSTM combination produced the best results, with the highest accuracy recorded at 98.20%. This indicates that the algorithm is highly accurate in the classification process.

8. References

[1] A. I. A. Abd-Elaal, A. Z. Badr, and H. M. K. Mahdi, "Detecting Violent Radical Accounts on Twitter," 2020. [Online]. Available: www.ijacsa.thesai.org

[2] M. Fraiwan, "Identification of markers and artificial intelligence-based classification of radical Twitter data," *Applied Computing and Informatics*, 2022, doi: 10.1108/ACI-12-2021-0326.

[3] A. Rajendran *et al.*, "Detecting Extremism on Twitter during U.S. Capitol Riot Using Deep Learning Techniques," *IEEE Access*, vol. 10, pp. 133052–133077, 2022, doi: 10.1109/ACCESS.2022.3227962.

[4] C. Winter, P. Neumann, A. Meleagrou-Hitchens, M. Ranstorp, L. Vidino, and J. Fürst, "Online extremism: Research trends in internet activism, radicalization, and counter-strategies," *Int J Conf Violence*, vol. 14, no. 2, pp. 1–20, 2020, doi: 10.4119/ijcv-3809.

[5] H. Collison-Randall, R. Spaaij, E. J. Hayday, and J. Pippard, "Media framing of far-right extremism and online radicalization in esport and gaming," *Humanit Soc Sci Commun*, vol. 11, no. 1, p. 1195, Sep. 2024, doi: 10.1057/s41599-024-03680-4.

[6] Ulrik. Brandes, Chandan. Reddy, and Andrea. Tagarelli, *ASONAM 2018 : proceedings of the 2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining : Barcelona, Spain, 28-31 August, 2018*. IEEE, 2018.

[7] N. Albadi, M. Kurdi, and S. Mishra, "Investigating the effect of combining GRU neural networks with handcrafted features for religious hatred detection on Arabic Twitter space," *Soc Netw Anal Min*, vol. 9, no. 1, Dec. 2019, doi: 10.1007/s13278-019-0587-5.

[8] M. A. AlGhamdi and M. A. Khan, "Intelligent Analysis of Arabic Tweets for Detection of Suspicious Messages," *Arab J Sci Eng*, vol. 45, no. 8, pp. 6021–6032, Aug. 2020, doi: 10.1007/s13369-020-04447-0.

[9] S. Aldera, A. Emam, M. Al-Qurishi, M. Alrubaian, and A. Alothaim, "Exploratory Data Analysis and Classification of a New Arabic Online Extremism Dataset," *IEEE Access*, vol. 9, pp. 161613–161626, 2021, doi: 10.1109/ACCESS.2021.3132651.

[10] I. T. A. Noor Al-Deen Alaa, "A Comprehensive Study of Usage-Based Web Mining," *Mustansiriyah Journal of Pure and Applied Sciences*, vol. Vol.2, no. Vol. 2 No. 1. second volume, Jan. 2024.

[11] A. Alharbi, M. Kalkatawi, and M. Taileb, "Arabic Sentiment Analysis Using Deep Learning and Ensemble Methods," *Arab J Sci Eng*, vol. 46, no. 9, pp. 8913–8923, Sep. 2021, doi: 10.1007/s13369-021-05475-0.

[12] I. Budiman, D. T. Nugrahadi, M. R. Faisal, M. R. Faisal, and M. Rusli, "A Study on Effect of Generated Features From Word2Vec Vectors For Text Classification." [Online]. Available: https://www.researchgate.net/publication/348404518

[13] S. A. Yousif, V. W. Samawi, and I. Elkabani, *Arabic Text Classification: The Effect of the AWN Relations Weighting Scheme*. 2017. [Online]. Available: https://www.researchgate.net/publication/318787881

[14] A. O. Salau and S. Jain, "Feature Extraction: A Survey of the Types, Techniques, Applications," in *2019 International Conference on Signal Processing and Communication, ICSC 2019*, Institute of Electrical and Electronics Engineers Inc., Mar. 2019, pp. 158–164. doi: 10.1109/ICSC45622.2019.8938371.

[15] M. A. H. Wadud, M. F. Mridha, and M. M. Rahman, "Word Embedding Methods for Word Representation in Deep Learning for Natural Language Processing," *Iraqi Journal of Science*, vol. 63, no. 3, pp. 1349–1361, 2022, doi: 10.24996/ijs.2022.63.3.37.

[16] Z. Sultana Ritu, N. Nowshin, M. Mahadi Hasan Nahid, and S. Ismail, "Performance Analysis of Different Word Embedding Models on Bangla Language," in *2018 International Conference on Bangla Speech and Language Processing, ICBSLP 2018*, Institute of Electrical and Electronics Engineers Inc., Nov. 2018. doi: 10.1109/ICBSLP.2018.8554681.

[17] A. Ghosh, A. Sufian, F. Sultana, A. Chakrabarti, and D. De, "Fundamental concepts of convolutional neural network," in *Intelligent Systems Reference Library*, vol. 172, Springer, 2019, pp. 519–567. doi: 10.1007/978-3-030-32644-9_36.

[18] Z. Tang, T. Zhang, Y. Du, and J. Su, "Individual identification method of little sample radiation source based on SGDCGAN+DCNN," *IET Communications*, vol. 17, no. 3, pp. 253–264, Feb. 2023, doi: 10.1049/cmu2.12508.

[19] K. E. ArunKumar, D. V. Kalaga, C. Mohan Sai Kumar, M. Kawaji, and T. M. Brenza, "Comparative analysis of Gated Recurrent Units (GRU), long Short-Term memory (LSTM) cells, autoregressive Integrated moving average (ARIMA), seasonal autoregressive Integrated moving average (SARIMA) for forecasting COVID-19 trends," *Alexandria Engineering Journal*, vol. 61, no. 10, pp. 7585–7603, Oct. 2022, doi: 10.1016/j.aej.2022.01.011.

[20] I. S. Samanta *et al.*, "A Comprehensive Review of Deep-Learning Applications to Power Quality Analysis," Jun. 01, 2023, *MDPI*. doi: 10.3390/en16114406.

[21] C. Enyinna Nwankpa, W. Ijomah, A. Gachagan, and S. Marshall, “Activation Functions: Comparison of Trends in Practice and Research for Deep Learning.”

[22] D. M. W. Powers and Ailab, “EVALUATION: FROM PRECISION, RECALL AND F-MEASURE TO ROC, INFORMEDNESS, MARKEDNESS & CORRELATION.”

[23] S. Kadhem, Z. Ali, and A. Suhad Malallah Zuhair Hussein Ali, “Multi-Document Summarization using Fuzzy Logic and Firefly Algorithm Multi-Document Summarization using Fuzzy Logic and Firefly Algorithm ارزميه ذبابة النار تلخيص النصوص المتعددة باستخدام المنطق وخوارزمية الضبابي2018 ”,. [Online]. Available: https://www.researchgate.net/publication/330006097

[24] B. T.k., C. S. R. Annavarapu, and A. Bablani, “Machine learning algorithms for social media analysis: A survey,” May 01, 2021, *Elsevier Ireland Ltd*. doi: 10.1016/j.cosrev.2021.100395.