WJCMS

Wasit Journal of Computer and Mathematics Science

Journal Homepage: https://wjcm.uowasit.edu.iq/index.php/WJCM
e-ISSN: 2788-5879 p-ISSN: 2788-5887

Cyberbullying Detection in Arabic Text Using Different Deep Learning Approaches

Sura Sabah Rasheed[1], Sura Mazin Ali[2], Humera Shaziya[3],*,Ahmed T. Sadiq[4], Saif Ali Abd Alradha Alsaidi[5]

[1]Scientific Research Commission, Ministry of Higher Education and Scientific Research, Iraq
[2]College of Political Science, Mustansiriyah University, Iraq
[3] Department of Informatics, Nizam College, Osmania University, India.
[4] Computer Science College , University of Technology, Iraq
[5]Department of software, college of computer science and information technology , Wasit University , Iraq.

*Corresponding Author: Dr.Humera Shaziya

DOI: https://doi.org/10.31185/wjcms.390
Received 10 June 2025; Accepted 22 June 2025; Available online 30 June 2025

ABSTRACT: The rise of social media has enabled the rapid spread of user-generated content across various domains, including advertising, entertainment, politics, and economics. However, this growth has also facilitated the increase of harmful behaviors, notably cyberbullying. Addressing this issue requires advanced emotional and sentiment analysis techniques. In this study, the ArCyC (Arabic Cyberbullying Corpus) were integrated with Twitter data to develop robust models for cyberbullying detection. Several deep learning models have been suggested which are: Deep Neural Net. (DNN), Convolutional Neural Nets. (CNN), Recurrent Neural Net. (RNN), Hybrid CNN+RNN, BERT & AraBERT. Text and Emoji have been experimented in the dataset. Models' performance was evaluated based on accuracy. Experiments using the ArCyC dataset demonstrated that both textual and symbolic elements contributed significantly to classification accuracy. In contrast, analysis with the ArCyC dataset revealed that textual features had a more dominant influence due to the limited use of emojis. The results underscore the effectiveness of deep learning approaches in detecting cyberbullying within Arabic social media content. AraBERT for text has obtained the highest accuracy equal to 95%, similarly LSTM obtained the same accuracy for both text and emoji's.

Keywords: cyberbullying , detection , deep learning ,text minig.

1. INTRODUCTION

The rapid evolution of social media platforms has become an indispensable tool for communication. Nevertheless, even with all the benefits they offer, new issues have arisen, most importantly cyberbullying. Cyberbullying is the abuse or harassment of another person via social media and the internet. This form of bullying is not limited by time or location; it pervades people's life continuously. For making the internet a safer environment for everyone, we will investigate in this paper the phenomena of cyberbullying, its psychological and social consequences, and methods to minimize it. We will look at utilizing emojis in cyberbullying, a more recent phenomenon, in which emojis are sent harmful or unpleasant messages, and cyberbullying can be utilized for sarcasm, mocking, or maybe threats. Nowadays, emojis are often utilized in combination with text in correspondence; sometimes, bullying occurs just with emoticons. Other times it comes from text alone, or from both at once. Those that engage in this kind of behavior often go through negative emotions like despair and worry, which aggravates loneliness and insecurity. Cyberbullying detection was addressed using algorithms examining social media posts, textual data, or messaging using deep learning (DL) and machine learning (ML) methods. Neural networks (NNs), sentiment analysis, and natural language processing (NLP) among other methods help identify harmful or abusive language patterns. Two examples of ML models that classify content as either bullying or not are support vector machines (SVM) and decision trees (DTs.). More improved detection is made possible by DL techniques such transformers and recurrent neural networks (RNNs) by understanding context and subtle signs of violence.

*Corresponding author:humera.shaziya@gmail.com

ML and DL routinely use SVM, RNNs, DT, and CNNs to detect cyberbullying using emojis. SVM distinguishes bullying from non-bullying signals thereby helping to classify material. Conversely, RNNs fit for spotting minor kinds of cyberbullying in Emoji use since they handle sequential data and are good in capturing context and nuanced expressions. These techniques improve detection accuracy by use of Emojis in communication's emotional tone and intent interpretation.

There are numerous studies on detecting cyber bullying using Machine Learning. The focus of the paper will be on detecting cyberbullying through Emojis using Deep Learning algorithms, whether in text, Emoji form, or a combination of both. These algorithms are capable of analyzing patterns in both textual and visual data, making it possible to identify harmful behaviors conveyed through Emojis, text, or both together. The goal is to improve detection methods for cyberbullying across various communication forms.

2. RELATED WORKS

The social web's quick development has made it possible to carry out NLP research. Various methods, including ML, DL, and NLP, were utilized by researchers for identifying the polarity regarding sentiment expressed on different social media platforms, given that the social web is an active study area. Every day, people of different races, languages, and cultures leave comments on a wide range of topics, celebrities, photographs, and other content on social media platforms like Twitter, Facebook, and weblogs. Bullying, aggression, and hate speech threads may result from the variety of users on social media. This type of online activity is difficult to handle because of the complexity of foreign languages.

Researchers employ terms like cyberbullying [1–3], racism [6–7], hate speech [8], offensive language [9], and profanity [4,5] to describe aggression. In [10], the author examined whether non-cyber-aggressors seek more psychological consultations than cyber-aggressors and compared them to cyber-aggressors. In order to identify aggression in a Twitter dataset, the authors in [11] employed network, text, and user-based features. Additionally, they proposed that whereas bullies actively participate in online forums and attempt to spread more negativity, victims of bullying tend to write fewer posts and take part in fewer discussions. Another research [12] performed a subjectivity analysis using sentiment lexicon and the Lexicon-based method to detect hate speech. Authors of [13] shown that selecting useful text elements, like nouns, verbs, adjectives, a number of emoticons, and others, can enhance the effectiveness regarding offensive language detection.

The authors of [14] conducted tests to find "bully" traces on the social network. They identified "bully" traces on Twitter dataset by using semantic and syntactic features for textual data depending on emojis. Additionally, they used other social media datasets, like Formspring and YouTube, to evaluate their model. To detect sarcasm in tweets, [15] developed a pragmatic model with three features: negative emoticons, user tagging, and positive emoticons. To shed light on the challenge of sarcasm detection, they contrasted their model's performance with that of human sarcasm detection. In order to identify irony and sarcasm in an English-language Twitter dataset, the author in [16] conducted tests. Emoji and TF-IDF vector methods for converting text into vectors were among the features. The author covered the connection between sarcasm, irony, and cyberbullying. Twitter tweets about cyberbullying in English were gathered for another study [17]. After that, in order to predict the tweets, they developed an auto-detection model regarding cyberbullying detection depending on sentiment score, text, and readability. Additionally, the authors employed a dictionary of curse words to determine the text's emotional score by counting the instances of negative emotions in a tweet. Ref. [6] utilized the English data set for identifying aggression detection with the use of Bert model.

In order to identify objects in images, Krizhevsky et al. [18] developed a DCNN method. Better features are extracted from object detection using this method. To identify abusive comments, store web sites that spread these kinds of messages, stop these web sites, and enhance safety discussions in online platforms, Eswari and Anand [19] presented the LSTM without and with integration of word GloVe embeddings. Kaggle dataset is used in this study to identify the different types of harmful comments. Text mining method for classifying text messages with the use of term-based method was suggested by Li et al. [20]. A number of problems, including polysemy and synonymy, are found in the current methods. Pattern-based processes have outperformed term-based methods for many years. Large datasets cannot be mined using such methods, which is still a significant problem in text mining. To identify hate speech from user comments found online, Nobata et al. [21] created an ML-based method. User comments that have been annotated for abusive language make up a dynamic corpus. For classifying sentence-level tasks, Kim [22] suggested that the CNN model train the vectors. To provide extreme results on a variety of datasets, the author combines multiple DL models. The integrated model was suggested by Ibrohim et al. [23] utilizing the word embedding (word2vec) feature. In order to detect hate speech as well as abusive language on Twitter in Indonesian, the suggested method was merged with part of speech and emoji. RF, SVM, LR, and DT have been the classification algorithms employed in this investigation. Using a publicly accessible corpus of 16,000 tweets, Hovy and Waseem [24] provided a method for identifying hate speech. Character ngrams are integrated with extra-linguistic features to enhance the detection regarding hate speech and identify accurate hate speech. The alert-based method (ABA), which detects hate speech on SNS, was suggested by Vigna et al. [25]. Based on the text, ABA concentrated on identifying instances of religious, caste, and personal abuse. For classifying hate speech words and recognize speech, ABA was paired with LSTM and SVM. Both

classification methods accurately identify hate speech. The unique DL method that automatically identifies irrelevant language was suggested by Yenala et al. [26].

Finding irrelevant language is made easier by the novel method. Spelling mistakes as well as linguistic variances are indicative of irrelevant language. Convolutional Bi-Directional LSTM (C-BiLSTM), a hybrid of BLSTM and CNN, is the name of the suggested method. CNN is utilized for extracting the important features found in the provided dataset, whereas BLSTM is utilized for filtering out the irrelevant language. Therefore, when put to comparison with the current models, C-BiLSTM achieved higher accuracy. To identify bullying messages online, Islam et al. [27] created a useful method. NLP and ML methods were combined with this technique. The accuracy of this BoW and TF-IDF combination was higher than that of current ML algorithms. Using the Bag-of-Phonetic-Codes model, Venkatesan and Shekhar [28] presented a unique method for detecting cyberbullying. Words that are misspelled or misused should be eliminated based on pronunciation. The textual features were extracted using the BoW model in the suggested method. In order to improve the suggested system's performance, Soundex algorithm concentrated on developing phonetic coding. Tests demonstrate that the innovative method achieved accurate cyberbullying detection. A novel model for determining the text's accurate meaning was presented by Sharma et al. [29]. Additionally, a novel model is employed to decrease the spreading of harmful messages via the internet. Better results are obtained when ML is combined with NLP features. A novel model created by Wadhwani et al. [30] addresses a number of problems, including irrelevant content detection and mismatched bullying. The primary goal of this article was to identify harmful comments that cause problems for users of SNS. Based on metrics like hate, toxic, threat, serious harmful, etc., the suggested DNN model analyzes the type of messages and looks for patterns in input messages. Bhandary and Wu [31] established a classification system depending on whether a video is a hate or normal video. The online crawler was used to gather the dataset videos from online sources. DEA_RNN model was suggested by Murshed et al. [32] to identify cyberbullying messages on online SNS. On analyze the cyberbullying text, the method was applied on a dataset of 10,000 tweets. Comparing the suggested DEA_RNN method to other models, including RNN, Bi-LSTM, MND, SVM, and RF, the results were the best. In order to identify cyberbullying messages in real-time apps like as Twitter, Malempati and Bai [33] introduced the ETMA method. ETMA combined with the CNN model and TF-IDF to classify the text into non-bullying and bullying notes. When it comes to classifying cyberbullying messages, ETMA provides reliable results.

3. PROPOSED WORK

The system's main task is to analyze whether the comments in Arabic are negative or positive. This section explains the system's practical components. Gathering the dataset and employing Arabic text for sentiment analysis are the two main components regarding the system; it also entails getting a significant amount of Arabic comment data. Compiling the dataset is a crucial step. To guarantee the efficacy and accuracy of the system, the dataset must encompass a broad range of themes and emotions. The second part of the system is a sentiment analysis of Arabic texts, which aims to identify the emotions conveyed in a particular text. Here, the system prioritizes comments in the Arabic language. This section therefore recommends using a range of technologies, such as DL algorithms and NLP methods. The Arabic Sentiment Analysis (ASA) objective was accomplished by the suggested system with the application of DL methods. As shown in Figure 1, the procedure comprised reading the dataset, getting it ready, and breaking the work up into more manageable, smaller chunks. Emoji were encoded in the early part of the process using the Spacy method. Transformer models such as AraBERT have been employed in the second section. LSTM and CNN were among the DL techniques employed in the third section. The two embedding methods, FastText and Spacy, were used to generate each of such models.

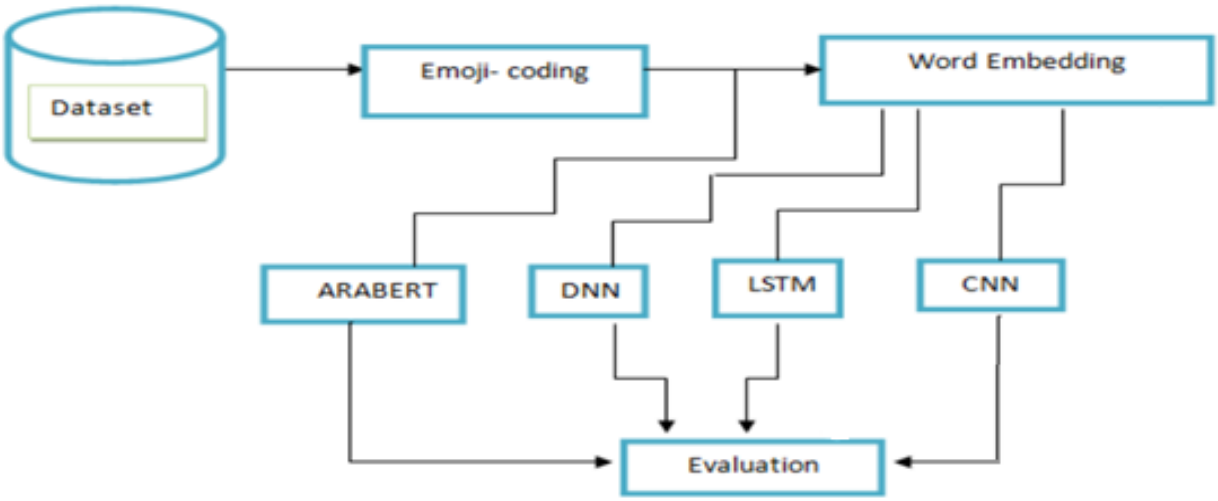


FIGURE 2. - Block Schematic for the Proposed System

3.1 DATASET

The current study made use of a Fully Annotated Arabic Cyberbullying Corpus (ArCyC), a dataset created specifically to investigate and identify cyberbullying in Arabic-language social media texts. For developers and researchers working on Arabic text-related NLP tasks, the corpus is a valuable resource, especially when it comes to offensive language, online harassment, and cyberbullying. This dataset contains over 10,000 labeled instances of Arabic text. These instances are sourced from various Arabic social media platforms and are annotated with labels indicating whether the text contains cyberbullying content or not. This volume allows researchers to conduct extensive training and testing of machine learning models aimed at detecting and addressing cyberbullying in Arabic-language content.

The texts in the corpus are annotated for cyberbullying-related content, meaning that each instance of text is labeled as either bullying or non-bullying. The annotations typically involve identifying harmful, abusive, or threatening behavior in the text, such as insults, threats, or discriminatory remarks. It includes various cyberbullying forms, like hate speech, verbal abuse, and harassment. The corpus may also identify targets of bullying (such as specific groups or individuals) and distinguish between different types of abusive behaviors.

This dataset is especially important due to the fact that much of the research and available datasets have historically focused on English, which makes it harder to build effective detection tools for languages like Arabic. Having a specific dataset like ArCyC aids in the development of more efficient systems for content moderation and online safety, especially in light of the growing usage of AI and automated moderation on social media platforms. ArCyC is an invaluable resource for advancing research in Arabic NLP and combating online abuse in the Arabic-speaking world. It helps bridge the gap between the need for cyberbullying detection and the scarcity of suitable data in non-English languages, contributing to the development of more inclusive and culturally sensitive AI systems.

3.2 EMOJI CODING AND PREPROCESSING

Quantifying and defining emoji symbols is the first step in developing a successful program. The weight, frequency and repetition of emojis in evaluations were determined by the researchers using digital coding techniques. A total of 284 emoji symbols have been found using the Spacy techniques, and FastText was trained on a large number of texts and comments for classifying and investigate how emojis affect reviews. Emoji analysis of Arabic tweets showed certain cases of conflicting sentiment expression. The researchers gathered Arabic tweets containing emojis in order to further examine the relation between sentiment and emojis. They discovered that tweets with more emojis performed better. The sentiment of tweets with a one emoji was after that manually evaluated, and both emojis and texts had consistent sentiment expressions. Depending on usage, the frequency of symbols changed, creating four situations: incomprehensible language, ad-like tweets, emotionless tweets, and opposite sentiment expression.

Emojis were extracted from every review and their frequency was determined by the process algorithm. The weight of each emoji was after that calculated using this data, taking into account how frequently it appeared in both negative and positive reviews. To show the emojis and their weights and classifications, a data frame was made. A vocabulary dictionary for emojis and their weights was also created. The average contexts associated with each emoji were determined by loading the Arabic version of the FastText model that had already been trained. Lastly, the context-based averages were saved as expected quantities for every review. This similar technique was used to generate the other model (Spacy), although the latter was more developed and trained in a variety of Arabic dialects and emojis. Figure 2 illustrates an example for emoji weights.

12	34	23	20
24	22	25	28
13	14	16	11
22	18	21	15

FIGURE 1. - Represented Emoji as Weights

Some of the researches have taken symbols as a basic element for evaluation, due to the fact that they were studied as either a digital code or a code that is specific to every one of the emojis, and the majority of them have been converted to a text that expresses the state, for example (😊) a smiling face or (😔) a sad face or other expressions, but we have utilized a more precise method, as we encoded them (1-,1) to express them directly if they were negative or positive.

Emojis are highly important due to their presence in reviews impacts the results significantly, as well as the likelihood to rely on them for making a decision in the cases of not knowing the clear text's direction in comments that are incomprehensible and not clear.

Pre-processing: dataset preparation is the process of making the raw data ready for the analysis or modeling. Data is cleaned, and after that it is transformed then persisted for the purpose of improving ease of use and usability. Pre-

processing represents a highly important part of data science journey due to the fact that data quality has an impact on the efficiency of any subsequent modeling or analyses.

The steps of pre-processing include

Removing all the punctuation and diacritics from text. Those elements are distracting and might have nothing to do with the way that the feelings are understood. Which is why, there is a high importance in removing them before performing the mood analysis.

Making Arabic text more consistent through the changing of some letters. Normalization indicates the changing of some of the letters in Arabic text into their normal form for the purpose of removing mistakes and making the word embedding smaller. For example, [جمييييل] can be changed to [جميل].

Taking out any characters appearing for a number of times in text. Characters appearing several times can result in mistakes when performing mood analysis.

Removing all of the Arabic words that aren't negation characters: Words contributing little or nothing to the sentence's meaning, they can be removed without resulting in a change to the text's overall tone.

3.3 WORD EMBEDDING

3.3.1 FASTTEXT EMBEDDING:

The main objective of the 2016 [29] release of FastText, an effective NLP library, by Facebook AI researchers was to enhance the continuous skip-gram model [30] by incorporating subword information. Their new method includes learning word representations as the vector sum of acquired character's n-grams. Because of the difficulty of noisy data and the notion that subword information could be useful for rare words. As a potential text representation technique, fastText word embedding was incorporated into the study's experimental framework. Words have been represented in this framework as sets of character n-grams, where n may be any value from 3 to 6. Thus, capturing a variety of word characteristics, from suffixes to extensive roots. This advancement, which was made possible by combining DL with DQN, highlighted how crucial internal word information is. The word (where), for example, arises when n equals three, giving the word the following expression: (we, wh, here, her, where, re). The entire sequence is also given, as is clear. The training corpus had N-grams, and each one was given a vector representation. Ultimately, the word was represented as the sum of its vector representations in n-grams.

3.3.2 SPACY EMBEDDING:

A neural network (NN) model as well as an open-source library for fundamental NN activities are represented by Spacy [31]. Tokenization, lemmatization, named entity recognition, similarity detection, part-of-speech tagging, tokenization, rule-based matching, other linguistic aspects are some of these tasks. Additionally, gradient as well as loss functions are used in the stochastic model's training phase. Additionally, the trained model does well in transfer learning scenarios. Spacy is an open-source software library that offers a number of useful utilities for handling textual data. Support for more than 73 languages, cutting-edge speed, a production-ready training system, linguistically driven tokenization, simple extension through custom components and attributes, and solid accuracy that has undergone extensive evaluation are some of these technologies.

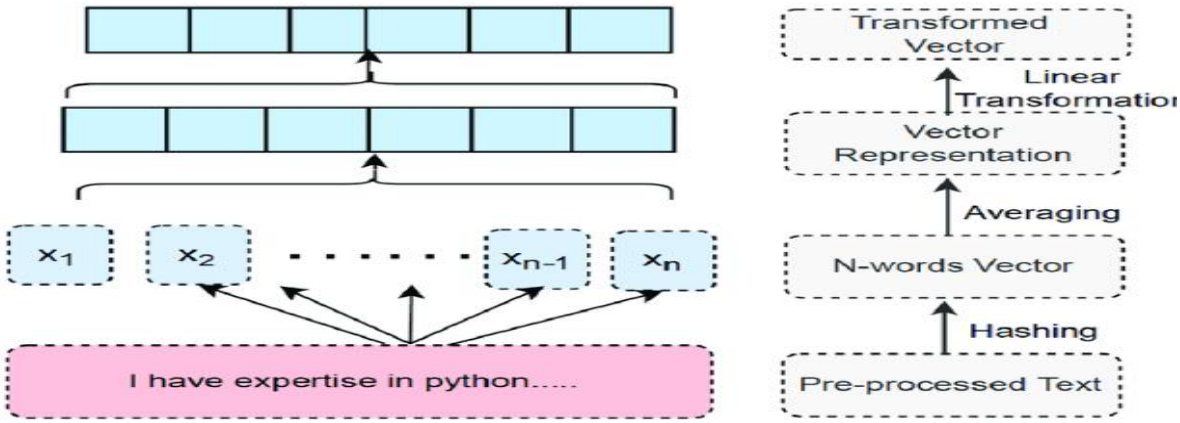
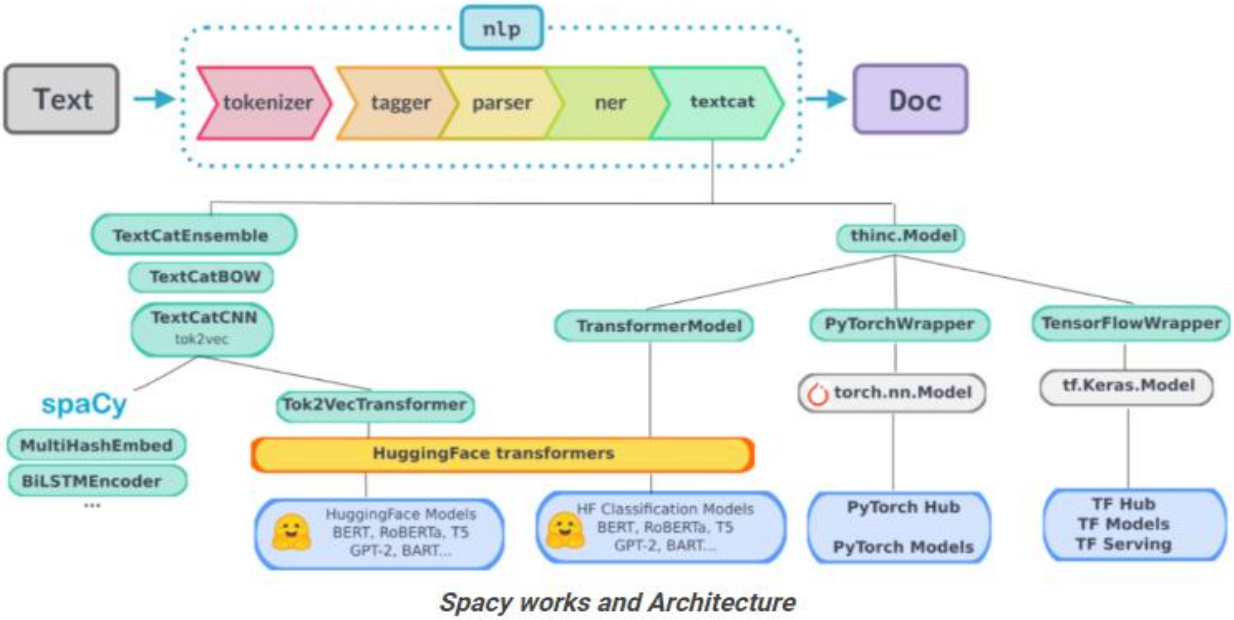


FIGURE 3. - Architecture FastText Embedding [29]

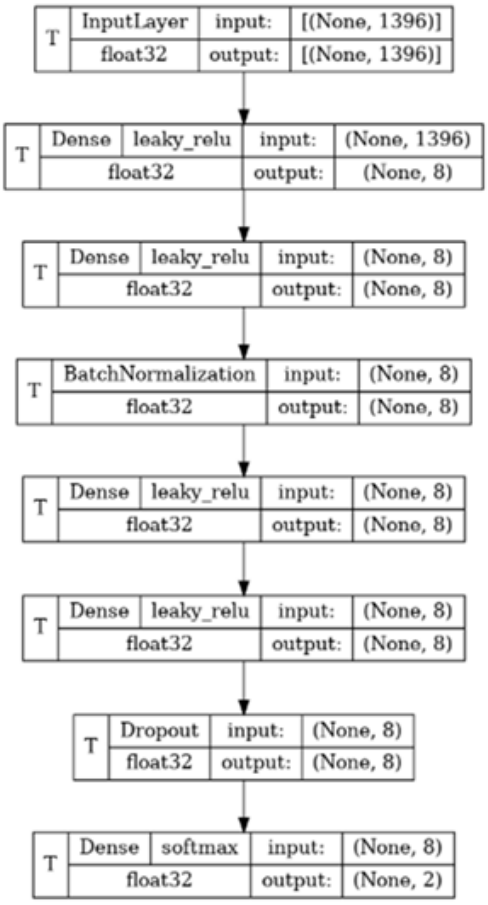


Spacy works and Architecture

FIGURE 4. - Spacy Architecture [31]

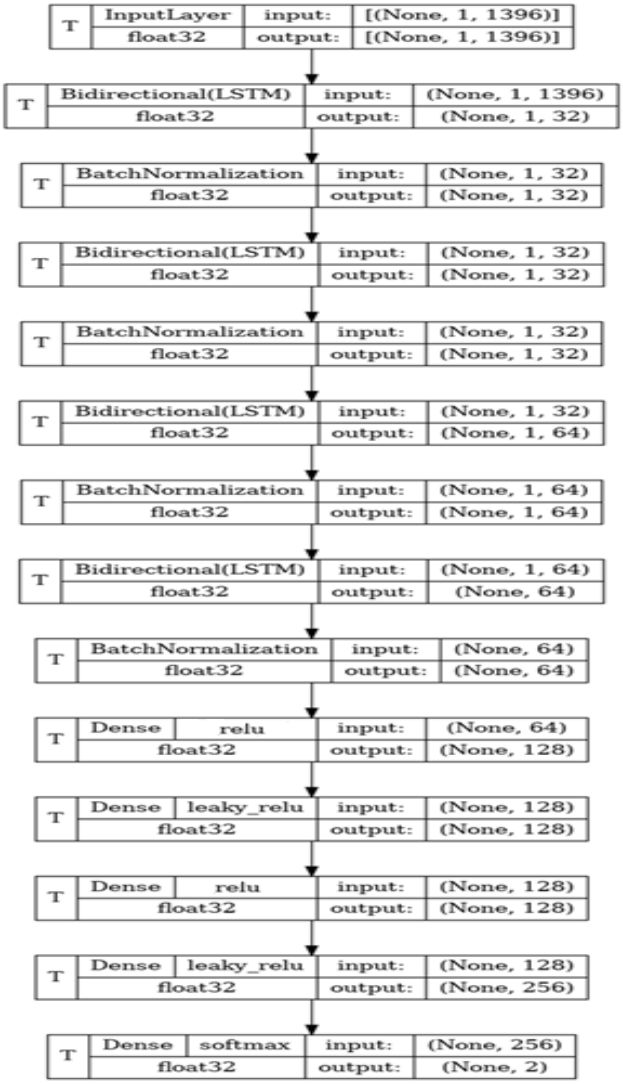
3.4 DEEP NEURAL NETWORKS (DNNS)

DNNs have been widely used in data-driven modeling. Layers having mathematical relations between edges and nodes make up a DNN. As the data is being trained, such associations are updated via back propagation. Depending on the training data, the output variables are after that predicted using the altered relationships as equations. As a result, one of DNNs' main advantages is their ability to communicate the system's links in spite of its nonlinearity and complexity.



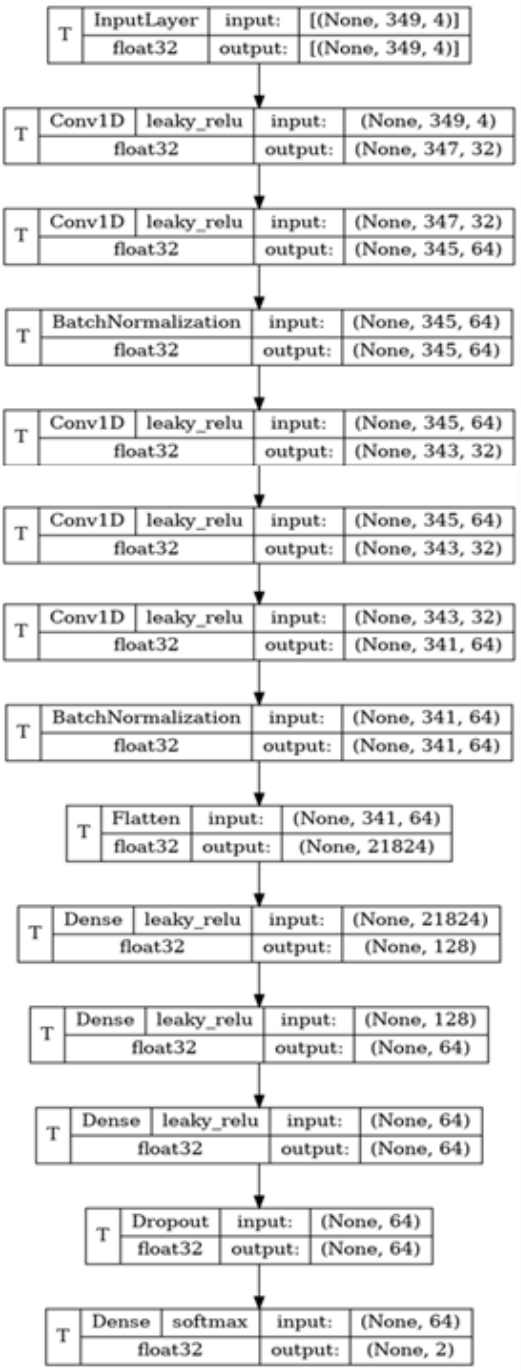
3.5 LSTM

Networks are designed to retain information in longer sequences and are especially adept at handling long-term dependencies. Three gates—the forget, output, and input gates—control the internal state of memory in LSTM networks. These gates regulate which memory contents must be added, deleted, or preserved. The gates use element-wise multiplication and a sigmoid layer to limit the amount of data which could pass through. Depending on the training process, the memory cell eventually picks up the essential information. Through maintaining an error constant across time while imposing no bias against recent and changing observations, LSTM aims to address some issues, such as vanishing gradients. [32]



3.6 CNN

Originally created for computer vision applications, CNNs are feed-forward neural networks [33, 34]. CNNs have made significant strides in NLP technologies. They contain a layer with convolutional filters that are applied locally. CNNs use convolution rather than universal matrix multiplication, in contrast to traditional NNs. CNNs' decreased weight count and network complexity have made them one of the DL algorithms with the quickest execution times. CNNs also have the benefit of requiring less preprocessing. This feature has enabled its usage in a number of fields, such as NLP, voice as well as handwriting recognition, and image processing.



3.7 ARABERT

A language model based on the BERT architecture is developed for processing Arabic. It is a crucial tool for Arabic NLP, enabling scholars and experts to train models for a variety of NLP tasks, such as sentiment analysis, named entity recognition, and text classification. A large collection of Arabic text, comprising a range of online documents, such as news articles, blogs, and social media posts, served as the initial training set for AraBERT model. Similar to BERT's pre-training phase, AraBERT's pre-training involved teaching it to infer words without adding sentences. AraBERT could produce Arabic text representations of exceptional quality by using this paradigm since it is able to comprehend the contextual relationships between words as well as their meanings [35]. AraBERT was tested in numerous NLP projects with positive results. For example, AraBERT outperformed previous pre-trained language models and attained state-of-the-art performance in Arabic sentiment analysis [34, 35].

Experimental Results

In the presented work, the challenges of data collection in Arabic were encountered, along with selecting the suitable one. It was ensured that the data included sufficient emojis to evaluate the studies. Various deep-learning models were then implemented, with modifications made by adjusting the number of layers and selecting an appropriate design. This process aimed to demonstrate the most effective outcomes through experimentation, evaluation, and comparison between

different models. The models varied regarding word inclusion and were subjected to numerous methods and techniques to achieve superior results than those documented in prior research endeavors. Emphasis was placed on enhancement and the swift identification of the most efficient and effective methods to yield distinctive outcomes when analyzing Arabic emotions and their respective dialects. Table 1 show the accuracy results of ArCyC test dataset.

Table 1. - Accuracy Ratio of our proposed work Deep Learning Methods on ArCyC Dataset

Accuracy%	Model	Spacy	Fasttext
Text	DNN	63%	71%
	LSTM	74%	78%
	CNN	75%	81%
	AraBERT	95%	
	BERT	92%	
Text & Emoji	DNN	92%	
	LSTM	95%	
	CNN	93%	
	Hybrid CNN+LSTM	92%	

The results we obtained were excellent, as we compared the two-word embedding methods and the comparison between deep learning models.

As a result, the Spacy inclusion was better than Fast Text because Spacy had been trained in a large number of languages and emojis, with approximately 1026 emoji symbols learned, and this result was better than fast text. In the work on deep learning models, we find that the LSTM in text and emoji and CNN in text model lead the two methods in embedding. In addition to the transformer (AraBERT), it was better than everyone, as it obtained 95%, the highest result compared to all other models. A previous study was conducted on the dataset, and BERT (transformer) was applied. The results were as described above.

4. Conclusions

This investigation addressed the challenge of limited resources for detecting cyberbullying in Arabic text, particularly in the context of Twitter, by leveraging advanced embedding models. To tackle this issue, the ArCyC dataset was utilized—comprising annotated Arabic tweets containing cyberbullying content. Through extensive experimentation with various deep learning models, the effectiveness of ArCyC was demonstrated, underscoring the important role of both textual features and emojis in identifying abusive behavior. The data were segmented into textual and symbolic components, with AraBERT outperforming other deep learning models on the ArCyC dataset. Prior research had not sufficiently explored the integration of Twitter-based emoji semantics with Arabic cyberbullying detection, making this study a significant contribution. Comparative analyses of word representation and embedding techniques revealed the strength of transformer-based models in capturing contextual meaning. AraBERT delivered the highest accuracy, while spaced word embeddings outperformed FastText in most cases. Among all architectures, CNN emerged as the top-performing model. LSTM obtained the best results (95%) for text and emoji. While in only text the AraBERT has the best results (95%). Future research will explore more sophisticated transformer architectures, advanced feature extraction methods, and the influence of sociolinguistic factors on Arabic cyberbullying to build even more robust detection systems.

Funding

None

ACKNOWLEDGEMENT

None

CONFLICTS OF INTEREST

The author declares no conflict of interest.

REFERENCES

[1] Van Hee, C.; Lefever, E.; Verhoeven, B.; Mennes, J.; Desmet, B.; De Pauw, G.; Daelemans, W.; Hoste, V. Detection and fine-grained classification of cyberbullying events. In Proceedings of the International Conference Recent Advances in Natural Language Processing (RANLP), Hissar, Bulgaria, 5–11 September 2015.

[2] Chen, Y. Detecting Offensive Language in Social Medias for Protection of Adolescent Online Safety. Master's Thesis, Penn State University, State College, PA, USA, 2011.

[3] Balakrishnan, V.; Khan, S.; Arabnia, H.R. Improving cyberbullying detection using Twitter users' psychological features and machine learning. Comput. Secur. 2020, 90, 101710. [CrossRef]

[4] Akhter, M.P.; Jiangbin, Z.; Naqvi, I.R.; Abdelmajeed, M.; Sadiq, M.T. Automatic detection of offensive language for urdu and roman urdu. IEEE Access 2020, 8, 91213–91226. [CrossRef].

[5] Kumar, R.; Lahiri, B.; Ojha, A.K. Aggressive and offensive language identification in hindi, bangla, and english: A comparative study. SN Comput. Sci. 2021, 2, 1–20. [CrossRef]

[6] Plaza-del-Arco, F.M.; Molina-González, M.D.; Urena-López, L.A.; Martín-Valdivia, M.T. Comparing pre-trained language models for Spanish hate speech detection. Expert Syst. Appl. 2021, 166, 114120. [CrossRef]

[7] Herwanto, G.B.; Ningtyas, A.M.; Nugraha, K.E.; Trisna, I.N. Hate speech and abusive language classification using fastText. In Proceedings of the 2019 International Seminar on Research of Information Technology and Intelligent Systems (ISRITI), Yogyakarta, Indonesia, 5–6 December 2019.

[8] Fortuna, P.; Soler-Company, J.;Wanner, L. How well do hate speech, toxicity, abusive and offensive language classification models generalize across datasets? Inf. Process. Manag. 2021, 58, 102524. [CrossRef]

[9] Alotaibi, A.; Hasanat, M.H.A. Racism Detection in Twitter Using Deep Learning and Text Mining Techniques for the Arabic Language. In Proceedings of the 2020 First International Conference of Smart Systems and Emerging Technologies (SMARTTECH), Riyadh, Saudi Arabia, 3–5 November 2020.

[10] Malmasi, S.; Zampieri, M. Challenges in discriminating profanity from hate speech. J. Exp. Theor. Artif. Intell. 2018, 30, 187–202.

[11] Garaigordobil, M.; Mollo-Torrico, J.P.; Machimbarrena, J.M.; Páez, D. Cyberaggression in adolescents of Bolivia: Connection with psychopathological symptoms, adaptive and predictor variables. Int. J. Environ. Res. Public Health 2020, 17, 1022. [CrossRef]

[12] Chatzakou, D.; Kourtellis, N.; Blackburn, J.; De Cristofaro, E.; Stringhini, G.; Vakali, A. Mean birds: Detecting aggression and bullying on twitter. In Proceedings of the 2017 ACM on Web Science Conference, Troy, NY, USA, 25–28 June 2017.

[13] Gitari, N.D.; Zuping, Z.; Damien, H.; Long, J. A lexicon-based method for hate speech detection. Int. J. Multimed. Ubiquitous Eng. 2015, 10, 215–230. [CrossRef]

[14] Zois, D.S.; Kapodistria, A.; Yao, M.; Chelmis, C. Optimal online cyberbullying detection. In Proceedings of the 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Calgary, AB, Canada, 15–20 April 2018.

[15] Di Capua, M.; Di Nardo, E. Unsupervised cyber bullying detection in social networks. In Proceedings of the 2016 23rd International Conference on Pattern Recognition (ICPR), Cancun, Mexico, 4–8 December 2016.

[16] González-Ibánez, R. Identifying sarcasm in twitter: A closer look. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Stroudsburg, PA, USA, 19–24 June 2011.

[17] Chia, Z.L.; Ptaszynski, M.; Masui, F.; Leliwa, G.;Wroczynski, M. Machine Learning and feature engineering-based study into sarcasm and irony classification with application to cyberbullying detection. Inf. Process. Manag. 2021, 58, 102600. [CrossRef]

[18] Lee, P.J.; Hu, Y.H.; Chen, K.; Tarn, J.M.; Cheng, L.E. Cyberbullying Detection on Social Network Services. In Proceedings of the 22nd Pacific Asia Conference on Information Systems, PACIS 2018, Yokohama, Japan, 26–30 June 2018.

[19] Krizhevsky, A., Sutskever, I., Hinton, G.E. (2012). Imagenet classification with deep convolutional neural networks. Communications of the ACM. https://doi.org/10.1145/3065386

[20] Anand, M., Eswari, R. (2019). Classification of abusive comments in social media using deep learning. 2019 3rd International Conference on Computing Methodologies and Communication (ICCMC), Erode, India, pp. 974-977. https://doi.org/10.1109/ICCMC.2019.8819734

[21] Li, Y., Algarni, A., Albathan, M., Shen, Y., Bijaksana, M.A. (2015). Relevance Feature Discovery for Text Mining. IEEE Transactions on Knowledge and Data Engineering, 27(6): 1656-1669. https://doi.org/10.1109/TKDE.2014.2373357

[22] Nobata, C., Tetreault, J., Thomas, A., Mehdad, Y., Chang, Y. (2016). Abusive Language Detection in Online User Content. Proceedings of the 25th International Conference on World Wide Web - WWW '16, pp. 145-149. https://doi.org/10.1145/2872427.2883062

[23] Kim, Y. (2014). Convolutional neural networks for sentence classification. arXiv.org.

[24] https://doi.org/10.48550/arXiv.1408.5882

[25] Zhang, X., LeCun, Y. (2016). Text Understanding from Scratch. arXiv:1502.01710 [cs], https://doi.org/10.48550/arXiv.1502.01710
[26] Prusa, J.D., Khoshgoftaar, T.M., Dittman, D.J. (2015). Impact of feature selection techniques for tweet sentiment classification. Proceedings of the 28th International FLAIRS Conference, 2015: 299-304.
[27] Hani, J., Nashaat, M., Ahmed, M., Emad, Z., Amer, E., Mohammed, A. (2019). Social Media Cyber-bullying Detection using Machine Learning. International Journal of Advanced Computer Science and Applications, 10(5):https://doi.org/10.14569/ijacsa.2019.0100587
[28] Ibrohim, M.O., Setiadi, M.A., Budi, I. (2019). Identification of hate speech and abusive language on Indonesian Twitter using the Word2vec, part of speech and emoji features. Proceedings of the International Conference on Advanced Information Science and System.
[29] https://doi.org/10.1145/3373477.3373495
[30] Waseem, Z., Hovy, D. (2016). Hateful Symbols or Hateful People? Predictive Features for Hate Speech Detection on Twitter. Proceedings of the NAACL Student Research Workshop, pp. 88-92.
[31] https://doi.org/10.18653/v1/n16-2013
[32] Vigna, F., Cimino, A., Dell'orletta, F., Petrocchi, M., Tesconi, M. (2022). Hate me, hate me not: Hate speech detection on Facebook. https://ceur-ws.org/Vol- 1816/paper-09.pdf., accessed on Dec. 2, 2022.
[33] Yenala, H., Jhanwar, A., Chinnakotla, M.K., Goyal, J. (2017). Deep learning for detecting inappropriate content in text. International Journal of Data Science and Analytics, 6(4): 273- 286.https://doi.org/10.1007/s41060-017-0088-4
[34] Islam, M.M., Uddin, M.A., Islam, L. Akter, A. Sharmin, S., Acharjee, U.K. (2020). Cyberbullying detection on social networks using machine learning methods. 2020 IEEE Asia-Pacific Conference on Computer Science and Data Engineering (CSDE), Gold Coast.
[35] https://doi.org/10.1109/csde50874.2020.9411601
[36] Shekhar A., Venkatesan, M. (2018). A Bag-of-Phonetic- Codes Modelfor Cyber-Bullying Detection in Twitter. 2018 International Conference on Current Trends towards Converging Technologies (ICCTCT), Coimbatore, pp. 1-7. https://doi.org/10.1109/ICCTCT.2018.8550938
[37] Sharma, R., Ramakrishnan, A., Pendse, P., Chimurkar, Talele, K.T. (2021). Cyber-bullying detection via text mining and machine learning. 2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT), Kharagpur, pp. 1- 6.
[38] https://doi.org/10.1109/ICCCNT51525.2021.9579625
[39] Wadhwani, A., Jain, P., Sahu, S. (2021). Injurious Comment Detection and Removal utilizing Neural Network. 2021 International Conference on Innovative Practices in Technology and Management (ICIPTM), Noida, pp.165-168.https://doi.org/10.1109/ICIPTM52218.2021.9388331