Journal of Computer Science

Original Research Paper

Classification of Arabic Comments to Detect Cyberbullying from Social Media Using Convolutional Neural Network and Meta-Learning

[1,2]Sarah Mansour Elgaud, [3]Mustafa Ali Abuzaraida and [4]Abdullah Alshehab

[1]*Department of Computer Science, Libyan Academy, Misurata, Libya*
[2]*Department of Computer Science, College of Information Technology, Al-Asmaria Islamic University, Libya*
[3]*Department of Computer Science, Faculty of Information Technology, Misurata University, Libya*
[4]*Department of Computing, College of Basic Education, Public Authority of Applied Education and Training, Ardeya, Kuwait*

Article history
Received: 19-09-2024
Revised: 04-11-2024
Accepted: 25-11-2024

Corresponding Author:
Mustafa Ali Abuzaraida,
Department of Computer Science, Faculty of Information Technology, Misurata University, Libya
Email: abuzaraida@it.misuratau.edu.ly

Abstract: As a result of the proliferation of social media, cyberbullying has become widespread in the Arab community on social media and cyberbullying has become a concern targeting individuals and may cause some serious side effects. The problems of Natural Language Processing (NLP) for the Arabic language and its complexity make it difficult to classify texts accurately. In recent years, deep learning models have emerged as a viable option for solving some of the NLP problems. In this study, we constructed a hybrid approach of Convolutional Neural Network (CNN) and Meta-learning for classifying cyberbullying Arabic comments. A set of electronic text data in Libyan dialect and Modern Standard Arabic was collected from several Libyan social media platforms such as Facebook, YouTube, and other online platforms to identify instances of cyberbullying on social media. Pre-processing is a vital part of the data preparation process for detecting cyberbullying, where a CNN model was trained on the data. Finally, the model was evaluated for accuracy, recall, precision, and F1 scores. Thus, the results showed that CNN outperforms better when combined with Meta and gave higher results than CNN only. We obtained the best classification accuracy of 98, 91, and 84% for three datasets. The accuracy of CNN alone was 71, 69, and 52% respectively for the three experiments. These results confirm the success of the model and the improvement in CNN performance with Meta and that it gives better results than CNN. These results confirm the potential of neural networks in developing and succeeding in cyberbullying detection systems.

Keywords: Cyberbullying, Natural Language Process, Deep Learning, Convolutional Neural Network, Meta-Learning, Text Mining

Introduction

With the rapid growth of technology, it is very hard to get insights into this data to make sensible judgments in this information era, as data is being created at an unprecedented pace by both people and algorithms. As a result, abusive language has become a widespread issue on social media sites (Alakrot *et al.*, 2018). Many different kinds of abusive language, including cyberbullying, poisonous remarks, hate speech, and offensive material, may be found on social media sites (Haidar *et al.*, 2018).

Recently, using platforms for social media like X (formerly Twitter), Facebook, Instagram, and YouTube has been increasing. A huge number of comments (texts) are being posted on these platforms that express users' opinions on various topics. Facebook, X, and YouTube are among the most widely used social media applications in the world in general and the Arab region in particular (Mubarak *et al.*, 2017; Baiganova *et al.*, 2024). The mentioned platforms allow bullies to intimidate and harass people due to the lack of oversight by government agencies. In addition to the lack of awareness through social media and a law that protects the rights of those affected by cyberbullying. Throughout the Arab countries, the use of social media has become entrenched, and thus cyberbullying has become a real issue of concern. However, defining cyberbullying in Arabic

Science Publications

increases the challenges due to the complexity of the Arabic language, the different dialects and the informal language mostly used on social media (Perera and Fernando, 2024).

Cyberbullying occurs when an individual or group targets someone who is vulnerable and uses technology, such as a computer, mobile phone, or other electronic device, to harass, threaten, or otherwise abuse them. Among the many manifestations of cyberbullying include the transmission of damaging or hostile messages, the creation of false accounts, the posting of victims' private or humiliating images, and the use of foul language and profanity. Cyberbullying is more harmful than conventional bullying because bullying-related postings may stay online for a long time and reach many people (Kanan *et al.*, 2020).

The occurrence of Cyberbullying is when someone uses the Internet to harm or annoy a famous or ordinary person on a social media platform. It includes posts, comments, messages, conversations, live broadcasts, photos, videos, and emails, which include explicit and rude sexual references, hateful, offensive remarks related to people's race, religion, and/or country (Haidar *et al*., 2017; Kanan *et al*., 2020).

There has been a lot of effort to find ways to identify cyberbullying in recent years about the categorization of texts, particularly those written in the English language (Enamoto *et al*., 2021). Since cyberbullying is a problem facing the whole world, this problem also faces the Arab nation and only a small number of research has been conducted to work on discovering cyberbullying in the Arabic language.

The Studies of cyberbullying detection field have increased and cyberbullying on pages dedicated to Arabic content on different platforms is still restricted and considered problematic for several reasons, among these reasons are that spoken Arabic contains very complex morphological entities and Instead of speaking Modern Standard Arabic, most Arabs use colloquial Arabic (Husain, 2020). The truth is that there are different words that are not allowed in Arab culture, while in other cultures, they are very accepted. For example, the words "حمار" (Donkey) or "كلب" (Dog) are considered to be agricultural animals. Are considered to be agricultural animals. However, these forms of expression may not be used in different contexts, such as describing people or actions.

Recent studies like (Alkhatib *et al.,* 2024) have shown that among the many areas where Deep Learning has shown promise is Natural Language Processing (NLP). Convolutional Neural Networks (CNNs) in particular have shown exceptional performance in text classification tasks, thanks to their ability to grasp sequential relationships and identify patterns in data.

The task of classifying Arabic texts and detecting abusive language and cyberbullying is a major challenge because the terminology used on social media is often casual and vague, where the shape of letters and the spelling of words vary according to their context (spelling ambiguity). It is well known that Arabic does not receive the same level of attention as English. However, recently it has been observed a significant effort has been made in Arabic sources to detect cyberbullying. This study explores the importance of deep learning in detecting cyberbullying on social media using Arabic.

In this study, we propose to use a Convolutional Neural Network (CNN) model with Meta to detect social media bullying in Arabic and provide a new dataset for detecting social media bullying in the Libyan dialect. We also seek to improve the CNN model to obtain distinctive results for the Arabic language. For this purpose, we extracted a large number of hateful and offensive posts from Twitter, YouTube, and Facebook using keyword search strategies and user profiles. Based on a reliable dataset on hate speech and offensive speech in Arabic and the collected dataset. Cyberbullying text classification in Arabic is the primary focus of this research using a hybrid approach of CNN and meta-learning algorithms. After that, the meta-learning affecting CNN will be evaluated by measuring the accuracy. The results will help to benefit from and improve the CNN deep learning model in classifying Arabic texts and detecting cyberbullying on social media.

Related Studies

Recently, a significant amount of research has been conducted on hate speech and cyberbullying across a range of languages, like English (Nikitha *et al*., 2024) Turkish (Karayiğit *et al*., 2021), Urdu (Dewani *et al*., 2021), Ethiopian (Ganfure, 2022) and Bengali (Ghosh *et al*., 2021). These recent studies were conducted for different languages around the world and indicate that this field of research is active worldwide. The field of Arabic language research, on the other hand, has done very little research on cyberbullying detection (Habberrih and Abuzaraida, 2024a). One of the recent studies was done by (Rachid *et al*., 2020). They presented a study using different neural network models (CNNs and RNNs) and pre-trained word embeddings for classifying Arabic comments from news channel datasets into "cyberbullying" and "non-cyberbullying" categories. The different data were balanced to account for the class imbalance inherent in cyberbullying cases. Simple and combined CNN/RNN models successfully detected cyberbullying in Arabic with a high degree of accuracy (up to 84% F1-score). Machine learning models, especially those using TF-IDF features, perform competitively with deep learning models. Complex hybrid models can be effective when trained on balanced datasets.

In the same context as previous researchers on the possibility of deep learning to automatically detect hate speech in Arabic comments on social media and other platforms, Anezi (2022) performed a study on the development of a Deep Recurrent Neural Network (DRNN) for accurate classification. Building a new dataset for hate speech in Arabic that includes seven categories. The proposed DRNN models achieved high accuracy rates: 96.73% for binary classification, 95.38% for three-class classification, and 84.14% for seven-class classification. These results outperform current methods for detecting hate speech in Arabic. The developed dataset and DRNN models contribute valuable tools for further research and development in this field (Anezi, 2022).

Similar to the difference in dialects, a study by (Mazari and Kheddar, 2023) has been presented to reveal the toxic and hateful content in the Algerian Arabic dialect. They created a new annotated dataset of 14,150 Algerian dialect comments classified as toxic, collected from Facebook, YouTube, and X. For the model evaluation, a comparison of several traditional Machine Learning (ML) and Deep Learning (DL) models for detecting toxic content in an Algerian dialect dataset was done. The Bi-GRU model achieved the best performance among DL models with 73.6% accuracy and 75.8% F1 accuracy outperforming traditional ML models. Recurrent Neural Network (RNN) models such as Bi-GRU are more suitable for text analysis than Convolutional Neural Networks (CNN), especially for sequential data such as dialects.

AlKhamissi and Diab (2022) presented a multi-task ensemble model for the accurate detection of hate speech in Arabic tweets, focusing on three subtasks, detection of hate speech, detection of harmful language, and precise categorization of hate speech. The proposed model, AraHS, used multi-task learning and self-consistency correction. Achieve significant improvements compared to previous baselines. AraHS achieved an F1-macro score of 82.7% on the hate speech detection subtask, an increase of 3.4% compared to previous work (AlKhamissi and Diab, 2022).

Shaker and Dhannoon (2024) presented a study using Twitter (currently known as X) comments on two datasets used to identify instances of cyberbullying. The Arabic Cyberbullying Dataset was the first dataset, while the English Cyberbullying Dataset was the second. In the original and pre-processed datasets, the words were represented by three distinct sets of pre-trained global vectors (GloVe) with varying dimensions. Recursive classifiers (GRU), recursive classifiers (BiGRU), Long Short-Term Memory (LSTM), Bidirectional LSTM (BiLSTM), and RNN have all been used, assessed, and contrasted. GRU performed better than other classifiers on both datasets, according to the findings. Its accuracy was 93.38% on the pre-trained English dataset and 87.83% on the Arabic cyberbullying dataset (Shaker and Dhannoon, 2024).

Habberrih and Abuzaraida discussed the use of machine learning using TF-IDF and N-grams for sentiment analysis in the central area of Libya. The study focuses on sentiment analysis in the central Libyan dialect using ML classifiers and several feature extraction methods, including TF-IDF, Unigrams, and Trigrams. In the tests, the Support Vector Machine (SVM) produced the best accuracy. The study concluded that Unigrams can enhance the classifier performance, while Trigrams may lead to inferior performance. 22,762 records were included in the used dataset. Notably, Logistic Regression (LR) had the maximum accuracy of 58.60% in the second trial, while Support Vector Machines (SVM) had the highest accuracy of 69.04% in the first and 68.92% in the third. Furthermore, LR performed 70.49% better in the first trial than the other classifiers in all tests in terms of accuracy (Habberrih and Abuzaraida, 2024b).

They extended their work by another study. The study presents the effect of using crossword and stop-word removal techniques on ML classifiers, SVM, and LR in detecting emotions from poetry in the Libyan dialect. In addition to TF-IDF, other pre-processing methods were used to extract characteristics from a collection of Unigrams and Trigrams. The findings indicate that the classifier's performance may suffer as a consequence of the paused word removal strategy. In both trials, SVM performed better than LR, with an accuracy of 71.63%, while in the second experiment, LR attained an accuracy of 70.92%. (Habberrih and Ali Abuzaraida, 2024).

A study by Xingyi and Adnan (2024) presented a BERT pre-training model approach for word embedding. A convolutional layer was used to capture multi-scale local semantic features of text and a bidirectional simple recurrent unit with a built-in attention mechanism for contextual semantic modeling at different levels. To jointly train sentiment analysis and cyberbullying detection, a multi-task learning architecture is suggested. The framework includes word vector transformation, deep text feature extraction, and classification using BERT, MTL, and a transformer architecture. The framework provides the task of text sentiment classification as an auxiliary resource to improve detection accuracy and generalization ability. Results from experiments showed that, in comparison to conventional models, the suggested model could comprehend semantic information more effectively, making it easier to identify potential cyberbullying comments online. The proposed framework effectively improves cyberbullying detection accuracy and generalization ability by combining BERT and MTL. The use of attention and MTL mechanisms improves the model’s performance and BERT word embeddings capture more complex semantic relationships.

The researchers (Ahmad Al-Khasawneh *et al*., 2024) presented a cyberbullying detection framework that uses three modules to extract unique information from different media within a social network, highlighting the importance of integrating different data media and addressing ethical considerations. The research used a number of machine learning methods, such as Support Vector Machines (SVMs), Random Forests, and Naive Bayesian and logistic regression with deep learning models, which are Hierarchical Attention Network (HAN) and LSTM. The study's superior performance over a number of cutting-edge models demonstrated how well it handles the intricate nature of cyberbullying in social networks. The MMCD model achieved high performance compared to existing models, with higher F1 scores and higher accuracy scores across all datasets. The performance of the model is improved by integrating attention mechanisms and comprehensive features. The study demonstrates the effectiveness of the MMCD model in detecting cyberbullying content on social media platforms, highlighting the importance of integrating multimedia and attention mechanisms. The study concludes that future efforts should focus on improving the current approach, investigating advanced deep learning architectures, and integrating real-time data processing capabilities to enhance the construction of cyberbullying detection systems.

The researchers (Musleh *et al*., 2024) presented a study of a machine learning-based approach to detect cyberbullying in Arabic tweets. This study focused on collecting and processing a standard dataset and applying machine-learning algorithms. The study used SVM, NB, RF, LR, LightGBM, CatBoost, XGBoost, AdaBoost, and Bagging algorithms for classification. The dataset was preprocessed using techniques such as segmentation, filtering, normalization, and partitioning. For the feature extraction methods, TF-IDF and N-gram were used. Data collection: Arabic tweets were collected from Twitter APIs. The results of the study showed that the XGBoost algorithm achieved the highest accuracy of 89.95% in detecting cyberbullying in Arabic tweets, using TF-IDF as the feature extraction method. The study concluded that machine learning-based methods can be effective in detecting cyberbullying in Arabic tweets and that the proposed approach can be improved by applying hybrid models.

In the same context, (Daraghmi *et al*., 2024) presented our hybrid deep learning approach to detect cyberbullying in Arabic, by combining the strengths of CNN, Bi-LSTM, and GRU models with stacked word embedding using a grid search method. The study uses grid search technology to fine-tune hyperparameters and integrate GloVe and FastText micro words as an approach to represent text. The proposed hybrid model achieves an accuracy of up to 98.83%, outperforming individual models in detecting cyberbullying. The CNN-Bi-LSTM-GRU hybrid model is effective in detecting cyberbullying in Arabic and its integration with a typical tool based on mobile forensics can help monitor and mitigate cyberbullying cases across different social media platforms, as the hybrid model outperforms individual models in detecting cyberbullying, achieving an accuracy of 98.83%, high recall, precision, specificity, and F1 score. The model architecture is created to take advantage of the advantages of CNN, BiLSTM, and GRU while addressing their individual limitations.

Materials

In this study, Python version 3.10 was chosen as the programming language for implementing the proposed system. Python provides a variety of libraries and modules for working with artificial intelligence. The libraries used in this study, which include.

Data Analysis Libraries:

Pandas (pd): A powerful library for data manipulation (loading, cleaning, transforming, analyzing) and creating tabular data structures (DataFrames, Series).

Plotly.express (px): Enables the creation of various interactive and visually appealing charts and graphs using a high-level declarative syntax.

Plotly.graph_objects (go): Provides lower-level building blocks for customized plotly visualizations, offering more control over chart elements and interactions.

Seaborn (sns): Built on top of matplotlib, seaborn offers a convenient and aesthetically pleasing API for creating statistical visualizations like scatterplots, heatmaps, violin plots, and more.

Matplotlib.pyplot (plt): A fundamental library for creating static and interactive visualizations, providing a wide range of plot types and customization options.

Word Processing Libraries

NLTK: The Natural Language Toolkit (nltk) provides a comprehensive set of tools for natural language processing (NLP) tasks.

Farasapy: A library for Arabic language processing, which is a powerful toolset for natural language processing in Arabic, within Python code. It allows tasks such as encoding, morphological analysis, and named entity recognition of Arabic text.

Tashaphyne: A library for processing the Arabic language. It is a Python module for processing Arabic text that works as a source and optical splitter. Its primary purpose is to perform light derivation, which involves removing prefixes and suffixes from words, and generates all possible hashes for the resulting roots.

Machine Learning Libraries

Scikit -learn (sklearn): A popular toolkit for machine learning tasks, including:

Tfidf Vectorizer: It is used to generate TF-IDF (Term Frequency - Inverse Document Frequency) features of text data, representing the importance of words in a document to the entire set of documents.

StandardScaler: Standardize features by removing the mean and scaling to unit variance.

Deep Learning Libraries

Tensorflow: A powerful library for numerical computation and deep learning.

keras: A high-level API built on TensorFlow, simplifying the creation and training of deep learning models.

Other Libraries

Emoji: Allows for working with emoji characters in text data.

Numpy (np): Provides powerful functions for numerical computations and array manipulation.

Re: Regular expressions for string processing and pattern matching.

Methods

There will be four phases to the presentation of this study's methodology. The first step involves gathering data and the second step involves pre-processing and cleaning the obtained and annotated data to eliminate unnecessary tokens and symbols. A deep learning model in this example, the CNN model, and the meta-learning algorithm are used to train the data in the third step. The assessment is done in the last stage when the models are used to determine how successful the suggested method is. The methodology used in this investigation is shown in Fig. (1).

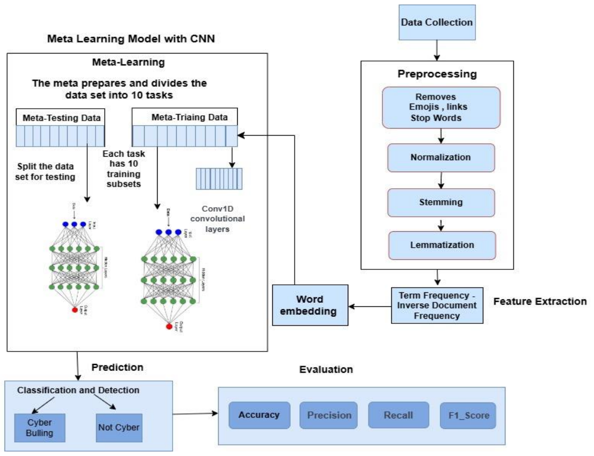

Data Collection
Meta Learning Model with CNN
Meta-Learning
The meta prepares and divides the data set into 10 tasks
Meta-Testing Data
Meta-Triaing Data
Split the data set for testing
Each task has 10 training subsets
Conv1D convolutional layers
Preprocessing
Removes Emojis , links Stop Words
Normalization
Stemming
Lemmatization
Word embedding
Term Frequency - Inverse Document Frequency
Feature Extraction
Prediction
Classification and Detection
Cyber Bulling
Not Cyber
Evaluation
Accuracy
Precision
Recall
F1_Score

Fig. 1. Proposed methodology Phases

Datasets

The study uses three datasets; the first dataset is the Libyan dialect data. This data was gathered manually from social media sites including YouTube, Facebook, Instagram, and Twitter from the Libyan Arabic-speaking Internet community. This dataset took about a month to be collected from the mentioned social media. This data was in the field of politics in Libyan affairs, through Libyan news pages and groups, and represents a wide range of political positions on various topics from 2019 to the recent events of 2023 in Libyan affairs. It includes comments from people of different age groups and genders.

To annotate the dataset, we asked a group of Arabic language specialists to manually annotate each sentence. Here, an additional attribute was added to the dataset which is "bullying" for comments that contain abuse, insults, or offensive as bullying class. While neutral if there are non-offensive comments. A majority vote method was used to group the largest number of votes for each category into the “bullying” or “neutral” category.

The second dataset is available from https://www.kaggle.com/datasets/alanoudaldealij/arabic-cyberbullying-tweets and is expressed through MSA, meaning there are no accents. Sourced from social media platforms, this collection has been annotated with binary labels, indicating the presence or absence of cyberbullying behaviors in each comment. The binary rating is coded with “1” representing “bullying” and “0” representing “non-bullying”. The structure of the dataset is divided into two main columns: “Comments” which contain the raw textual data and “Type” which are the corresponding binary labels.

The third dataset is available from https://github.com/omammar167/Arabic-Abusive-Datasets. Since these datasets were collected from Arabic websites, the comments and responses are expressed in standard Arabic with no specific dialects. This dataset is regarded as balanced as it includes an equal amount of comments, both positive and negative. Table (2) shows the sampling distribution of the dataset across each category.

Table 1: The first dataset's sample count

Class	Number of samples
Bullying	3843
Neutral	3648
Total	7491

Table 2: The second dataset's sample count

Class	Number of samples
Cyber	6838
Not Cyber	6355
Total	13,193

Data Pre-Processing

The preprocessing techniques used in this study follow these steps which based on (Dien *et al*., 2019) study:

- Stop words: One stage in pre-processing texts before analysis is the word removal of stop words. Its foundation is the removal of frequent terms that add no value to the text and do not aid in text analysis or the extraction of the text's requirements. A stop list is a list that includes every word in a phrase, such: Pronouns (أنا، هو، هي، أنت، نحن، هم), conjunctions (في، على، إلى،), prepositions (ثم، حتى، أو، ولكن), words that are not useful in classification such as (بعض، أيضا، حاليا،), directions (أمام، خلف، بجانب، قبل، بعد، يسار يمين) and any term that doesn't provide the text any more significance, such (أمس، اليوم، غداً) in addition to the additional words those are added to the list in the Libyan dialect, such as in Table (3)

The first experiment will include a balanced dataset that has been collected to be used in this study and stopword removal techniques will be used specifically for the Libyan dialect. A total of 1,300 stop words for the Libyan dialect were extracted, of which 500 were extracted from (Jarrar *et al*., 2023; Omar *et al*., 2022) and 500 from (Habberrih and Abuzaraida, 2024b). Only about 300 words were added as stopwords in line with the dataset of this study to avoid deleting important words that might affect the accuracy of the classifier. These stopwords were used during the pre-processing phase, specifically for the stopword removal technique:

- Tokenization: The processed text then undergoes tokenization, a crucial procedure in which the text is broken down into tokens essentially distinct units of meaning, usually words or phrases. Using the “Tokenizer” interface of the Keras library, comments are converted into sequences of integers, creating a digital representation of the text which is a prerequisite for processing by neural network models (Mubarak, 2019)
- Normalization: Throughout this procedure, terms are standardized by removing any possible misinterpretations of letters. Then, replacing some Arabic letters with their official form was performed due to the common spelling error in some words, such as in Table (4)

Table 3: Examples of the stop words list in the Libyan dialect

Libyan dialect	Meaning in Arabic	Meaning in English
معش	لا تفعل	Don't
شوف	انظر	Look
هلبة	كثيراً	Much
كنك	ماذا حدث	What happened
هكي	هكذا	Thus
شني	ماذا	What
خلاص	يكفي	Enough, that’s it
خلي	اترك	Leave it

Table 4: Some Arabic letters with their form are replaced

Arabic letters	Replacement
Replace the letter (ة)	with the letter (ه)
Replace the letter (ى)	with the letter (ي)
Replace the letter (إ، أ، ا)	with the letter (ا)
Replace the letter (ؤ)	with the letter (ء)
Replace the letter (گ)	with the letter (ك)

Normalization of the dataset before classification is often necessary to ensure that all variations in the spelling of particular words are converted into a standard form (Zeroual and Lakhouaja, 2017):

- Stemming: Derivation is done and aims to reduce words to their basic uninflected forms. Although it may be helpful since similar words are often connected to the same stem, even if the stem is not a real root, it can sometimes be distinct from the root (Alduailaj and Belghith, 2023). In this study, the Tashaphyne library was used to apply the Stemming technique to the dataset. The Farasa root was used as an additional tool: Farasa provides data extraction because words have different structures, especially in Arabic (for example, "حمار", "حمارة", "يالحمار" and "حمير"). The number of features is reduced when using stem, as both suffixes and prefixes are removed. Derivation is performed before classification. Inflected terms are reduced to their root format and all terms are grouped together using a single base word “donkey”, which is the root of these words (Salama *et al*., 2018)
- Lemmatization: This step is a linguistic technique that entails examining a word's shape, removing its inflectional affix, and creating its base form. A lemma is a specific model that represents the lexicon. The latter corresponds to a group of all word forms that share the same semantic and grammatical structure. In the Arabic language, the verb lima chosen is the present perfect, indicative, and pronoun, as this technique is used to return words to their original root based on the morphological analysis of the sentence. The process of the slurry technique is described in Table (5)

Feature Extraction

Selecting and extracting the best features for the text classification process and eliminating redundant, noisy, and irrelevant data are steps in the feature extraction process. In the end, this phase increases the analysis's efficacy and efficiency by decreasing the feature space's dimensions and processing time. The following feature extraction method will be used in this study Term Frequency-Inverse Document Frequency (TF-IDF).

Table 5: Lemmatization process

Lemmatization	Sentence
Before	أي قبائل ترضى أن يحمي إخوتهم روس
After	اى قباءل ترضى ان يحمى اخوتهم روس
Before	شن هلغباء القناة الهبلاء كلها أغبياء
After	شن هلغباء القناه الهبال كلها غباء

The relevance of a term in a document is assessed using TF-IDF, where word relevance is defined as the quantity of information supplied in the term's context. The statistic known as phrase Frequency (TF) measures how often a phrase occurs in a text. A word is more relevant to the content than other terms if it occurs more often in the text. Furthermore, by dividing the total number of documents by the total number of documents in the group they include, the Inverse Document Frequency score (IDF) is determined (Alammary, 2021).

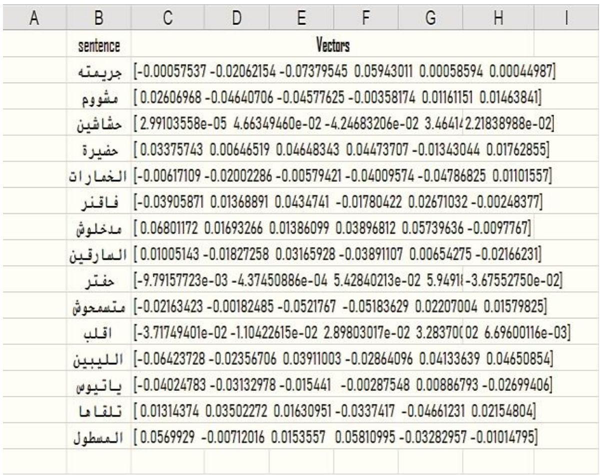
Word Embedding

The final stage of the preprocessing phase is initializing the embedding layer within the neural network. The embedding dimension is specified, which determines the size of the vector space in which the words will be represented into vectors (Li and Gong, 2021). The length of each word is 100, 200, and 300 dimensions provided by Keras. This function makes it possible to organize the text corpus by turning each text into a string of integers, where each True represents the index of a dictionary word.

The words will be taught in the deep learning model's embedding layer before being passed on to the next layer. In this process, data is converted to another format in a manner, and categorical data is translated into a numerical format, which facilitates the model's classification efforts by providing a clear and measurable goal for the algorithmic prediction as in Fig. (2).

Data Splitting

In this study, both datasets are split into train and test subsets. This follows a traditional 80-20 split, where The training set contains the majority of the data that the model will use to learn. While the smaller distinct portion is reserved for testing tasks. The training set for the first dataset contains about 6,000 samples, representing 80% of the data, while the test set contains 1,600 samples, representing the remaining 20%. For the training set for the second dataset, there are 3,000 samples in the training set (80%) and 1,000 samples in the test set (20%). Similarly, the dataset for the third dataset contains about 5,000 samples (80%) and the test set contains 1,500 samples (20%). This test set is useful for an unbiased assessment of the model's predictive ability.



A	B	C	D	E	F	G	H	I
	sentence	Vectors						
	جريمته	[-0.00057537 -0.02062154 -0.07379545 0.05943011 0.00058594 0.00044987]						
	مشووم	[0.02606968 -0.04640706 -0.04577625 -0.00358174 0.01161151 0.01463841]						
	حشاشين	[2.99103558e-05 4.66349460e-02 -4.24683206e-02 3.4641 2.21838988e-02]						
	حضيرة	[0.03375743 0.00646519 0.04648343 0.04473707 -0.01343044 0.01762855]						
	الخمارات	[-0.00617109 -0.02002286 -0.00579421 -0.04009574 -0.04786825 0.01101557]						
	فاقنر	[-0.03905871 0.01368891 0.0434741 -0.01780422 0.02671032 -0.00248377]						
	مدخلوش	[0.06801172 0.01693266 0.01386099 0.03896812 0.05739636 -0.0097767]						
	السارقين	[0.01005143 -0.01827258 0.03165928 -0.03891107 0.00654275 -0.02166231]						
	حفتر	[-9.79157723e-03 -4.37450886e-04 5.42840213e-02 5.9491 -3.67552750e-02]						
	متسمحوش	[-0.02163423 -0.00182485 -0.0521767 -0.05183629 0.02207004 0.01579825]						
	اقلب	[-3.71749401e-02 -1.10422615e-02 2.89803017e-02 3.28370 02 6.69600116e-03]						
	الليبين	[-0.06423728 -0.02356706 0.03911003 -0.02864096 0.04133639 0.04650854]						
	ياتيوس	[-0.04024783 -0.03132978 -0.015441 -0.00287548 0.00886793 -0.02699406]						
	تلقاها	[0.01314374 0.03502272 0.01630951 -0.0337417 -0.04661231 0.02154804]						
	المسطول	[0.0569929 -0.00712016 0.0153557 0.05810995 -0.03282957 -0.01014795]						

Fig. 2: Representing words into vectors

Training Parameters

For the training process, these parameters are considered for the three experiments:

- Optimization technique: The Adam Optimiser is a well-liked option for deep learning due to its capacity to modify the learning rate for every parameter separately. This helps the model converge faster and achieve better optimization compared to fixed learning rate methods
- Loss measurement: Class cross loss: The difference between the expected probabilities is measured by this function (how likely the model is to believe that a sample belongs to each class) and the actual class labels. It is ideal for multi-class classification problems because it penalizes the model for assigning high probabilities to incorrect classes. The model is encouraged to give the right class greater probability when this loss function is decreased
- Evaluation metric: Accuracy (ACC): This metric simply calculates the percentage of samples that the model correctly classified. It provides a clear picture of how well the model performs overall
- Training duration: 10 Epochs: The complete training dataset is traversed once for each epoch. 10 epochs of training enable the model to repeatedly learn from the data. With each epoch, the model updates its internal weights, gradually improving its performance
- Validation strategy: 20% validation set: This is a common approach to prevent overfitting. 20% of the training data is retained in this case and isn't used for direct training. Rather, it is used to assess the model's performance as it is being trained. This makes it more likely that the model is picking up on significant patterns rather than just learning the training set

Classification Process

Classification is the process of predicting the popularity of a document by assigning it to a particular class or category. This can be achieved by using different classification techniques or classifiers (Xu and Du, 2020). Basically, predicting qualitative responses or document categories is equivalent to document classification.

In this study, a model based on Meta-learning and the proposed deep CNN model was built as shown in Fig. (1). The work depends on the research task of classifying Arabic comments to detect cyberbullying. The dataset will be divided into 10 tasks for training and testing, where each task has 10 training subsets. This process will be implemented using Meta-learning.

Meta-Learning

Meta-learning was incorporated in this study into the methodology to address the challenge of improving a CNN tasked with classifying Arabic tweets for indicators of cyberbullying. A series of distinct but interrelated tasks were used, each representing a different aspect of the classification challenge to train the meta-model. This approach aims not only to enhance the model's functionality on the first mission but also to give it the flexibility to handle variations in data and task structure effectively.

The process began by defining a number of meta-tasks, where each task includes a unique set of training and testing samples drawn from the overall dataset. Then, proceeded to the meta-training phase, where each task was used to train a version of the CNN architecture. This phase was crucial, as it allowed the model to confront a variety of learning scenarios, effectively teaching it how to learn from diverse datasets representing different aspects of the cyberbullying phenomenon.

Meta Training and Testing

- Meta-training phase: Given a set of tasks (T), each with its own dataset (D^{T}), seeking to find a set of meta-parameters (θ) that can generalize across all tasks:

$$\theta^{*} = arg\ \underset{\theta}{argmax} \sum_{T_i \in T} log\ log\ p\ \left(D^{T_i}_{\boldsymbol{meta-train}}\right) \tag{1}$$

where, $(D^{T_i}_{meta-train})$ is the meta-training data for a task (T_i).

As the model interacted with each task, capturing the accuracy training and validation losses is required, creating an empirical account of the model's learning path across tasks. These metrics were crucial to understanding how the model adapts and improves its learning strategy with each new task it encounters.

- Adaptation phase (for each task): For each task (T_i), a further tune of the model parameters (ϕ) is computed using the task-specific training data $\left(D^{T_i}_{train}\right)$:

$$\phi^{*}_{T_i} = log\ log\ p\ \left(D^{T_i}_{\boldsymbol{train}}, \theta^{*}\right) \tag{2}$$

- Task-specific training and testing:

  – *Training:* Training the CNN model on $(D^{T_i}_{train})$ to obtain task-specific parameters $(\phi^{*}_{T_i})$
  – *Testing:* Evaluating the performance of the task-specific model on $(D^{T_i}_{test})$ using the learned parameters$(\phi^{*}_{T_i})$

- Loss optimization: The objective is to minimize the loss on the meta-training tasks and maximize the model's ability to adapt to new tasks quickly:

$$\theta^{*} = \underset{\theta}{argmax} \sum_{T_i \in T} L\left(f_{\theta}\left(D^{T_i}_{\boldsymbol{train}}\right), D^{T_i}_{\boldsymbol{test}}\right) \tag{3}$$

Where, $(\ L\ )$ is the loss function, (f_{θ}) represents the CNN model parametrized by (θ), $(D^{T_i}_{train})$ is the training data for task (T_i) and $(D^{T_i}_{test})$ is the corresponding test data.

Architecture of Convolutional Neural Network

The architecture used in this research is built on the Keras sequential model framework. The sequential model is a linear combination of layers: A CNN is created with the following layers:

- Embedding layer: Maps words to numerical representations, capturing semantic relationships
- 1D convolutional layers: Extracts features from sequences and identifies patterns within text
- Global maximum pooling layer: Summarizes the extracted features and provides a summary representation
- Dense layers: Perform further processing and classification, turning features into predictions

Dropout layer: Introduces randomization to prevent overfitting, which improves the generalizability of the model.

Results and Discussion

Three experiments were conducted in this study. The model is trained on three datasets of different sizes (946KB, 519KB, 1,290KB) respectively to each experiment. Along with a balanced dataset, different pre-processing techniques were applied on all the experiments, such as stemming, lemmatization and stopword removal. In addition to encoding, normalization and data cleaning.

The first experiment included a dataset collected for this study where the Libyan dialect unwanted word removal techniques were used. Then TF-IDF was treated as a feature extraction technique. The CNN algorithm was used in the first part of the experiment to compare the results and show the improvement that can be achieved by using CNN + Meta-learning. We will go through the steps of the pre-processing and feature extraction techniques for the rest of the experiment. As shown in Table (6), the results of the first experiment were obtained, which included unwanted word removal and analysis in addition to TF-IDF. The experiment was conducted by applying Stemming, Lemmatization, and Stop-words in the first part of the CNN algorithm without using Meta-learning, and in the second part, these parameters were applied to CNN + Meta-learning using the same pre-processing techniques that included unwanted word removal and analysis in addition to TF-IDF performing Stemming, Lemmatization, and Stop-words.

The second experiment included an available dataset that has been used in previous studies (Habberrih and Ali Abuzaraida, 2024). The same steps for the first experiment on CNN were used, the second section and CNN + Meta-learning. The results of the second experiment gave higher accuracy than the previous experiment as illustrated in Table (7).

The third experiment will include another available dataset that has been used in a previous study (Khairy *et al*., 2024). This experiment included the same steps as the first and second experiments. The results show its superiority in terms of accuracy to the first and second experiments, as the accuracy reached 98%, as shown in Table (8).

Table 6: Results of the first experiment

Classified	Preprocessing	Accuracy (%)
CNN	Stemming, stop-words	70
	Stemming and, lemmatization & stop-words	71
CNN + meta-learning	Stemming, stop-words	82
	Stemming, lemmatization & Stop-words	83

Table 7: Results of the second experiment

Classified	Preprocessing	Accuracy (%)
CNN	Stemming, stop-words	70.5
	Stemming, lemmatization & Stop-words	69.1
CNN + Meta-Learning	Stemming, stop-words	91
	Stemming, lemmatization & stop-words	91.3

Table 8: Results of the third experiment

Classified	Preprocessing	Accuracy (%)
CNN	Stemming, stop-words	51.1
	Stemming, lemmatization & stop-words	52
CNN + meta-learning	Stemming, stop-words	98.6
	Stemming, lemmatization & Stop-words	98.3

Through the three experiments and dealing with TF-IDF as a feature extraction technique. By applying Stemming, Lemmatization, and Stop-words, we noticed their effect in the first section of the CNN algorithm, while in the second section, the effect of applying these parameters to CNN + Meta-learning was better and thus the results were better using the same pre-processing techniques with CNN + Meta-learning.

To assess the efficacy of the suggested approach, the confusion matrix of the descriptive model and the distribution of the predictions of the confusion model are shown in Figs (3-5) for the three experiments respectively. While analyzing the confusion matrix, it can be easily observed that the model correctly predicted non-bullying (true negatives) and correctly identified the cases as bullying (true positives). These figures demonstrate the model's efficacy in detecting actual instances of cyberbullying.

The confusion matrix shows that the model has a reasonable mix of sensitivity (ability to properly identify positives) and specificity (ability to accurately identify negatives) as illustrated in Fig. (6).

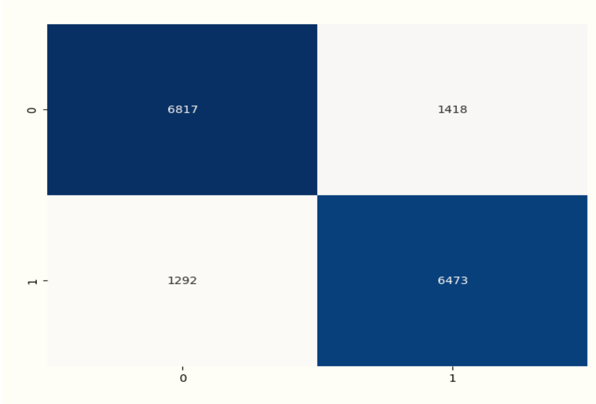


Fig. 3: Confusion matrix for CNN and meta-model in experiment 1

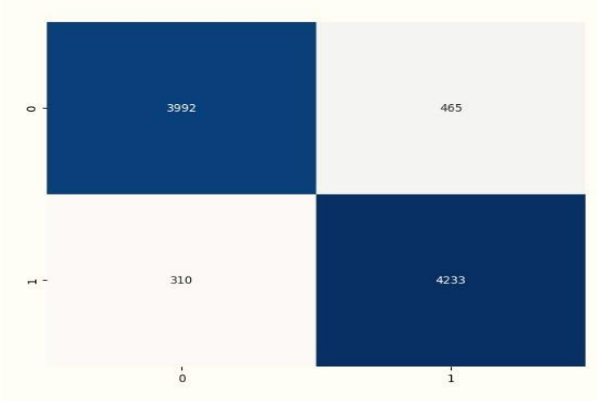


Fig. 4: Confusion matrix for CNN and meta-model in experiment 2

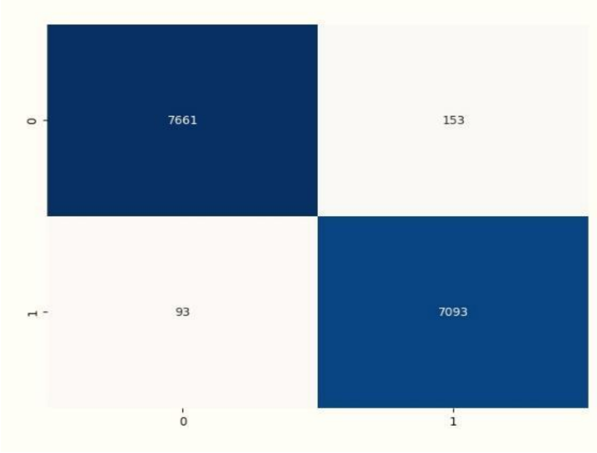


Fig. 5: Confusion matrix for CNN and meta-model in experiment 3

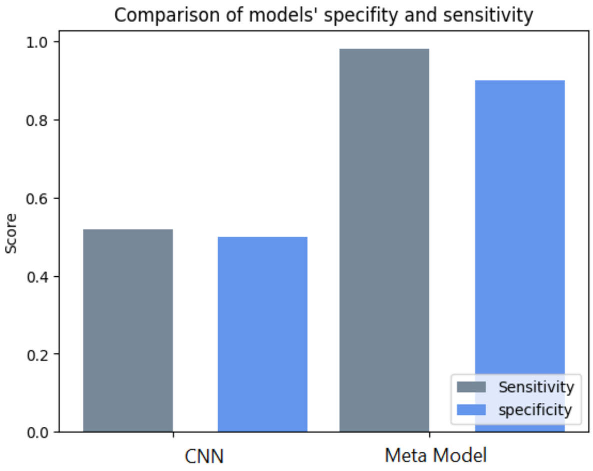


Fig. 6: Sensitivity and Specificity for CNN and (CNN, Meta) model *for the third experiment*

Among the performance measures are also the significant learning curves of the meta-learning model that are trained to detect cyberbullying in the Arabic language. In the accuracy learning curve, it is observed that the training accuracy rises sharply in the initial epochs, indicating a fast learning rate and then plateaus, maintaining a high level of accuracy throughout the remaining epochs. The validation accuracy, after a large initial increase, begins to deviate from the training accuracy, indicating that the model may have begun to outperform the training data.

The learning loss curve in Fig. (7) shows a rapid decrease in training loss, which is an encouraging indicator of the model's ability to reduce error on the training set. However, the validation loss does not follow the same trend. This is a prevalent problem in machine learning since the model learns effectively from the training data, which includes noise and imprecision, resulting in worse performance on the validation set.

According to Table (9), the obtained results for the proposed model regarding precision, recall, F1-score and accuracy for CNN and CNN + Meta-learning for cyberbullying for the two class classifications. For cyberbullying and not cyberbullying, the numbers represent the same performance pattern. As those measured. In terms of accuracy, CNN + Meta-learning achieved a higher performance of 84% in terms of accuracy during the first experiment.

Table (10) shows the results of the second experiment, as mentioned previously, for CNN and CNN + Meta-learning.

The results of the third experiment are shown in the Table (11).

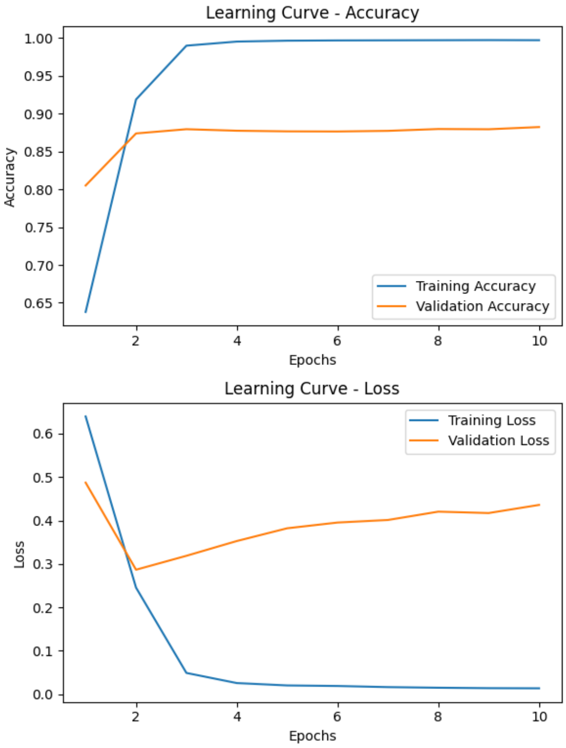


Fig. 7: Accuracy and loss curves

Table 9: First experiment results

Model	Categories	Precision (%)	Recall (%)	F1-Score (%)	Accuracy (%)
CNN	CB	69	81	74	71
	CB_Not	76	62	68	
CNN + meta-learning	CB	83	80	83	84
	CB_Not	80	85	83	

Table 10: Second experiment results

Model	Categories	Precision (%)	Recall (%)	F1-Score (%)	Accuracy (%)
CNN	CB	65	80	72	69
	CB_Not	76	59	66	
CNN + Meta-Learning	CB	93	90	91	91
	CB_Not	90	93	92	

Table 11: Third experiment results

Model	Categories	Precision (%)	Recall (%)	F1-Score (%)	Accuracy (%)
CNN	CB	51	59	68	52
	CB_Not	50	60	66	
CNN + meta-learning	CB	97	98	98	98.6
	CB_Not	98	96	97	

It is clear from these results that the CNN + Meta-learning model outperformed CNN alone in the three experiments on different datasets across all evaluation metrics. Specifically, the model achieved the highest accuracy of 98%. The results demonstrate the effectiveness of the approach and that the combination of the CNN and Meta-learning model was a clear improvement and better performance. Inaccurately detecting cyberbullying in Arabic text, even in the presence of dialectal differences and informal language.

Key Notes

Meta-learning: The work is based on the task of searching for Arabic comment classification to detect cyberbullying. The dataset was split into ten training and test tasks, where each of these tasks contains 10 training and testing subsets. Each training task is matched by a testing task. It acts as a proxy for real-world application where the model must encounter and interpret previously unseen data.

Feature extraction: The CNN layers successfully captured local features such as n-grams, frequent offensive phrases and patterns that indicate cyberbullying across dialects. The ability of these layers to identify features is crucial for detecting cyberbullying.

Dialectal differences: One of the main issues in Arabic NLP is dealing with the wide range of dialects spoken in different regions. The model's capacity to generalize across multiple dialects was evaluated by including different data from different countries where we used colloquial Arabic and Libyan dialects.

Conclusion

Cyberbullying is becoming more common in the Arab world as more people utilize social media. Deep learning models, such as convolutional neural networks and recurrent neural networks, have shown success. There is an urgent need for solutions and more studies to detect cyberbullying. This study has provided a method to identify cyberbullying, using a primary dataset in the Libyan dialect that includes about 7,500 thousand comments to determine the efficiency of the suggested approach.

However, creating these models requires a large amount of disaggregated data, which may be difficult to obtain for cyberbullying of Arabic. Accurate model creation is further challenged by the cultural and linguistic characteristics of Arabic script. It often takes a significant quantity of data to produce many low-shot tasks for meta-training, which is impractical in real-world low-shot scenarios. The training and testing for each task were combined, for a few snapshots and divided the data into groups. Each training task was matched by a test task to bridge the gap between the pre-training tasks and the testing phase tasks.

The findings of this research were encouraging and demonstrated the usefulness of the proposed model for the task of detecting cyberbullying. The CNN gives results with lower accuracy than the CNN + Meta-learning model, which achieved balanced and adequate accuracy and has somewhat better reliability than the CNN.

Future Research Directions

There may be a number of opportunities for future work to expand the research in the area of cyberbullying classification and detection using the CNN model and develop it to discover ways to increase accuracy.

Using a larger dataset to cover a wider range of features. Expand the scope of data collection to include other topics. To improve the results of the bullying detection task beyond binary classification by adding multiple labels. Create Stemmer pre-processing tools specifically for the Libyan dialect. Additionally, support for images and analysis of other dialects could also be added. We want to investigate the boundaries of integrating multilingual models with insights from CNN and Meta.

Acknowledgment

The authors are grateful to all the anonymous reviewers for their valuable remarks and suggestions; to the editors of this study for their prompt efforts in managing the manuscript that have considerably helped to enhance the original submission.

Funding Information

No funding was received for this study.

Author’s Contributions

Sarah Mansour Elgaud: Conduct the experiments and writing the report.

Mustafa Ali Abuzaraida: Writing introduction section, improve the writing and supervise the project.

Abdullah Alshehab: Finalize the report with contributing checking the process of the system.

Ethics

This article is original and contains unpublished material. The corresponding author confirms that all of the other authors have read and approved the manuscript and no ethical issues involved.

Conflict of Interest

The authors declare that there is no conflict of interest.

References

Ahmad Al-Khasawneh, M., Faheem, M., Abdulsalam Alarood, A., Habibullah, S., & Alsolami, E. (2024). Toward Multi-Modal Approach for Identification and Detection of Cyberbullying in Social Networks. *IEEE Access*, *12*, 90158–90170. https://doi.org/10.1109/access.2024.3420131

Alakrot, A., Murray, L., & Nikolov, N. S. (2018). Dataset Construction for the Detection of Anti-Social Behaviour in Online Communication in Arabic. *Procedia Computer Science*, *142*, 174–181. https://doi.org/10.1016/j.procs.2018.10.473

Alammary, A. S. (2021). Arabic Questions Classification Using Modified TF-IDF. *IEEE Access*, *9*, 95109–95122. https://doi.org/10.1109/access.2021.3094115

Alduailaj, A. M., & Belghith, A. (2023). Detecting Arabic Cyberbullying Tweets Using Machine Learning. *Machine Learning and Knowledge Extraction*, *5*(1), 29–42. https://doi.org/10.3390/make5010003

AlKhamissi, B., & Diab, M. (2022). Meta AI at Arabic Hate Speech 2022: MultiTask Learning with Self-Correction for Hate Speech Classification. *ArXiv:2205.07960*. https://doi.org/10.48550/arXiv.2205.07960

Alkhatib, M., Faisal, A., Alfalasi, F., Shaalan, K., & Mohmed, A. (2024). Deep Learning Approaches for Detecting Arabic Cyberbullying Social Media. *Procedia Computer Science*, *244*, 278–286. https://doi.org/10.1016/j.procs.2024.10.201

Anezi, F. Y. A. (2022). Arabic Hate Speech Detection Using Deep Recurrent Neural Networks. *Applied Sciences*, *12*(12), 6010. https://doi.org/10.3390/app12126010

Baiganova, A., Toxanova, S., Yerekesheva, M., Nauryzova, N., Zhumagalieva, Z., & Tulendi, A. (2024). Hybrid Convolutional Recurrent Neural Network for Cyberbullying Detection on Textual Data. *International Journal of Advanced Computer Science and Applications*, *15*(5). https://doi.org/10.14569/ijacsa.2024.0150584

Daraghmi, E.-Y., Qadan, S., Daraghmi, Y.-A., Yousuf, R., Cheikhrouhou, O., & Baz, M. (2024). From Text to Insight: An Integrated CNN-BiLSTM-GRU Model for Arabic Cyberbullying Detection. *IEEE Access*, *12*, 103504–103519. https://doi.org/10.1109/access.2024.3431939

Dewani, A., Memon, M. A., & Bhatti, S. (2021). Cyberbullying detection: advanced preprocessing techniques & deep learning architecture for Roman Urdu data. *Journal of Big Data*, *8*(1), 160. https://doi.org/10.1186/s40537-021-00550-7

Dien, T. T., Loc, B. H., & Thai-Nghe, N. (2019). Article Classification using Natural Language Processing and Machine Learning. *2019 International Conference on Advanced Computing and Applications (ACOMP)*, 78–84. https://doi.org/10.1109/acomp.2019.00019

Enamoto, L., Weigang, L., & Filho, G. P. R. (2021). A generic framework for multilingual short text categorization using convolutional neural network. *Multimedia Tools and Applications*, *80*(9), 13475–13490. https://doi.org/10.1007/s11042-020-10314-9

Ganfure, G. O. (2022). Comparative analysis of deep learning based Afaan Oromo hate speech detection. *Journal of Big Data*, *9*(1), 76. https://doi.org/10.1186/s40537-022-00628-w

Ghosh, R., Nowal, S., & Manju, G. (2021). Social media cyberbullying detection using machine learning in bengali language. *International Journal of Engineering Research & Technology*, *10*(5), 190–193.

Nikitha, G., Shenoyy, A., Chaturya, K., Latha, J., & Janani, S. M. (2024). Detection of Cyberbullying Using NLP and Machine Learning in Social Networks for Bi-Language. *International Journal of Scientific Research & Engineering Trends*, *10*(1), 153–161.

Habberrih, A., & Abuzaraida, M. A. (2024a). Sentiment Analysis of Arabic Dialects: A Review Study. *Computing and Informatics*, 137–153. https://doi.org/10.1007/978-981-99-9589-9_11

Habberrih, A., & Abuzaraida, M. A. (2024b). Sentiment Analysis of Libyan Middle Region Using Machine Learning with TF-IDF and N-grams. *Information and Communications Technologies*, 197–209. https://doi.org/10.1007/978-3-031-62624-1_16

Habberrih, A., & Ali Abuzaraida, M. (2024). Sentiment Analysis of Libyan Dialect Using Machine Learning with Stemming and Stop-words Removal. *5th International Conference on Communication Engineering and Computer Science (CIC-COCOS'24)*, 259–264. https://doi.org/10.24086/cocos2024/paper.1171

Haidar, B., Chamoun, M., & Serhrouchni, A. (2017). A Multilingual System for Cyberbullying Detection: Arabic Content Detection using Machine Learning. *Advances in Science, Technology and Engineering Systems Journal*, *2*(6), 275–284. https://doi.org/10.25046/aj020634

Haidar, B., Chamoun, M., & Serhrouchni, A. (2018). Arabic Cyberbullying Detection: Using Deep Learning. *2018 7th International Conference on Computer and Communication Engineering (ICCCE)*, 284–289. https://doi.org/10.1109/iccce.2018.8539303

Husain, F. (2020). Arabic Offensive Language Detection Using Machine Learning and Ensemble Machine Learning Approaches. *ArXiv:2005.08946*. https://doi.org/10.48550/arXiv.2005.08946

Jarrar, M., Zaraket, F. A., Hammouda, T., Alavi, D. M., & Wählisch, M. (2023). Lisan: Yemeni, Iraqi, Libyan, and Sudanese Arabic Dialect Corpora with Morphological Annotations. *2023 20th ACS/IEEE International Conference on Computer Systems and Applications (AICCSA)*, 1–7. https://doi.org/10.1109/aiccsa59173.2023.10479250

Kanan, T., Aldaaja, A., & Hawashin, B. (2020). Cyber-bullying and cyber-harassment detection using supervised machine-learning techniques in Arabic social media contents. *Journal of Internet Technology*, *21*(5), 1409–1421. https://doi.org/0.3966/160792642020092105016

Karayiğit, H., İnan Acı, Ç., & Akdağlı, A. (2021). Detecting abusive Instagram comments in Turkish using convolutional Neural network and machine learning methods. *Expert Systems with Applications*, *174*, 114802. https://doi.org/10.1016/j.eswa.2021.114802

Khairy, M., Mahmoud, T. M., Omar, A., & Abd El-Hafeez, T. (2024). Comparative performance of ensemble machine learning for Arabic cyberbullying and offensive language detection. *Language Resources and Evaluation*, *58*(2), 695–712. https://doi.org/10.1007/s10579-023-09683-y

Li, S., & Gong, B. (2021). Word embedding and text classification based on deep learning methods. *MATEC Web of Conferences*, *336*, 06022. https://doi.org/10.1051/matecconf/202133606022

Mazari, A. C., & Kheddar, H. (2023). Deep Learning-based Analysis of Algerian Dialect Dataset Targeted Hate Speech, Offensive Language and Cyberbullying. *International Journal of Computing and Digital Systems*, *13*(1), 965–972. https://doi.org/10.12785/ijcds/130177

Mubarak, H. (2019). Build fast and accurate lemmatization for Arabic. *ArXiv:1710.06700*. https://doi.org/10.48550/arXiv.1710.06700

Mubarak, H., Darwish, K., & Magdy, W. (2017). Abusive Language Detection on Arabic Social Media. *Proceedings of the First Workshop on Abusive Language Online*, 52–56. https://doi.org/10.18653/v1/w17-3008

Musleh, D., Rahman, A., Alkherallah, M. A., Al-Bohassan, M. K., Alawami, M. M., Alsebaa, H. A., Alnemer, J. A., Al-Mutairi, G. F., Aldossary, M. I., Aldowaihi, D. A., & Alhaidari, F. (2024). A Machine Learning Approach to Cyberbullying Detection in Arabic Tweets. *Computers, Materials & Continua*, *80*(1), 1033–1054. https://doi.org/10.32604/cmc.2024.048003

Omar, A., Essgaer, M., & Ahmed, K. M. S. (2022). Using Machine Learning Model To Predict Libyan Telecom Company Customer Satisfaction. *2022 International Conference on Engineering & MIS (ICEMIS)*, 1–6. https://doi.org/10.1109/icemis56295.2022.9914055

Perera, A., & Fernando, P. (2024). Cyberbullying Detection System on Social Media Using Supervised Machine Learning. *Procedia Computer Science*, *239*, 506–516. https://doi.org/10.1016/j.procs.2024.06.200

Rachid, B. A., Azza, H., & Ben Ghezala, H. H. (2020). Classification of Cyberbullying Text in Arabic. *2020 International Joint Conference on Neural Networks (IJCNN)*, 1–7. https://doi.org/10.1109/ijcnn48605.2020.9206643

Salama, R. A., Youssef, A., & Fahmy, A. (2018). Morphological Word Embedding for Arabic. *Procedia Computer Science*, *142*, 83–93. https://doi.org/10.1016/j.procs.2018.10.463

Shaker, N. H., & Dhannoon, B. N. (2024). Word embedding for detecting cyberbullying based on recurrent neural networks. *IAES International Journal of Artificial Intelligence (IJ-AI)*, *13*(1), 500. https://doi.org/10.11591/ijai.v13.i1.pp500-508

Xingyi, G., & Adnan, H. M. (2024). Potential cyberbullying detection in social media platforms based on a multi-task learning framework. *International Journal of Data and Network Science*, *8*(1), 25–34. https://doi.org/10.5267/j.ijdns.2023.10.021

Xu, J., & Du, Q. (2020). Learning transferable features in meta-learning for few-shot text classification. *Pattern Recognition Letters*, *135*, 271–278. https://doi.org/10.1016/j.patrec.2020.05.007

Zeroual, I., & Lakhouaja, A. (2017). Arabic information retrieval: Stemming or lemmatization? *2017 Intelligent Systems and Computer Vision (ISCV)*, 1–6. https://doi.org/10.1109/isacv.2017.8054932