Comparative Study of Machine Learning Models for the Detection of Abusive Messages: Case of Wolof-French Codes Mixing Data

Ibrahima Ndao[1(✉)], Khadim Dramé[1], Gorgoumack Sambe[1], and Gayo Diallo[2]

[1] Laboratoire d'Informatique et d'Ingénierie pour l'Innovation, Université Assane Seck de Ziguinchor, Diabir, Ziguinchor, Sénégal
i.ndao20150570@zig.univ.sn, {khadim.drame,gsambe}@univ-zig.sn
[2] AHead, Bordeaux Population Health - INSERM 1219 and LABRI, University, Bordeaux, France
Gayo.Diallo@u-bordeaux.fr

Abstract. This paper presents a comparative study of machine learning models for detecting abusive messages, focusing on code-mixed data in Wolof and French languages. With the increasing use of digital platforms, there has been a surge in derogatory comments, necessitating effective detection strategies. The study introduces a meticulously annotated dataset of 2022 Twitter tweets, manually classified as abusive or not. Extensive experiments are conducted with various machine learning algorithms, including deep learning, with a focus on comparing their performance on the test dataset.

Keywords: abusive messages · hate messages · code mixing · machine learning · deep learning · language models · low-resource languages

1 Introduction

The number of digital platform users has significantly increased [1]. The exponential increase, combined with the unregulated nature of social media usage, has facilitated the widespread spread of abusive messages. Abusive language encompasses a range of expressions that include excessive, false, exaggerated, or attacking communications directed at individuals or groups based on attributes such as race, ethnicity, or sexual orientation [2]. The statement "*i aint never worried bout no nigga*" can be considered a racist expression. Categorizing messages into distinct forms of abuse presents a classification obstacle when it comes to identifying abusive language [3]. As a result, these discourses that promote antisocial behavior require substantial actions from governments, companies, and other organizations to develop efficient strategies to counteract them [4]. Several methods have been suggested to control such behaviors [5, 6]. Nevertheless, users are progressively utilizing evasion strategies such as message camouflage, abbreviations, phonetic input, and code mixing, which make manual analysis and moderation more challenging, especially considering the immense volume of information

K. Cheikh M. F. Kebe et al. (Eds.): InterSol 2024, LNICST 610, pp. 252–263, 2025.
https://doi.org/10.1007/978-3-031-86493-3_20

being shared on social media platforms. Therefore, there is a pressing need for the automated identification of offensive messages.

While there has been a significant amount of research conducted on this matter for languages that have abundant resources, such as English and French, there has been relatively little effort dedicated to languages that have limited resources. Furthermore, there has been a lack of focus on identifying abusive messages in code-mixed data.

In order to fill this gap, we introduce the first annotated collection of Wolof-French code-mixed texts, which has been specifically created for the purpose of identifying abusive messages. Following that, we proceed with a sequence of experiments utilizing various machine learning (ML), deep learning (DL), and language models. The results of these experiments provide insight into the most efficient models within this distinct linguistic and contextual environment.

The following sections of this paper are structured as follows: Sect. 2 explores previous research on identifying offensive messages. Section 3 offers a comprehensive understanding of the corpus's construction and annotation process. Section 4 provides a comprehensive account of the experiments carried out using different models on our annotated corpus and analyzes their outcomes.

2 Related Work

In this section, we present related work on the detection of abusive messages which can be classified into three approaches: linguistic approach, ML-based, and DL-based approaches. Additionally, research conducted on low-resource languages and code-mixing is discussed.

2.1 Linguistic Approach

Linguistic approaches exploit different manually defined features. These include word lexicons, dictionaries, and so one. These features can be related to the number of offensive words used, hashtags, personal pronouns used, word distance metrics. In [7], the authors focus on users who openly display their hateful emotions in tweets using the sentence structure: "I < intensity > < user's intention > < target of hate > ". In [8], the authors address the subjective aspect of tweets and construct a word lexicon to perform a classification of hate into three distinct categories (highly hateful, mildly hateful, or non-hateful).

Although the results of these methods are satisfactory, they faced with manual dependency (definitions from dictionaries, lexicons, etc.). In addition, their performance is limited when faced with low-quality texts, such as spelling mistakes, phonetic input, etc. [5]. Furthermore, the use of certain terms (e.g., "nigger") may not be racist, requiring a contextual analysis of the content. Research conducted using this approach has focused on a specific type of abusive speech (hate [7, 8], racism [5], Offensive [3], sexist [9], etc.), a specific platform (twitter [2], etc.), or a particular language (english [3], arabic [2], etc.) due to its inability to be generalized.

2.2 Machine Learning-Based Approach

Several machine learning algorithms were explored in order to classify abusive messages: Support Vector Machine (SVM), Naive Bayes (NB), Logistic Regression (LR), Random Forest (RF) and so one. These classifiers were used with various features: bag of words (BOW), word or character n-grams, TF-IDF, etc. In [5], for example, the authors trained NB classifier with BOW features to perform a binary classification (racist or non-racist). In [9], the authors focused on sexist messages using SVM to classify tweets as hostile, benevolent, or other. The detection of anti-migrant discourses was studied in [10], using different classifiers with various feature extractors. Their model using word n-grams achieved the highest score.

Hate speech has also been studied, particularly in [11], where hierarchical regression models are used. The authors determine the amount of hate speech associated with a person through personal characteristics such as party affiliation, gender, or ethnic origin. In [12], the authors used patterns and unigrams as input features with several classifiers (NB, KNN, SVM, RF) to perform binary and ternary classification on a test dataset of 2010 tweets. Their best model achieved an accuracy of 87.4% for binary classification (offensive or not) and an accuracy of 78.4% for ternary classification (hateful, offensive, or clean). In [13], the authors experimented several classifiers with different features such as 3 to 5-g, unigrams, bigrams, linguistic features (average word length, punctuation count, comment length, etc.), and syntactic features to identify online hate content. They showed that models using n-grams features yielded good results, but their combinations with text extensions were more performant.

These different propositions allowed improving classification performance of tweets across a wide range of abusive message types. The supervised approach focuses on a set of features that will be used by machine learning algorithms. However, the different types of abusive messages in tweets lack discriminative features. Even though these studies obtained promising results, we noted degradation of their performance in other cases of abusive remarks.

2.3 Deep Learning-Based Approach

The use of deep learning approaches is justified by their ability to learn new feature representations from input data. The input data can be raw data or feature embeddings. Thus, the different operations performed on the stacked layers of these deep neural networks allow the classification of tweets, and the results show that they are more effective for this task. Among the studies in this approach, Chiril [14] explored in his doctorate thesis deep learning models such as BERT, FlauBERT, CNN, and Bi-LSTM with attention in the automatic detection of abusive messages, particularly for sexism. In [15], the authors introduced the treatment of polysemy, syntax, semantics, out-of-vocabulary words, as well as sentiment information combined into an input vector to a neural model (Bi-LSTM) to detect hateful messages and abusive language on Twitter. The authors in [16] addressed the problem under a more general approach by proposing a unified method for classifying tweets into different categories (hate speech, sexism, racism, bullying, sarcasm). They proposed DL-based model that allowed the identification of these different categories of abusive messages without requiring model tuning for each case. In [4],

the authors explored the use of lexical extensions (word2vec, GloVe, ELMo,) and graph extensions (neural networks) for the detection of abusive messages. The evaluation of their models showed a clear improvement of performance when combining lexical and graphical extensions.

Other works combined ML-based and DL-based models. In [17], a set of features were used with SVM, CNN, and Multi-layer Perceptron (MLP) to perform binary classification (hateful or non-hateful). The authors in [18] focused on the treatment of hateful metaphors as features to identify hate speech and their targets in Dutch comments on Facebook. Evaluation of SVM, BERT, and RoBERTa models shows that the features of hateful metaphors increase the classification performance of hate speech. In [19], the authors collected a set of 197,566 comments from different platforms (YouTube, Reddit, Wikipedia, Twitter) and used several feature extractors (BOW, TF-IDF, word2vec, BERT) with classifiers (LR, NB, SVM, XGBoost, NN) for hate detection. Text extensions provided by BERT had a greater effect on the classification of tweets. In addition, the XGBoost model achieved the best F-measure of 0.92.

DL-based approaches improved the performance of the state of art in detecting abusive messages. However, despite the diversity of abusive message cases (hate, offense, cyberbullying, trolling, misinformation, etc.), most of existing works focus on one case by conducting binary classification (e.g., hateful or non-hateful). Furthermore, the majority focus on one rich-resource language (English, Arabic). Thus, detecting abusive messages is challenging, particularly in low-resource languages.

2.4 Low-Resource Languages and Code Mixing

Over the past few decades, many studies have addressed the detection of hate speech in low-resource languages. These languages are characterized by a scarcity or limited availability of high-quality annotated dataset [20]. In certain languages, there is the practice of incorporating characters or words from Latin derivatives through borrowing. This phenomenon is called code mixing. Several works have addressed these languages and linguistic phenomena. In [21], the authors proposed new method for detecting hate messages in Hindi-English code-mixing data. Their method involves using word embeddings with FastText to feed SVM and radial basis function (RBF) models. Their results showed that FastText produces much better representations than word2vec and doc2vec. In [22], the authors reported a comparative study of different transformer architectures on Hindi-English code-mixed texts for sentiment analysis, emotion recognition, and hate speech identification. The results of code-mixed models (HingBERT, HingRoBERTa, HingRoBERTa-Mixed, mBERT) were compared to models without code mixing (ALBERT, BERT, and RoBERTa). This study revealed notable performances of the HingBERT model and very low performance of the BERT model.

Other works focused specifically on certain dialects or low-resource languages without studying code-mixing. In [23], the authors proposed DL-based approach for detecting hate speech in Algerian dialect written in Arabic. This study was conducted on a corpus of 135,000 tweets annotated into two classes (hateful and non-hateful). The authors in [24] studied hate speech against women on YouTube. A corpus annotated by three annotators is used to train CNN, LSTM, and Bi-LSTM models. The CNN model achieved the best F-measure of 0.86. In [25], the authors studied offensive and abusive speech in

Facebook comments written in Algerian dialect in Arabic. Bi-LSTM, CNN, FastText, SVM, and NB models were used on a corpus of 8,700 tweets annotated as normal, abusive, and offensive.

These studies represent major advancements for these languages. They have led to the creation of annotated corpora for hate speech detection and the proposal of models with promising performance. However, this aspect remains to be studied in many other low-resource languages. This is the case for Senegalese comments where code-mixing is prevalent (English/French + Wolof). To our knowledge, Wolof, which is spoken by nearly 90% of the population, does not have annotated textual data for abusive messages detection [26].

In the following section, we present a dataset of Wolof/French code-mixing collected from Twitter and the annotation process of this dataset.

3 Data Collection and Annotation

The construction of high-quality dataset is a challenging and time-consuming task, especially for the unofficial languages like Wolof. Most users of this language do not strictly follow spelling and grammatical rules. In addition, French-Wolof mixing codes is widely used in social medias. This makes the available texts very heterogeneous and difficult to exploit.

In this work, we used the twint[1], an advanced Python scraping library, to collect tweets from Twitter. Twint allows scraping tweets from Twitter without using the Twitter API, which has limitations (3200 tweets per request for example).

A collection of 144,225 tweets from January 1, 2021 to May 31, 2023 was extracted. The extraction of these tweets includes keywords (e.g. World Cup, racist, politician, etc.), location or proximity (e.g. Senegal, Qatar, Cameroon, Morocco, Ghana, Algeria, Tunisia, Africa; etc.), person (e.g. Aliou cisse, Macky Sall, Ousmane Sonko, etc.), language (e.g. French) and so on. The queries launched for data collection enable coverage of tweets related to various domains, people, localities and over a specific time period. They also help to resolve class imbalance issues. The raw data collected contains both monolingual and multilingual data. Identifying the language in the messages is an integral part of our annotation process. Thus, we only consider tweets with Wolof-French code mixes. Since, to our knowledge, there are no resources available for abusive message detection on Wolof-French code-mix data, we are striving to produce one of the first coarse-grained datasets for abusive speech in Wolof-French. The annotation process includes pre-processing steps, such as the removal of emojis, URLs, hashtags and so on. The deletion of these entities is actually due to their lack of relevance to the analysis of the code-mix aspect under study.

Due to the lack of human resources for annotation, we had to annotate the corpus ourselves. As we are native speakers of Wolof and have a background in French, we see ourselves as endowed with the ability to understand both languages. In addition to the subjectivity of abusive message detection, we drew on the definition in [3] "Abusive language includes all excessive, false or attacking communications towards a person

[1] https://github.com/twintproject/twint.

or group of people on the basis of characteristics such as their race, ethnicity, sexual orientation, etc.“ to annotate the corpus.

Thus, we annotated 2022 tweets manually for the detection of abusive messages. The annotation concerns two (2) classes: class 0 for non-abusive messages and class 1 for abusive messages. The corpus consists of 1069 tweets from class 0 and 953 tweets from class 1. Table 1 shows examples of annotated messages from our Wolof-French code-mix corpus. Text highlighted in red corresponds to words in Wolof and text in blue corresponds to words in French.

Table 1. Examples of messages from the annotated corpus

Label	Tweets	Tweets translated into English
1	La manifestation de la société civile: « Sunuy milliards du reesss »	Civil society protest: "Our billions will not be tolerated".
0	Sinon concert casserole bi tay degouma dara deh wala sama site bokoul ci senegal	Otherwise I haven't heard anything today about the saucepan concert, or maybe my neighbourhood isn't part of Senegal.
1	Mon cerveau a bug en entendant cette phrase. Xamna daf am benen Senegal bou outek bini guiss. Sacré keur	My brain bugged when I heard this sentence. There is certainly another senegal different from the one we see in Sacré Keur.
0	Maky deh dafa yakar ni senegal new-york leu . Discours bi on dirait deuk bi lep nice alors que non	Macky thinks Senegal is like the United States. His speech sounds like everything is impeccable, but it's not.

This resulting annotated dataset is then used to conduct experiments with different ML, DL and language models.

4 Experiments

In this section, we present the evaluation of the different models on our annotated dataset and compare their results. The first subsection present experiments with ML algorithms while the second report experiments with DL algorithms. The last subsection provides the results obtained by different language models. In each of the experiments, 70% of the dataset is used for training and 30% for evaluating the models. Precision, Recall, and F-measure as well as accuracy are used as evaluation metrics.

Machine Learning (ML) Algorithms. Seven ML algorithms were used: SVM, KNN, Decision Tree (DT), NB, RF, LR, and the Multi-Layer Perceptron (MLP). In addition, we experimented five (5) boosting algorithms (Cat Boost, LigthGBM, XGBoost, AdaBoost and Gradient Booster) as well as three (3) voting ensemble models (hard vote, soft vote and weighted vote). In the different experiments, we used four vectorization tools: TF-IDF, BOW, 2-g and word2vec. Table 2 presents the results of the top five algorithms,

ranked in order (based on their accuracy) with the previously mentioned vectorization tools.

Table 2. Results of the top five ML algorithms according to the vectorization

Evaluation Measures		Precision		Recall		F-measure		Accuracy
Feature Extractors	Algorithms	0	1	0	1	0	1	X
TF-IDF	Naive Bayes	0.68	0.72	0.79	0.59	0.73	0.65	0.70
	Logistic Regression	0.67	0.71	0.78	0.59	0.72	0.64	0.69
	SVM	0.68	0.68	0.74	0.62	0.71	0.65	0.68
	MLP	0.67	0.68	0.73	0.61	0.70	0.64	0.67
	Random Forest	0.65	0.70	0.78	0.54	0.71	0.61	0.67
Bag Of Words	MLP	0.68	0.73	0.80	0.59	0.74	0.66	0.70
	Naive Bayes	0.70	0.68	0.71	0.68	0.70	0.68	0.69
	SVM	0.67	0.71	0.78	0.58	0.72	0.64	0.69
	Logistic Regression	0.66	0.72	0.81	0.55	0.72	0.62	0.68
	Random Forest	0.65	0.75	0.85	0.49	0.73	0.60	0.68
Ngram 2	Logistic Regression	0.62	0.83	0.93	0.38	0.74	0.52	0.66
	Naive Bayes	0.64	0.68	0.78	0.52	0.70	0.59	0.65
	SVM	0.59	0.75	0.90	0.34	0.71	0.46	0.63
	MLP	0.59	0.74	0.90	0.31	0.71	0.44	0.62
	XGBoost	0.58	0.82	0.95	0.26	0.72	0.39	0.62
Word2vec	Naive Bayes	0.69	0.68	0.71	0.65	0.70	0.66	0.68
	Logistic Regression	0.66	0.72	0.81	0.54	0.73	0.62	0.68
	CatBoost	0.65	0.77	0.86	0.49	0.74	0.60	0.68
	SVM	0.65	0.69	0.76	0.56	0.70	0.62	0.67
	Random Forest	0.64	0.72	0.82	0.50	0.72	0.59	0.67

The top five models yielded good results with accuracies ranging from 0.63 to 0.70. The results obtained with TF-IDF, BOW, and Word2vec features are quite similar while with the 2-g feature (2 words) the results are much lower in accuracy but higher according to the precision. This can be explained by the fact that the words that determine the abuse

are mostly represented in multiple words (bi-grams). Overall, the NB, LR, and SVM algorithms got the best performances with 0.70, 0.69, 0.65, and 0.68 for NB, 0.69, 0.68, 0.66, and 0.68 for LR, and 0.68, 0.69, 0.63, and 0.67 for SVM respectively with the TF-IDF, BOW, 2-g, and Word2vec feature extractors.

Deep Learning (DL) Algorithms. We present the results obtained on different combinations of DL algorithms with features extracted with TF-IDF. First, eight (8) combinations of DL algorithms were explored: CNN (Convolutional Neural Network), RNN (Recurrent Neural Network), GRU (Gated Recurrent Unit), Seq2seq, LSTM (Long Short-Term Memory) with 1 layer, LSTM with 1 layer + dropout, LSTM with 2 layers + dropout, and Bi-LSTM. Then, attention layers were added to these models. Table 3 presents the results obtained by the DL models.

Table 3. Results of DL models

Evaluation Measures		Precision		Recall		F-measure		Accuracy
Attention	Algorithms	0	1	0	1	0	1	X
Without attention	Bi-LSTM	0.68	0.70	0.75	0.62	0.71	0.66	0.69
	LSTM + dropout	0.67	0.71	0.78	0.57	0.72	0.63	0.68
	LSTM with 2 layers + dropout	0.67	0.68	0.74	0.60	0.71	0.64	0.68
	Seq2seq	0.68	0.67	0.72	0.62	0.70	0.65	0.67
	CNN	0.65	0.69	0.78	0.55	0.71	0.61	0.67
	GRU	0.66	0.64	0.69	0.61	0.67	0.63	0.65
	LSTM with 1 layer	0.61	0.59	0.55	0.63	0.57	0.60	0.60
	RNN	0.54	0.50	0.55	0.49	0.54	0.49	0.52
With attention	LSTM + dropout	0.69	0.73	0.79	0.61	0.74	0.66	0.70
	LSTM with 2 layers + dropout	0.63	0.70	0.81	0.49	0.71	0.58	0.66
	CNN	0.66	0.75	0.83	0.54	0.74	0.62	0.69
	GRU	0.68	0.72	0.78	0.60	0.73	0.60	0.69
	LSTM with 1 layer	0.65	0.68	0.76	0.56	0.70	0.61	0.66
	RNN	0.60	0.80	0.92	0.33	0.73	0.47	0.64

Experiments on deep learning algorithms showed that CNN (0.67) and seq2seq (0.67) algorithms outperform the RNN (0.52) algorithm as well as its variants GRU (0.65) and LSTM (0.60). However, adding a dropout layer (0.68) or an additional layer to the LSTM model allows to improve its performance (0.68) compared to CNN and Seq2seq models. The extension of LSTM, Bi-LSTM, achieved the best results (0.69 of accuracy).

Adding attention layers increases the performance of each model. Thus, the LSTM + dropout model gains 2 more points in accuracy, the CNN gains 3 more points, the GRU gains 4 more points, the single-layer LSTM gains 6 more points, and the RNN model gains 12 more points.

Language Models. A series of experiments were conducted on five families of language models: multilingual models, monolingual models for English, monolingual models for French, monolingual models for Wolof and a bilingual French-Wolof model. All these models are available on https://huggingface.co. Table 4 presents the results obtained by language models in each category.

Table 4. Results of language models

Evaluation Measures		Precision		Recall		F-measure		Accuracy
Characteristic categories	Algorithms	0	1	0	1	0	1	X
Multilingual	bert-base-multilingual-uncased	0.80	0.49	0.46	0.82	0.58	0.61	0.66
	bert-base-multilingual-cased	0.74	0.50	0.55	0.69	0.63	0.58	0.62
	xlnet-base-cased	0.65	0.63	0.91	0.25	0.76	0.35	0.65
	Distilbert-base-multilingue-cased	0.74	0.58	0.72	0.61	0.73	0.60	0.68
	xlm-roberta-base	0.73	0.53	0.65	0.63	0.69	0.58	0.64
	google/electra-small-discriminator	0.73	0.51	0.60	0.66	0.66	0.57	0.62
English monolingual	roberta-base	0.87	0.41	0.13	0.97	0.22	0.58	0.45
	bert-base-cased	0.71	0.53	0.69	0.56	0.70	0.54	0.64
	bert-base-uncased	0.79	0.51	0.53	0.78	0.65	0.62	0.63
	albert-base-v2	0.72	0.51	0.62	0.62	0.66	0.56	0.62
	vinai/bertweet-base	0.78	0.44	0.28	0.88	0.41	0.58	0.51
	flaubert/flaubert_base_cased	0.76	0.54	0.63	0.68	0.69	0.61	0.65
	flaubert/flaubert_base_uncased	0.64	0.40	0.43	0.66	0.48	0.50	0.49
French monolingual	dbmdz/bert-base-french-europeana-cased	0.75	0.57	0.70	0.63	0.72	0.60	0.67
	dbmdz/electra-base-french-europeana-cased-discriminator	0.76	0.50	0.52	074	0.62	0.59	0.65
	dbmdz/electra-base-french-europeana-cased-generator	0.69	0.55	0.75	0.48	0.72	0.51	0.62
	claudelkros/bert-base-french	0.72	0.48	0.53	0.67	0.61	0.56	0.59
	geotrend/bert-base-fr-cased	0.74	0.48	0.49	0.73	0.59	0.58	0.58
	camembert-base	0.79	0.53	0.57	0.76	0.66	0.62	0.65
	abhilash1910/french-roberta	0.61	0.39	0.72	0.29	0.66	0.33	0.55
Monolingual Wolof	davlan/bert-base- multilingual-cased- finetuned-wolof	0.75	0.55	0.67	0.65	0.71	0.60	0.66
	abdouaziiz/ bert-base- wolof	0.80	0.52	0.54	0.79	0.64	0.63	0.63
	abdouaziiz/so raberta	0.69	0.51	0.67	0.54	0.68	0.52	0.62
Bilingual (Wolof / French)		0.73	0.53	0.65	0.63	0.69	0.58	0.64

The results obtained for class 0 (non-abusive) are very good, while those for class 1 (abusive) are very poor. However, the "Distilbert-base-multilingual-cased" model (multilingual) obtained the best f-measurement, i.e. 0.68. It was closely followed by the "dbmdz/bert-base-french-europeana-cased" model (monolingual French) with a fmeasure of 0.67, then the "bert-base-multilingual-uncased" model (multilingual) and the "davlan/bert-base- multilingual-cased- finetuned-wolof" model (monolingual Wolof)

with a f-measure of 0.66. While the best accuracies for class 1 (abusive) are obtained by multilingual models ("xlnet-base-cased" (0.63) and "Distilbert-basemultilingual-cased" (0.58)). Assuming that the results obtained are close and mixed as a function of the measures, we can note that multilingual models are more stable than other language models as a function of precision for class 1 and accuracy.

In summary, we noted that classical ML algorithms, such as NB, SVM and LR with TF-IDF or BOW features, achieve comparable results to certain DL models with attention (LSMT + dropout, CNN, GRU). However, their results remain significantly superior to other models, especially language models, particularly in class 1 (abusive). This can be explained by the fact that language models are trained on large quantities of data, which are mostly non-abusive. Whereas for deep learning algorithms, they require more data to learn more complex data representations.

5 Conclusion and Future Work

The need for effective automation in identifying abusive messages on social media requires a proactive strategy. This study examines the scope of social media abuse, including cyberbullying, offensive language, trolling, and hate speech, providing a summary of relevant research. We have created a carefully annotated dataset consisting of 2022 tweets, which have been categorized into two classes for the purpose of detecting abusive messages. By conducting numerous experiments using various models and feature extractors on our distinct mixed-code Wolof-French dataset, we have determined the most efficient methods within this particular context.

In the future, our efforts will focus on expanding the dataset and developing novel strategies to tackle an important question: When faced with limited resources and code mixing, it is crucial to prioritize specific aspects in order to effectively identify abusive messages. This ongoing investigation is positioned to not only improve our comprehension of identifying abusive content in various languages and limited-resource settings, but also to enhance the efficiency of automated systems in tackling this widespread societal problem.

References

1. Toujani, R.: Opinions Mining from Posters' Users in Social Networks (2021)
2. Mubarak, H., Darwish, K., et Magdy, W.: Abusive language detection on arabic social media. In: Proceedings of the First Workshop on Abusive Language Online, Vancouver, BC, Canada: Association for Computational Linguistics, août 2017, pp. 52–56 (2017). https://doi.org/10.18653/v1/W17-3008
3. Davidson, T., Warmsley, D., Macy, M., et Weber, I.: Automated Hate Speech Detection and the Problem of Offensive Language. arXiv, 11 mars 2017. Consulté le: 22 novembre 2022. [En ligne]. http://arxiv.org/abs/1703.04009
4. N. Cécillon, R. Dufour, et V. Labatut, « Approche multimodale par plongement de texte et de graphes pour la détection de messages abusifs », p. 26, 2021
5. Kwok, I., Wang, Y.: Locate the hate: detecting tweets against blacks. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 27, no. 1, pp. 1621–1622 (2013). https://doi.org/10.1609/aaai.v27i1.8539

6. Das, A., Gambäck, B.: Code-Mixing in Social Media Text, p. 24 (2013)
7. Silva, L., Mondal, M., Correa, D., Benevenuto, F., Weber, I.: Analyzing the targets of hate in online social media. In: Proceedings of the International AAAI Conference on Web and Social Media, vol. 10, no, 1, pp. 687–690 (2021). https://doi.org/10.1609/icwsm.v10i1.14811
8. Gitari, N.D., Zuping, Z., Damien, H., Long, J.: A Lexicon-based approach for hate speech detection. Int. J. Multimed. Ubiquitous Eng. **10**(4), 215–230 (2015). https://doi.org/10.14257/ijmue.2015.10.4.21
9. Jha, A., Mamidi, R.: When does a compliment become sexist? Analysis and classification of ambivalent sexism using twitter data. In: Proceedings of the Second Workshop on NLP and Computational Social Science, Vancouver, Canada: Association for Computational Linguistics, pp. 7–16 (2017). https://doi.org/10.18653/v1/W17-2902
10. Pitropakis, N., Kokot, K., Gkatzia, D., Ludwiniak, R., Mylonas, A., Kandias, M.: Monitoring users' behavior: anti-immigration speech detection on Twitter. Mach. Learn. Knowl. Extr. **2**(3), 192–215 (2020). https://doi.org/10.3390/make2030011
11. Solovev, K., Pröllochs, N.: Hate speech in the political discourse on social media: disparities across parties, gender, and ethnicity. Undefined (2022). https://doi.org/10.1145/3485447.3512261
12. Swamy, M.K., Jyothi, U.P.: An effective approach to hate speech detection on social media, vol. 08, no. 07 (2021)
13. Nobata, C., Tetreault, J., Thomas, A., Mehdad, Y., Chang, Y.: Abusive language detection in online user content. In: Proceedings of the 25th International Conference on World Wide Web, Montréal Québec Canada: International World Wide Web Conferences Steering Committee, avr. 2016, pp. 145–153 (2016). https://doi.org/10.1145/2872427.2883062
14. Chiril, P.: Automatic Hate Speech Detection on Social Media. Université Toulouse 3 - Paul Sabatier, 2022. Consulté le: 9 janvier 2024. [En ligne]. Disponible sur: https://theses.hal.science/tel-03599458
15. Naseem, U., Razzak, I., Hameed, I.A.: Deep context-aware embedding for abusive and hate speech detection on Twitter, vol. 15, no. 4, p. 8 (2019)
16. Founta, A.M., Chatzakou, D., Kourtellis, N., Blackburn, J., Vakali, A., Leontiadis, I.: A unified deep learning architecture for abuse detection. In: Proceedings of the 10th ACM Conference on Web Science, in WebSci 2019. New York, NY, USA: Association for Computing Machinery, juin 2019, pp. 105–114 (2019). https://doi.org/10.1145/3292522.3326028
17. Amjad, M., Ansari, M.Z., Alam,N.: An MLP based approach of hate speech detection on Twitter. vol. 6, no. 3 (2018))
18. J. Lemmens, J., Markov, I., Daelemans, W.: Improving hate speech type and target detection with hateful metaphor features. In: Proceedings of the Fourth Workshop on NLP for Internet Freedom: Censorship, Disinformation, and Propaganda, Online: Association for Computational Linguistics, pp. 7–1 (2021). https://doi.org/10.18653/v1/2021.nlp4if-1.2
19. Salminen, J., Hopf, M., Chowdhury, S.A., Jung, S.G., Almerekhi, H., Jansen, B.J.: Developing an online hate classifier for multiple social media platforms. Hum.-Centric Comput. Inf. Sci. **10**(1), 1 (2020). https://doi.org/10.1186/s13673-019-0205-6
20. Muhammad, S.H., et al.: AfriSenti: a twitter sentiment analysis benchmark for African languages. arXiv, 4 novembre 2023. Consulté le: 18 décembre 2023. http://arxiv.org/abs/2302.08956
21. Sreelakshmi, K., Premjith, B., Soman, K.P.: Detection of hate speech text in Hindi-English code-mixed data. Procedia Comput. Sci. **171**, 737–744 (2020). https://doi.org/10.1016/j.procs.2020.04.080
22. Patil, A., Patwardhan, V., Phaltankar, A., Takawane, G., Joshi, R.: Comparative study of pretrained BERT models for code-mixed Hindi-English data. In: 2023 IEEE 8th International Conference for Convergence in Technology (I2CT), avr. 2023, pp. 1–7 (2023). https://doi.org/10.1109/I2CT57861.2023.10126273

23. Lanasri, D., Olano, J., Klioui, S., Lee, S.L., Sekkai, L.: Hate speech detection in algerian dialect using deep learning. arXiv, 20 septembre 2023. http://arxiv.org/abs/2309.11611
24. Guellil, I., Adeel, A., Azouaou, F., Boubred, M., Houichi, Y., Moumna, A.A.: Ara-women-hate: an annotated corpus dedicated to hate speech detection against women in the Arabic community. In: Proceedings of the Workshop on Dataset Creation for Lower-Resourced Languages within the 13th Language Resources and Evaluation Conference, J. Sälevä et C. Lignos, Éd., Marseille, France: European Language Resources Association, juin 2022, pp. 6875 (2022)
25. O. Boucherit, O., Abainia, K.: Offensive language detection in under-resourced Algerian dialectal Arabic language. In: Big Data, Machine Learning, and Applications, vol. 1053, M. D. Borah, D. S. Laiphrakpam, N. Auluck, et V. E. Balas, Éd., in Lecture Notes in Electrical Engineering, vol. 1053, 2022, pp. 639–647 (2022). https://doi.org/10.1007/978-981-99-3481-2_49
26. Mbaye, D., Diallo, M., Diop, T.I.: Low-Resourced Machine Translation for Senegalese Wolof Language. (2023). https://doi.org/10.48550/ARXIV.2305.00606