HATE SPEECH DETECTION IN ALGERIAN DIALECT USING DEEP LEARNING

A PREPRINT

Dihia LANASRI
OMDENA
New York, USA
dihia.lanasri@gmail.com

Juan OLANO
OMDENA
New York, USA
juan_olano@yahoo.com

Sifal KLIOUI
OMDENA
New York, USA
sifal.klioui@gmail.com

Sin Liang Lee
OMDENA
New York, USA
mangojamlee@gmail.com

Lamia SEKKAI
OMDENA
New York, USA
lsekkai@gmail.com

September 22, 2023

ABSTRACT

With the proliferation of hate speech on social networks under different formats, such as abusive language, cyberbullying, and violence, etc., people have experienced a significant increase in violence, putting them in uncomfortable situations and threats. Plenty of efforts have been dedicated in the last few years to overcome this phenomenon to detect hate speech in different structured languages like English, French, Arabic, and others. However, a reduced number of works deal with Arabic dialects like Tunisian, Egyptian, and Gulf, mainly the Algerian ones. To fill in the gap, we propose in this work a complete approach for detecting hate speech on online Algerian messages. Many deep learning architectures have been evaluated on the corpus we created from some Algerian social networks (Facebook, YouTube, and Twitter). This corpus contains more than 13.5K documents in Algerian dialect written in Arabic, labeled as hateful or non-hateful. Promising results are obtained, which show the efficiency of our approach.

Keywords Hate Speech · Algerian dialect · Deep Learning · DziriBERT · FastText

1 Introduction

Hate speech detection, or detection of offensive messages in social networks, communication forums, and websites, is an exciting and hot research topic. Many hate crimes and attacks in our current life started from social network posts and comments MacAvaney et al. [2019]. Studying this phenomenon is imperative for online communities to keep a safe environment for their users. It also has a significant benefit for security authorities and states to ensure the safety of citizens and prevent crimes and attacks.

A universally accepted definition of hate speech is currently unavailable Bogdani et al. [2021] because of the variation of cultures, societies, and local languages. Other difficulties include the diversity of national laws, the variety of online communities, and forms of online hate speech. Various definitions are proposed.

According to the Encyclopedia of the American Constitution: "Hate speech is speech that attacks a person or group based on attributes such as race, religion, ethnic origin, national origin, sex, disability, sexual orientation, or gender identity." Nockleby [2000]. Today, many authors largely used this definition Guellil et al. [2022]. Facebook considers hate speech as "a direct attack on people based on protected characteristics—race, ethnicity, national origin, religious affiliation, sexual orientation, caste, sex, gender, gender identity, and serious disease or disability. We also provide some protections for immigration status." [1]. Davidson et al., who defines hate speech as "language that is used to express hatred towards a targeted group or is intended to be derogatory, to humiliate, or to insult the members of the group" propose one of the most accepted definitions Davidson et al. [2017]. Alternatively, the one proposed by Fortuna et al., "Hate speech is a language that attacks or diminishes, that incites violence or hate against groups, based on specific characteristics such as physical appearance, religion, descent, national or ethnic origin, sexual orientation, gender identity or other, and it can occur with different linguistic styles, even in subtle forms or when humor is used." Fortuna and Nunes [2018].

The literature review shows that the term *Hate speech* (which is the most commonly used) has various synonym terms such as abusive speech, offensive language, cyberbullying, or sexism detection Schmidt and Wiegand [2017]. Many works have been published in the context of hate speech detection for different standard and structured languages, like French Battistelli et al. [2020], English Alkomah and Ma [2022], Spanish Plaza-del Arco et al. [2021], and Arabic Albadi et al. [2018]. These languages are known for their standardization with well-known grammar and structure, which make the language processing well mastered. However, detecting hate speech in dialects, mainly Arabic ones such as Libyan, Egyptian, and Iraqi, etc. is still challenging and complex work Mulki et al. [2019]. Even if they are derived from the literal Arabic language, each country's specific vocabulary and semantics are added or defined.

In this work, we are interested in detecting hate speech in the Algerian dialect. This latter is one of the complex dialects Mezzoudj et al. [2019] characterized by the variety of its sub-dialects according to each region within the country. Algeria is a country with 58 regions; each one has a specificity in its spoken language with different words and meanings. The same word may have various meanings for each region; for example, '*Flouka*' in the east means '*earrings*.' In the north, it means '*small boat*'.

Moreover, new '*odd*' words are continually added to the Algerian vocabulary. The Algerian dialect is known for its morphological and orthographic richness. Facing this situation, treating and understanding the Algerian dialect for hate speech detection is a complex work. The importance of this project for the Algerian context encourages us to work on this problem.

To the best of our knowledge, only few works have been proposed for hate speech detection in the Algerian dialect Boucherit and Abainia [2022], Menifi et al. [2022]. Some other related topics are treated like sentiment analysis Abdelli et al. [2019], sexism detection Guellil et al. [2021] which may be exploited to analyze the hate speech.

In this paper, we proposed a complete end-to-end natural language processing (NLP) approach for hate speech detection in the Algerian dialect. Our approach covers the main steps of an NLP project, including data collection, data annotation, feature extraction, and then model development based on machine and deep learning, model evaluation, and inference.

Moreover, we have evaluated various machine and deep learning architectures on our corpus built from diverse social networks (YouTube, Twitter, and Facebook) for several years (between 2017 and 2023). This corpus contains more than 13.5K annotated documents in Algerian dialect written in Arabic characters. Two classes are used for annotation (hateful, non-hateful). This work allows us essentially to provide a wealthy evaluation of many deep learning architectures, an essential value for academic and industrial communities. The obtained results are promising, and continuous tests are performed for further results.

This paper is structured as follows: Section 2 presents a necessary background, Section 3 reviews the most important related works, Section 4 details our proposed approach and evaluated models, Section 5 discusses the obtained results, and Section 6 concludes the paper.

2 Background

Hate speech is commonly defined as any communication that disparages a target group of people based on some characteristic such as race, color, ethnicity, gender, sexual orientation, nationality, religion, or other characteristic De Gibert et al. [2018].

2.1 Hate speech

According to Al-Hassan and Al-Dossari [2019] hate speech is categorized into five categories: (1) gendered hate speech, including any form of misogyny and sexism; (2) religious hate speech including any religious discrimination, such as Islamic sects, anti-Christian, etc.; (3) racist hate speech including any racial offense or tribalism, and xenophobia; (4) disability including any sort of offense to an individual suffering from health problems; and (5) political hate speech can refer to any abuse and offense against politicians Guellil et al. [2022].

[1] Community Standards; Available on:https://www.facebook.com/communitystandards/objectionable_content

[2] https://worldpopulationreview.com/country-rankings/arabic-speaking-countries

2.2 Algerian Dialect and Arabic Languages

Arabic is the official language of 25 countries[2]. More than 400 million people around the world speak this language. Arabic is also recognized as the 4th most-used language on the Internet Boudad et al. [2018]. Arabic is classified into three categories Habash [2022]: (1) Classical Arabic (CA), which is the form of the Arabic language used in literary texts. The Quran is considered the highest form of CA text Sharaf and Atwell [2012]. (2) Modern Standard Arabic (MSA) is used for writing and formal conversations. (3) Dialectal Arabic is used in daily life communication, informal exchanges, etc. Boudad et al. [2018] like the Algerian dialect, Tunisian dialect, etc.

The Algerian dialect on social networks can be written with Arabic characters (كرهت من هادي الميزيرية), Latin characters (kreht men hadi el miziria), or a mix of them. This dialect does not respect a specific syntax or grammar; the same word may have many meanings according to each region in Algeria (having 58 regions). In addition, the same word may be written in different manners (exp. Inchallah , nchallah, n'shala, etc. to say hopefully). This paper does not deal with the Amazigh (Berber) language, another language spoken and written in Algeria, which is entirely different from the Algerian dialect. It has its own vocabulary and grammar.

3 Related Work

A restricted number of works have been published in the context of hate speech detection dealing with Arabic dialects. This section will analyze the most important NLP approaches dedicated to (1) the Algerian dialect and (2) Other Arabic dialects like Iraqi, Egyptian, Syrian, and Tunisian. This analysis helps us to identify the used approaches, models, and corpora.

3.1 Hate speech detection in Algerian dialect

Guellil et al. [2022, 2021] Developed the first approach and corpus in Algerian dialect for hate speech detection against women in Arabic community on social media. This corpus contains more than 373K YouTube comments. Two different algorithms for feature extraction were used: Word2vec with machine learning models (GaussianNB, LogisticRegression, RandomForset, SGDClassifier, and LinearSVC) and FastText with Deep learning models (deep Convolutional Neural Network (CNN), long short-term memory (LSTM) network and Bi-directional LSTM (BiLSTM) network). Simulation results demonstrated the best performance of the CNN model with FastText.

Boucherit and Abainia [2022] addressed the problem of detecting offensive and abusive content in Facebook comments. The corpus contains 8.7K comments in Algerian dialect written in Arabic and Latin characters, manually annotated as usual, abusive, and offensive. They used BiLSTM, CNN, FastText, SVM, and Multinomial Naive Bayes (NB) as classifiers. The experimental results showed that SVM and Multinomial NB classifiers outperformed all the other classifiers.

Abainia et al. [2022] addressed the offensive language detection in the Amazigh language, which is one of the under-resourced languages. They were interested in the Kabyle dialect. A new corpus of offensive Amazigh language is proposed containing 6.2K documents collected from Facebook and manually annotated as usual or offensive. A new lexicon of offensive and abusive Amazigh words with 12.6k entries is also developed. Many models have been evaluated, like SVM and Multinomial Naive Bayes classifiers tested with tf-idf. FastText was tested with deep learning models CNN and BiLSTM. The naive statistical classifier based on lexicon checking was the winner classifier.

Mazari and Kheddar [2023] introduced a new dataset for Algerian dialect toxic text detection. An annotated multi-label dataset is built, containing around 14K comments extracted from Facebook, YouTube, and Twitter and labeled as hate speech, offensive language, and cyberbullying. Several tests have been conducted using many classification models of traditional machine learning: Random Forest, Naıve Bayes, Linear Support Vector (LSV), Stochastic Gradient Descent (SGD), and Logistic Regression. Furthermore, several assessments have been conducted using Deep Learning models such as CNN, LSTM, Gated Recurrent Unit (GRU), BiLSTM and Bidirectional-GRU (Bi-GRU). Results demonstrated the best performance of LSV, BiLSTM, and MLP when associated with the SGD model.

Guellil et al. [2020] proposed a system for detecting hateful speech in Arabic political debates. The approach was evaluated against a hateful corpus concerning Algerian political debates. It contains 5K YouTube comments in MSA and Algerian dialects, written in both Arabic and Latin characters. Both classical algorithms of classification (Gaussian NB, Logistic Regression, Random Forest, SGD Classifier, and Linear SVC(LSVC)) and deep learning algorithms (CNN, multilayer perceptron (MLP), LSTM, and BiLSTM) are tested. For extracting features, the authors use Word2vec and FastText with their two implementations, namely, Skip Gram and CBOW. Simulation results demonstrate the best performance of LSVC, BiLSTM and MLP.

Mohdeb et al. [2022] proposed an approach for analysis and the detection of dialectal Arabic hate speech that targeted African refugees and illegal migrants on the YouTube Algerian space. The corpus contains more than 4K comments annotated as Incitement, Hate, Refusing with non-hateful words, Sympathetic, and Comment. The transfer learning approach has been exploited for classification. The experiments show that the AraBERT monolingual transformer outperforms the mono-dialectal transformer DziriBERT and the cross-lingual transformers mBERT and XLM-R.

3.2 Hate speech detection in other Arabic dialects

Various datasets or corpora were published in different dialects, which can be used for different purposes like hate speech, racism, violence, etc. detection.

ALBayari and Abdallah [2022] is the first work to propose a corpus built from Instagram comments. This corpus contains 198K comments, written in MSA and three different dialects: Egyptian, Gulf, and Levantine. The comments were annotated as neutral, toxic, and Bullying. Al-Ajlan and Ykhlef [2018] and Haidar et al. [2019] datasets are collected from Twitter containing respectively 20K and 34K multi-dialectal Arabic tweets annotated as bullying and non-bullying labels. These tweets were from various dialects (Lebanon, Egypt, and the Gulf area). Moreover, two other datasets were proposed by Mubarak et al. [2017]. The first one with 1.1K tweets in different dialects and the second dataset contains 32K inappropriate comments collected from a famous Arabic news site and annotated as obscene, offensive, or clean. Albadi et al. [2018] proposed the religious hate speech detection where a multi-dialectal dataset of 6.6K tweets was introduced. It included an identification of the religious groups targeted by hate speech. Alakrot et al. [2018] also provided a dataset of 16K Egyptian, Iraqi, and Libyan comments collected from YouTube. The comments were annotated as either offensive, inoffensive, or neutral.

T-HSAB Haddad et al. [2019] and L-HSAB Mulki et al. [2019] are two publicly available corpora for abusive hate speech detection. The first one is in the Tunisian dialect, combining 6K comments. The second one is in Levantine dialect (Syrian, Lebanese, Palestinian, and Jordanian dialects) containing around 6K tweets. These documents are labeled as Abusive, Hate, or Normal.

Mubarak et al. [2020] looked at MSA and four major dialects (Egyptian, Levantine, Maghrebi, and Gulf). It presented a systematic method for building an Arabic offensive language tweet dataset that does not favor specific dialects, topics, or genres with 10K tweets. For tweet labeling, they used the count of positive and negative terms based on a polarity lexicon. FastText and Skip-Gram (AraVec skip-gram, Mazajak skip-gram); and deep contextual embeddings, namely BERTbase-multilingual and AraBERT are used. They evaluated different models: SVM, AdaBoost, and Logistic regression.

Mulki and Ghanem [2021] introduced the first Arabic Levantine Twitter dataset for Misogynistic language (LeT-Mi) to be a benchmark dataset for automatic detection of online misogyny written in the Arabic and Levantine dialect. The proposed dataset consists of 6.5K tweets annotated either as neutral (misogynistic-free) or as one of seven misogyny categories: discredit, dominance, cursing/damning, sexual harassment, stereotyping and objectification, derailing, and threat of violence. They used BOW + TF-IDF, SOTA, LSTM, BERT, and Majority class as classifiers.

Duwairi et al. [2021] investigated the ability of CNN, CNN-LSTM, and BiLSTM-CNN deep learning networks to classify or discover hateful content posted on social media. These deep networks were trained and tested using the ArHS dataset, which consists of around 10K tweets that were annotated to suit hateful speech detection in Arabic. Three types of experiments are reported: first, the binary classification of tweets into Hate or Normal. Ternary classification of tweets into (Hate, Abusive, or Normal), and multi-class classification of tweets into (Misogyny, Racism, Religious Discrimination, Abusive, and Normal).

Aldjanabi et al. [2021] have built an offensive and hate speech detection system using a multi-task learning (MTL) model built on top of a pre-trained Arabic language model. The Arabic MTL model was experimented with two different language models to cover MSA and dialect Arabic. They evaluated a new pre-trained model 'MarBERT' to classify both dialect and MSA tweets. They propose a model to explore multi-corpus-based learning using Arabic LMs and MTL to improve the classification performance.

Haidar et al. [2017] presented a solution for the issue of cyberbullying in both Arabic and English languages. The proposed solution is based on machine learning algorithms using a dataset from Lebanon, Syria, the Gulf Area, and Egypt. That dataset contained 35K Arabic texts. In this research, Naïve Bayes and SVM models were chosen to classify the text. The SVM model achieved greater precision.

Abdelali et al. [2016] The authors built a large dataset that consists of offensive Arabic words from different dialects and topics. The tweets were labeled into one of these categories: offensive, vulgar, hate speech, or clean. Since the offensive tweets involve implicit insults, the hate speech category was the tweets that contain racism, religious, and ethnic words. Different classifiers were employed in this study; the SVM model with a radial function kernel was mainly used with lexical features and pre-trained static embedding, while Adaptive Boosting and Logistic regression classifiers were employed when using Mazajak embedding. SVM gave the best precision.

According to this literature analysis, we detect that the topic of hate speech detection in the Algerian dialect is not widely considered, and only few works deal with this problem. Furthermore, a lack of Algerian datasets prepared for hate speech is found. All these findings motivate our proposal.

4 Our Methodology

To identify hate speech in messages written in Algerian dialects—whether in Arabic or Latin script— we outline a comprehensive methodology encompassing (1) data gathering, (2) data annotation, (3) feature extraction, (4) model development, and (5) model evaluation and inference. We'll delve into each of these stages in the subsequent sections.

4.1 Data Collection

Data collection serves as the foundational step in our approach. To effectively train our models, we require a robust dataset in the Algerian Arabic dialect. To achieve this, we sourced our data from three distinct social networks spanning the years 2017 to 2023:

1. YouTube: Numerous Algerian channels have emerged on YouTube, dedicated to discuss various topics, including politics, religion, social issues, youth concerns, education, and more. We have identified and focused on the most influential ones with a significant following and engagement. We employ the YouTube Data API through a Python script to gather comments from various videos.

2. Twitter: Even if Algerian citizens do not widely use Twitter, we targeted it to collect tweets. We used a list of keywords to search for tweets. Many hashtags were launched between 2017 and 2023 about some situations and crises in Algeria, which enhanced the activity on Twitter, like ما تشريش زيت ربراب (do not buy oil of rebrab), يتنحاو قاع (remove them all), العصابة (mafia), لا للعهدة الخامسة (no for fifth presidential term), etc. During this activity, we used these hashtags to collect an important number of tweets. Two techniques have been used for this objective: (1) Using Twitter API: Until February 2023, we were able to use this API for free and gather tweets. (2) Since February 2023, this API has become paid. Consequently, we used other solutions based on scrapping using the SNScrape library.

3. Facebook: To gather data from Facebook, we selected public pages talking and sharing content about politics, Algerian products, pages of some influencers, mobile operators, etc. We collected the posts, comments, and replies from these various pages. To collect data, we used different solutions: (1) Between 2017 and 2018, we were able to collect data from any public page using Graph API. (2) Since 2019, we have used either FacePager free application to collect data from public pages or (3) Facebook-scraper library for scraping.

From these sources, we have collected more than 2 million documents (messages) in different languages: Arabic, French, English, dialect, etc. The next step consists of filtering only documents written in Algerian dialects, either in Arabic or Latin characters. This work was done manually by a group of collaborators. At the end, we obtained around 900K documents.

4.2 Data Annotation (Data Labeling)

To annotate data, we followed two approaches: automatic and manual. We have decided to annotate only the dialect written in Arabic characters. Our approach consists of building one model that detects hate speech only for Algerian dialects written in Arabic characters. Then, a transliteration function is developed to transliterate any Algerian document

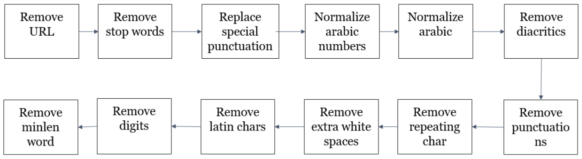

Figure 1: Preprocessing pipeline

written in Latin characters into Arabic ones, then use the built model to classify it. For example, "ma tech-rich zit el aliha" becomes "ما تشريش زيت الالهة" which means "Don't buy the oil of the gods," which expresses the expensiveness of this oil.

We used a binary annotation: 0 expressing NON-HATE, which represents a document that doesn't contain any hateful or offensive word. 1 in case of a Hateful message containing any hateful word or the meaning and the semantics of the message expresses it.

1- Automatic annotation: For automatic annotation, we prepared a set of hateful keywords in the Algerian dialect discovered from our corpus. These words express the hate and the violence in Algerian speech. This list contains 1.298 words. This list of keywords has been used in a Python script to automatically tag a document with 1 if it contains at least one hateful keyword. In the other case, it is considered as 0. The automatically annotated corpus contains 200K Algerian documents written in Arabic characters.

2- Manual annotation: The automatically annotated documents have been validated manually. A group of annotators checked the annotated corpus and corrected the wrong-labeled documents. The manual step validates 5.644 documents considered for the next step.

3- Dataset Augmentation for Enhanced Balance: To bolster our dataset and enhance its equilibrium, we employed a strategy involving the incorporation of positively labeled subsets sourced from sentiment analysis datasets. In doing so, we reclassified these subsets as non-hateful, under the reasonable assumption that expressions of positive sentiment inherently exclude hate speech. Specifically, we leveraged the dataset available at `https://www.kaggle.com/datasets/djoughimehdi/algerian-dialect-review-for-sentiment-analysis`, selecting solely the instances characterized by positive sentiment and relabeling them as 'normal.' However, due to preprocessing constraints, this process yielded a reduced set of just 500 documents.

Moreover, we used the corpus shared by Boucherit and Abainia [2022] containing 8.7K documents in the Algerian dialect. This dataset is labeled manually as Offensive (3.227), Abusive (1.334), and Normal (4.188). We changed the labels of this corpus into Hateful (1) for fused Offensive and Abusive ones and Non-Hateful (0) for Normal ones. This corpus has been filtered and treated to keep 7.345 labeled documents.

At the end of this step, we obtained an annotated balanced corpus of 13.5K documents in Algerian dialect written in Arabic characters, which will be used later to build classifiers.

4.3 Data Preprocessing

Before using any dataset, a cleaning or preprocessing task should be performed. We have defined a set of functions orchestrated in a pipeline, as illustrated in Figure 1.

- Remove URL: All URLs in a document are deleted.

- Remove stop words: The list of Arabic stop words provided by NLTK is used to clean meaningless words. This list has been enriched by a set of stop words detected in the Algerian dialect.

- Replace special punctuation: some punctuation concatenation can represent meaning and have an added value for the model, like: :) Means happy, :(Means upset, etc. This kind of punctuation is transformed into the corresponding emoji.

- Normalize Arabic numbers: Arabic numbers are transformed into the classic digits in order to standardize the writing, like ١ = 1, ٢=2, etc.

- Normalize Arabic: some letters have special symbols, which needs some treatment. Like:ـآك آگ ـآيْـنْ

- Remove diacritics: like the vowel marks «آّ» representing Tashdid; «أَ» meaning Fatha, etc.

- Remove punctuation: All punctuation except the ones representing emotions are deleted.

- Remove repeating character: Any repeated character is removed, keeping just one occurrence.

- Remove extra white spaces: All extra white spaces are deleted.

- Remove Latin chars: Latin characters in Arabic text are removed to avoid incoherence.

- Remove digits: In the Arabic dialect, digits do not have any added value. They are removed

- Remove min-length words: Some words written in less than two positions are deleted. The experiences show that these words are meaningless.

The preprocessing of the data was meticulously crafted to cater to the unique characteristics of the Algerian dialect text. By applying rigorous preprocessing to the dialect, the data was made consistent and well-suited for training our models. These preprocessing steps were vital in ensuring that the model was sensitive to the nuances of the language and could effectively classify hate and non-hate content.

4.4 Data Splitting

In the development of the models for binary classification of hate speech in the Algerian dialect, our corpus was loaded from CSV file and was stratified before it was split into three distinct sets: training (80%), validation (10%), and testing (10%). The stratification is based on the label column to maintain a balanced representation of hate (1) and non-hate (0) content in each subset. The training set facilitated the model training, while the validation set allowed for unbiased model evaluation during training to prevent over-fitting. The testing set served as an objective assessment of the model's generalization performance beyond the training data. Tokenization, which includes padding and truncation, is also performed.

4.5 Model Development

In this work, we evaluated many classifiers from machine and deep learning. Below, we discuss the architecture, methodology, and performance of each model.

1. Linear Support Vector Classifier (LinearSVC): We utilized a Linear Support Vector Classifier (LinearSVC) model to investigate how a traditional machine learning approach would perform on this task. The TF-IDF (Term Frequency-Inverse Document Frequency) method was employed to convert the text data into a numerical format suitable for machine learning models. The model was initialized with default parameters and trained on the feature matrix obtained from the TF-IDF vectorization.

2. Gzip + KNN: Deep Neural Networks are potent learners capable of tackling a wide array of tasks. However, for relatively straightforward tasks like topic classification they often prove excessive due to their substantial data requirements, high computational demands, and the need for meticulous hyper-parameter tuning. This part of the research centers on a more straightforward alternative known as "compressor-brd text classification," requiring no training parameters. which, despite its astonishing simplicity, exhibits interesting results. The approach comprises three key components: (1) utilization of a conventional lossless compression algorithm (gzip in this study); (2) application of the compressor-brd distance metric (Normalized Compression Distance in this study); (3) Implementation of a traditional KNN classifier.

this approach takes a simple route by compressing text data using Gzip, measuring the similarity of compressed data using NCD, and then classifying text using the traditional KNN algorithm

3. LSTM & BiLSTM with Dziri FastText: LSTM and BiLSTM are one of the deep learning models that are suitable for NLP problems, mainly in text classification like sentiment analysis and even for hate speech detection. In this paper, we have tested these two models against our corpus. To learn the semantics and context of messages, we used FastText as a word embedding model. In our case, we fine tuned a Dziri FastText model. This later was trained on a huge dataset of Algerian messages in Arabic characters based on the Skip-gram model. The obtained model (Dziri FastText) is used to generate an embedding matrix for our built corpus of hate speech. The sequential architecture is composed of: (i) Embedding layer which is the input layer representing the embedding matrix; (ii) Dropout layer with a rate of 0.2 to prevent over-fitting; (iii) LSTM or Bidirectional LSTM layer with units=100, dropout=0.4, recurrent_dropout=0.2; (iv) Dropout layer with a rate of 0.2 to prevent over-fitting; (v) Output dense layer, using sigmoid as an activation function. As optimizer we used Adam, and we used binary crossentropy as a loss function, batch_size = 64 and epochs= 100.

4. Dziribert-FT-HEAD: Pre-trained transformers, like BERT, have become the standard in Natural Language Processing due to their exceptional performance in various tasks and languages. The authors in Abdaoui et al. [2021] collected over one million Algerian tweets and developed DziriBERT, the first Algerian language model, outperforming existing models, especially for the Latin script (Arabizi). This demonstrates that a specialized model trained on a relatively small dataset can outshine models trained on much larger datasets. The authors have made DziriBERT[3] publicly available to the community.
In this experiments we fine-tuned Dziribert, by incorporating a classification head while keeping the rest of the Dziribert parameters frozen. The classification head consists of three key components: a fully connected layer with 128 units, followed by batch normalization for stability, a dropout layer to mitigate overfitting, and a final fully connected layer that produces a single output value. We apply a sigmoid activation function to ensure the output falls between 0 and 1, which suits our binary classification task. Training employed the binary cross-entropy loss function and the Adam optimizer with a fixed learning rate of 1e-3. Additionally, a learning rate scheduler was employed to dynamically adjust the learning rate during training for improved convergence.

5. DZiriBert with Peft+LoRA: In our experiment, we fine-tuned the pre-trained model "DZiriBERT" using techniques called Peft (Parameter-Efficient Fine-Tuning) Mangrulkar et al. [2022] + LoRA Hu et al. [2021]. These methodologies allowed us to tailor the model specifically for the Algerian dialect, making it sensitive to the unique nuances of this language. The Peft configuration is established using the LoRa technique. Parameters such as the reduction factor, scaling factor, dropout rate, and bias are defined according to the task requirements.
Peft and LoRa Configuration: PEFT method has recently emerged as a powerful approach for adapting large-scale pre-trained language models (PLMs) to various downstream applications without fine-tuning all the model's parameters. Given that fine-tuning such models can be prohibitively costly, PEFT offers a viable alternative by only fine-tuning a small number of (extra) model parameters. This greatly decreases the computational and storage costs without compromising performance.
LoRA is a technique specifically designed to make the fine-tuning of large models more efficient and memory-friendly. The essential idea behind LoRA is to represent weight updates using two smaller matrices (referred to as update matrices) through a low-rank decomposition. While the original weight matrix remains frozen, these new matrices are trained to adapt to the new data, keeping the overall number of changes minimal. LoRA has many advantages, mainly the: (1) Efficiency: by significantly reducing the number of trainable parameters, LoRA makes fine-tuning more manageable. (2) Portability: Since the original pre-trained weights are kept frozen, multiple lightweight LoRA models can be created for various downstream tasks. (3) Performance: LoRA achieves performance comparable to fully fine-tuned models without adding any inference latency. (4) Versatility: Though typically applied to attention blocks in Transformer models, LoRA's principles can, in theory, be applied to any subset of weight matrices in a neural network.
Model Initialization: DZiriBERT is loaded and configured with Peft using the defined parameters. The model is then fine-tuned using the tokenized datasets. We configure our model using the LoraConfig class, which includes the following hyperparameters:
- Task Type: We set the task type to Sequence Classification (SEQ_CLS), where the model is trained to map an entire sequence of tokens to a single label. Target Modules: The target modules are set to "query" and "value".
- Rank (r): We employ a low-rank approximation with a rank =16 for the LoRA matrices.
- Scaling Factor (α): The LoRA layer utilizes a scaling factor=32, which serves as a regularization term.
- Dropout Rate: We introduce a dropout rate of 0.35 in the LoRA matrices to improve generalization.
- Bias: The bias term is set to "none," reducing the model complexity.

Training Process: The model is trained using custom training arguments, including learning rate, batch sizes, epochs, and evaluation strategies. The training process leverages the Hugging Face Trainer class, providing a streamlined

[3] `https://huggingface.co/alger-ia/dziribert`

approach to model fine-tuning. We train our model with the following parameters:
-learning_rate=1e-3: Specifies the learning rate as 1e-3. Learning rate controls how quickly or slowly a model learns during the training process.
-per_device_train_batch_size=16: This indicates that each device used for training (usually a GPU) will handle a batch of 16 samples during each training iteration.
- per_device_eval_batch_size=32: Similar to the above, but for evaluation, each device will process batches of 32 samples.
- num_train_epochs=5: The training process will go through the entire training dataset 5 times. An epoch is one complete forward and backward pass of all the training examples.
- weight_decay=0.01: This is a regularization technique that helps prevent the model from fitting the training data too closely (overfitting). A weight decay of 0.01 will be applied.
- evaluation_strategy="epoch": Evaluation will be performed at the end of each epoch. This allows you to check the performance of your model more frequently and make adjustments if needed.
- save_strategy="epoch": The model will be saved at the end of each epoch, allowing you to revert to the model's state at the end of any given epoch if necessary.
- load_best_model_at_end=True: Once all training and evaluation are completed, the best-performing model will be loaded back into memory. This ensures that you always have access to the best model when your training is complete.

6. Dzarashield: We built the Dzarabert[4] which is a modification of the original Dziribert model that involves pruning the embedding layer, specifically removing tokens that contain non-Arabic characters. This pruning significantly reduces the number of trainable parameters, resulting in faster training times and improved inference speed for the model. This approach is aimed at optimizing the model's performance for tasks involving Arabic-based text while minimizing unnecessary complexity and computational overhead. Dzarashield[5] is built upon the Dzarabert base model by incorporating a classification head. This classification head consists of sequential architecture including: a linear layer (input: 768, output: 768), followed by a Rectified Linear Unit (ReLU) activation function; a dropout layer (dropout rate: 0.1); and another linear layer (input: 768, output: 2) for binary classification. The model's hyperparameters were determined through experimentation: a learning rate (lr) of 1.3e-05, a batch size of 16, and training for 4 epochs. The Adam optimizer was used with its default parameters for optimization during training. Experimentation resulted in a better score when updating all the weights of the model rather than freezing the base BERT model and updating the classification head.

7. Multilingual E5 Model: We conducted a fine-tuning process on a pre-existing model, specifically the Multilingual E5 base model Wang et al. [2022]. Our primary objective was to ascertain the efficacy of a multilingual model within the context of the Algerian dialect. In adherence to the training methodology, the prefix "query:" was systematically introduced to each data row. This precautionary measure was deemed necessary Wang et al. [2022] to avert potential indications of performance deterioration that might arise in the absence of such preprocessing. The foundation of our investigation rested upon the initialization of the pre-trained base model using the xlm-roberta-base[6] architecture, which was trained on a mixture of multilingual datasets. The model is fine-tuned with an additional Dense layer followed by a Dropout Layer. The model is trained with custom hyperparameters for fine-tuning (Warmup Steps: 100; Weight Decay: 0.01 ; Epoch: 5 ; Probability of Dropout: 0.1; Train batch size: 16 ; Evaluation batch size: 64)

8. sbert-distill-multilingual Fine Tuned: Similar to the Multilingual E5 Model, we fine-tuned a pre-trained model known as sbert-distil-multilingual model from sentence transformer to investigate how well a multilingual model performs in Algerian Dialect. The pre-trained model is based on a fixed (monolingual) teacher model that produces sentence embeddings with our desired properties in one language. The student model is supposed to mimic the teacher model, i.e., the same English sentence should be mapped to the same vector by the teacher and by the student model. The model is fine-tuned with an additional Dropout layer and a GeLU layer via K-Fold cross validation. The model is trained with custom hyperparameters for fine-tuning (Warmup Steps: 100; Weight Decay: 0.01; Probability of Dropout: 0.1 ; Epoch: 10 ; K-Fold: 4 ; Train batch size: 16 ; Evaluation batch size: 64)

9 AraT5v2-HateDetect AraT5-base is the result of testing the T5 model (mT5)[7] on Arabic. For comparison, three robust Arabic T5-style models are pre-trained and evaluated on ARGEN dataset Nagoudi et al. [2021]. Surprisingly, despite being trained on approximately 49% less data, these models outperformed mT5 in the majority of

[4] https://huggingface.co/Sifal/dzarabert
[5] https://huggingface.co/Sifal/dzarashield
[6] https://huggingface.co/xlm-roberta-base
[7] https://huggingface.co/docs/transformers/model_doc/mt5

Table 1: The results of each model (FT: Fine Tuned

Model Name	Accuracy	Precision	Recall	F1 Score
LinearSVC	0.83	0.84(Class0); 0.72(Class1)	0.96(Class0); 0.36 (Class1)	0.9(Class0); 0.48 (Class1)
gzip + KNN	0.67	0.63	0.56	0.60
Dziribert-FT-HEAD	0.83	0.81	0.81	0.81
LSTM	0.70	0.61	0.75	0.67
Bidirect LSTM	0.68	0.59	0.81	0.68
DZiriBERT FT PEFT+LoRA	0.86	0.83	0.85	0.84
Multilingual-E5-base FT	0.84	0.8	0.81	0.80
sbert-distill-multilingual FT	0.80	0.74	0.81	0.77
DzaraShield	0.87	0.87	0.87	0.87
AraT5v2-HateDetect	0.84	0.83	0.84	0.83

ARGEN tasks, achieving several new state-of-the-art results. The AraT5v2-base-1024 model [8] introduces several improvements compared to its predecessor, AraT5-base :

- More Data: AraT5v2-base-1024 is trained on a larger and more diverse Arabic dataset. This means it has been exposed to a wider range of Arabic text, enhancing its language understanding capabilities.
- Larger Sequence Length: This version increases the maximum sequence length from 512 to 1024. This extended sequence length allows the model to handle longer texts, making it more versatile in various NLP tasks.
- Faster Convergence: During the fine-tuning process, AraT5v2-base-1024 converges approximately 10 times faster than the previous version (AraT5-base). This can significantly speed up the training and fine-tuning processes, making it more efficient.
- Extra IDs: AraT5v2-base-1024 supports 100 sentinel tokens, also known as unique mask tokens. This allows for more flexibility and customization when using the model for specific tasks.

Overall, these enhancements make AraT5v2-base-1024 a more powerful and efficient choice for Arabic natural language processing tasks compared to its predecessor, and it is recommended for use in place of AraT5-base. AraT5v2-HateDetect[9] is a fine-tuned model based on AraT5v2-base-1024, specifically tailored for the hate detection task. The fine-tuning process involves conditioning the decoder's labels, which include target input IDs and target attention masks, based on the encoder's source documents, which consist of source input IDs and source attention masks. After experimentation, the following hyperparameters were chosen for training AraT5v2-HateDetect (Training Batch Size: 16; Learning Rate: 3e-5; Number of Training Epochs: 4). These hyperparameters were determined to optimize the model's performance on the hate detection task. The chosen batch size, learning rate, and training epochs collectively contribute to the model's ability to learn and generalize effectively for this specific NLP task.

4.6 Evaluation and Inference

To evaluate the different models, we used four main metrics: Accuracy, Precision, F1-Score, and Recall. To classify a message in case where it is written in Arabizi (a specific dialect using Latin characters), a transliteration process was implemented to convert the text into Arabic characters based on lang-trans[10] library.

5 Experiments and Results

To train and evaluate our models, we used TensorFlow or Pytorch deep learning frameworks. We used Google Colab and Kaggle GPUs to accelerate the experiments. In table **??**, we will provide the detailed results that we obtained.

Linear Support Vector Classifier (LinearSVC): The LinearSVC model offered a competitive accuracy but struggled with the recall for the hate speech class. The precision and recall trade-off indicates possible challenges in differentiating between the subtle nuances of hate and non-hate speech in the dialect. The model exhibited high precision and recall for class 0 but showed room for improvement for class 1, particularly in terms of recall. This suggests that while the model is quite good at identifying class 0, it could be improved for identifying class 1.

[8] https://huggingface.co/UBC-NLP/AraT5v2-base-1024

[9] https://huggingface.co/Sifal/AraT5v2-HateDetect

[10] https://pypi.org/project/lang-trans/

gzip + KNN: One of the worst models in terms of capabilities, although it is diverging from the baseline it is unclear whether these results will hold in out of distribution cases, especially when we know that there is no underlying process in the model that captures semantic representations of the documents.

Dziribert-FT-HEAD: the model exhibits a noteworthy precision score, signifying its accuracy in correctly classifying instances as hate speech or not. However, the relatively lower recall score suggests that it missed identifying some hate speech instances. This discrepancy might be attributed to the model's lack of specialized handling for the nuances of the Algerian dialect, potentially causing it to overlook certain hate speech patterns unique to that context.
Despite this, the model's overall accuracy remains commendably high, indicating its robust performance in making accurate predictions. Additionally, the balanced precision and recall values underline its ability to strike a reasonable trade-off between minimizing false positives and false negatives, a crucial aspect in hate speech detection.
The F1 Score, being the harmonic mean of precision and recall, further validates the model's capacity to effectively identify positive samples while avoiding misclassification of negative ones. The model consistently demonstrates strong performance across multiple evaluation metrics, especially in terms of accuracy and F1 score. These results reaffirm the practicality and effectiveness of employing deep learning techniques for the challenging task of hate speech detection.

LSTM and BiLSTM with FastText-DZ: Unfortunately, the results of this model are among the worst ones. The literature shows the strength of LSTM and BiLSTM in this kind of NLP project, but this is not the case for this project. The low precision is due to the incapability of the model to classify correctly the hate class. FastText is a good word embedding model that captures the context and semantics of a document. However, in this case, it does not perform well because of the fine-tuning done where we took an Arabic FastText and fine-tune it on Algerian dataset written in Arabic characters.

DZiriBert with Peft+LoRA: We utilize both PEFT and LoRA to fine-tune DZiriBERT, a model specifically adapted to the Algerian dialect. By employing these techniques, we were able to create a highly effective and efficient model for hate speech detection in the Algerian dialect while keeping computational costs at a minimum.

Multilingual-E5-base Fine Tuned and sbert-distill-multilingual Fine Tuned: The outcomes obtained from these models are noteworthy; nonetheless, their performances pale when compared with the parameter-efficient fine-tuning on the DZiriBERT model.

DzaraShield: The results returned by this model are satisfying considering the relatively low quantity of data it was finetuned on, this exhibits further that the pretraining plays the major role on downstream takes such as classification in our case, especially that the base model is an encoder only architecture which captures contextual information from the input data, making it useful for a wide range of text classification tasks.

AraT5v2-HateDetect: The results are slightly inferior to Dzarashield. One possible explanation is the increased complexity of the architecture when compared to the Dzarabert base model. Consequently, fine-tuning becomes a more intricate task due to the larger hyperparameter search space and the limited resources in terms of computing power and data availability. As a result, it is reasonable to expect that these models would perform similarly in real-world scenarios.

5.1 Results Discussion

The DzaraShield model has demonstrated remarkable capability in detecting hate speech in the Algerian dialect. Its outstanding precision score highlights its reliability in accurately identifying instances of hate speech. Additionally, it maintains a balanced precision and recall, indicating that it does not excessively sacrifice precision to achieve its higher recall. Such a balanced model holds considerable advantages, particularly when both false positives and false negatives carry significant consequences.

For the other models, mainly LSTM or BiLSTM with Dziri FastText, more fine-tuning should be performed to enhance the results. Moreover, future work may include hyperparameter tuning, class balancing techniques, or the integration of more complex models to improve performance across both classes.

The disparity between precision and recall in certain models warrants further investigation. Delving deeper into this issue could yield valuable insights into specific aspects of the dialect that might be contributing to this imbalance. Future experiments should prioritize understanding and addressing these discrepancies, with the goal of enhancing recall without compromising precision.

The results from various experimental models underscore the intricacies involved in hate speech detection in the Algerian dialect. While traditional machine learning and deep learning approaches provided some valuable insights, they fell short in capturing the dialect's nuanced characteristics. In contrast, the DzaraShield model emerged as the most successful approach, emphasizing the pivotal role of Encoder-only models in the realm of projects of this nature.

These findings offer valuable insights for future work in this area and underscore the potential of leveraging domain-specific knowledge, advanced fine-tuning techniques, and sophisticated architectures for the effective detection of hate speech in under-studied and complex dialects such as Algerian.

6 Conclusion

The importance of hate speech detection on social networks has encouraged many researchers to build solutions (corpora and classifiers) to detect suspect messages. The literature review shows that most works are interested in text in structured languages like English, French, Arabic, etc. However, few works deal with dialects, mainly the Algerian one, which is known for its complexity and variety. To fill in the gap, we propose in this paper a complete NLP approach to detect hate speech in the Algerian dialect. We built an annotated corpus of more than 13,5K documents, which is used to evaluate various deep learning architectures. The obtained results are very promising, where the most accurate was the DzaraShield .

Looking ahead, there is significant potential to enhance inference speed, particularly for the Dziribert-based and multilingual models. While this project primarily focused on Arabic characters, our next step will be to address the dialect when written in Latin characters. Embracing both Arabic and Latin characters will more accurately capture the nuances of the written Algerian dialect. Finally, we plan to expand our corpus size and explore alternative deep-learning architectures.

7 Acknowledgments

We would like to thank every person who has contributed to this project: Micha Freidin, Viktor Ivanenko, Piyush Aaryan, Yassine Elboustani, Tasneem Elyamany, Cephars Bonacci, Nolan Wang and Lydia Khelifa Chibout. We would also like to thank Omdena organization for giving us this valuable opportunity.

References

Sean MacAvaney, Hao-Ren Yao, Eugene Yang, Katina Russell, Nazli Goharian, and Ophir Frieder. Hate speech detection: Challenges and solutions. *PloS one*, 14(8):e0221152, 2019.

Mirela P Bogdani, Federico Faloppa, and Xheni Karaj. Beyond definitions. a call for action against hate speech in albania. a comprehensive study november 2021. 2021.

JT Nockleby. hate speech in encyclopedia of the american constitution. electronic journal of academic and special librarianship. 2000.

Imane Guellil, Ahsan Adeel, Faical Azouaou, Mohamed Boubred, Yousra Houichi, and Akram Abdelhaq Moumna. 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*, pages 68–75, 2022.

Thomas Davidson, Dana Warmsley, Michael Macy, and Ingmar Weber. Automated hate speech detection and the problem of offensive language. In *Proceedings of the international AAAI conference on web and social media*, volume 11, pages 512–515, 2017.

Paula Fortuna and Sérgio Nunes. A survey on automatic detection of hate speech in text. *ACM Computing Surveys (CSUR)*, 51(4):1–30, 2018.

Anna Schmidt and Michael Wiegand. A survey on hate speech detection using natural language processing. In *Proceedings of the fifth international workshop on natural language processing for social media*, pages 1–10, 2017.

Delphine Battistelli, Cyril Bruneau, and Valentina Dragos. Building a formal model for hate detection in french corpora. *Procedia Computer Science*, 176:2358–2365, 2020.

Fatimah Alkomah and Xiaogang Ma. A literature review of textual hate speech detection methods and datasets. *Information*, 13(6):273, 2022.

Flor Miriam Plaza-del Arco, M Dolores Molina-González, L Alfonso Urena-López, and M Teresa Martín-Valdivia. Comparing pre-trained language models for spanish hate speech detection. *Expert Systems with Applications*, 166: 114120, 2021.

Nuha Albadi, Maram Kurdi, and Shivakant Mishra. Are they our brothers? analysis and detection of religious hate speech in the arabic twittersphere. In *2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pages 69–76. IEEE, 2018.

Hala Mulki, Hatem Haddad, Chedi Bechikh Ali, and Halima Alshabani. L-hsab: A levantine twitter dataset for hate speech and abusive language. In *Proceedings of the third workshop on abusive language online*, pages 111–118, 2019.

Fréha Mezzoudj, Mourad Loukam, and Fatma Zohra Belkredim. Arabic algerian oranee dialectal language modelling oriented topic. *International Journal of Informatics and Applied Mathematics*, 2(2):1–14, 2019.

Oussama Boucherit and Kheireddine Abainia. Offensive language detection in under-resourced algerian dialectal arabic language. *arXiv preprint arXiv:2203.10024*, 2022.

Djamila Menifi, Wiam Moussa, and Ahmed Cherif Mazari. *Transfer Learning and Deep Learning for Multilingual Algerian Dialect Hate Speech Detection*. PhD thesis, 2022.

Adel Abdelli, Fayçal Guerrouf, Okba Tibermacine, and Belkacem Abdelli. Sentiment analysis of arabic algerian dialect using a supervised method. In *2019 International Conference on Intelligent Systems and Advanced Computing Sciences (ISACS)*, pages 1–6. IEEE, 2019.

Imane Guellil, Ahsan Adeel, Faical Azouaou, Mohamed Boubred, Yousra Houichi, and Akram Abdelhaq Moumna. Sexism detection: The first corpus in algerian dialect with a code-switching in arabic/french and english. *arXiv preprint arXiv:2104.01443*, 2021.

Ona De Gibert, Naiara Perez, Aitor García-Pablos, and Montse Cuadros. Hate speech dataset from a white supremacy forum. *arXiv preprint arXiv:1809.04444*, 2018.

Areej Al-Hassan and Hmood Al-Dossari. Detection of hate speech in social networks: a survey on multilingual corpus. In *6th international conference on computer science and information technology*, volume 10, pages 10–5121, 2019.

Naaima Boudad, Rdouan Faizi, Rachid Oulad Haj Thami, and Raddouane Chiheb. Sentiment analysis in arabic: A review of the literature. *Ain Shams Engineering Journal*, 9(4):2479–2490, 2018.

Nizar Y Habash. *Introduction to Arabic natural language processing*. Springer Nature, 2022.

Abdul-Baquee M Sharaf and Eric Atwell. Qurana: Corpus of the quran annotated with pronominal anaphora. In *Lrec*, pages 130–137, 2012.

Kheireddine Abainia, Kenza Kara, and Tassadit Hamouni. A new corpus and lexicon for offensive tamazight language detection. In *Proceedings of the 7th International Workshop on Social Media World Sensors*, pages 1–6, 2022.

Ahmed Cherif Mazari and Hamza Kheddar. Deep learning-based analysis of algerian dialect dataset targeted hate speech, offensive language and cyberbullying. *International Journal of Computing and Digital Systems*, 2023.

Imane Guellil, Ahsan Adeel, Faical Azouaou, Sara Chennoufi, Hanene Maafi, and Thinhinane Hamitouche. Detecting hate speech against politicians in arabic community on social media. *International Journal of Web Information Systems*, 16(3):295–313, 2020.

Djamila Mohdeb, Meriem Laifa, Fayssal Zerargui, and Omar Benzaoui. Evaluating transfer learning approach for detecting arabic anti-refugee/migrant speech on social media. *Aslib Journal of Information Management*, 74(6): 1070–1088, 2022.

Reem ALBayari and Sherief Abdallah. Instagram-based benchmark dataset for cyberbullying detection in arabic text. *Data*, 7(7):83, 2022.

Monirah A Al-Ajlan and Mourad Ykhlef. Optimized twitter cyberbullying detection based on deep learning. In *2018 21st Saudi Computer Society National Computer Conference (NCC)*, pages 1–5. IEEE, 2018.

Batoul Haidar, Maroun Chamoun, and Ahmed Serhrouchni. Arabic cyberbullying detection: Enhancing performance by using ensemble machine learning. In *2019 international conference on internet of things (ithings) and ieee green computing and communications (greencom) and ieee cyber, physical and social computing (cpscom) and ieee smart data (smartdata)*, pages 323–327. IEEE, 2019.

Hamdy Mubarak, Kareem Darwish, and Walid Magdy. Abusive language detection on arabic social media. In *Proceedings of the first workshop on abusive language online*, pages 52–56, 2017.

Azalden Alakrot, Liam Murray, and Nikola S Nikolov. Dataset construction for the detection of anti-social behaviour in online communication in arabic. *Procedia Computer Science*, 142:174–181, 2018.

Hatem Haddad, Hala Mulki, and Asma Oueslati. T-hsab: A tunisian hate speech and abusive dataset. In *International conference on Arabic language processing*, pages 251–263. Springer, 2019.

Hamdy Mubarak, Ammar Rashed, Kareem Darwish, Younes Samih, and Ahmed Abdelali. Arabic offensive language on twitter: Analysis and experiments. *arXiv preprint arXiv:2004.02192*, 2020.

Hala Mulki and Bilal Ghanem. Let-mi: an arabic levantine twitter dataset for misogynistic language. *arXiv preprint arXiv:2103.10195*, 2021.

Rehab Duwairi, Amena Hayajneh, and Muhannad Quwaider. A deep learning framework for automatic detection of hate speech embedded in arabic tweets. *Arabian Journal for Science and Engineering*, 46:4001–4014, 2021.

Wassen Aldjanabi, Abdelghani Dahou, Mohammed AA Al-qaness, Mohamed Abd Elaziz, Ahmed Mohamed Helmi, and Robertas Damaševičius. Arabic offensive and hate speech detection using a cross-corpora multi-task learning model. In *Informatics*, volume 8, page 69. MDPI, 2021.

Batoul Haidar, Maroun Chamoun, and Ahmed Serhrouchni. A multilingual system for cyberbullying detection: Arabic content detection using machine learning. *Advances in Science, Technology and Engineering Systems Journal*, 2(6): 275–284, 2017.

Ahmed Abdelali, Kareem Darwish, Nadir Durrani, and Hamdy Mubarak. Farasa: A fast and furious segmenter for arabic. In *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: Demonstrations*, pages 11–16, 2016.

Amine Abdaoui, Mohamed Berrimi, Mourad Oussalah, and Abdelouahab Moussaoui. Dziribert: a pre-trained language model for the algerian dialect. *arXiv preprint arXiv:2109.12346*, 2021.

Sourab Mangrulkar, Sylvain Gugger, Lysandre Debut, Younes Belkada, and Sayak Paul. Peft: State-of-the-art parameter-efficient fine-tuning methods. `https://github.com/huggingface/peft`, 2022.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. Text embeddings by weakly-supervised contrastive pre-training. *arXiv preprint arXiv:2212.03533*, 2022.

E Moatez Billah Nagoudi, A Elmadany, and M Abdul-Mageed. Arat5: Text-to-text transformers for arabic language understanding and generation. *arXiv preprint arXiv:2109.12068*, 2021.