# Hate Speech Detection in the Arabic Language: Corpus Design, Construction and Evaluation

Ashraf Ahmad , Mohammad Azzeh , Eman Alnagi , Qasem Abu Al-Hajja * , Dana Halabi , Abdullah Aref , Yousef AbuHour

Posted Date: 7 September 2023

doi: 10.20944/preprints202309.0497.v1

Keywords: Arabic Hate Speech; Natural Language Processing (NLP); Machine Learning; Arabic 18 Hate Speech Detection; Arabic Hate Speech Corpus

*Article*

# Hate Speech Detection in the Arabic Language: Corpus Design, Construction and Evaluation

**Ashraf Ahmad [1], Mohammad Azzeh [2], Eman Alnagi [1], Qasem Abu Al-Haija [3,*], Dana Halabi [1], Abdullah Aref [1] and Yousef AbuHour [4]**

[1] Department of Computer Science, Princess Sumaya University for Technology (PSUT), Amman 11941, Jordan; a.ahmad@psut.edu.jo (A.A.), ema20219005@std.psut.edu.jo (E.A.), d.halabi@psut.edu.jo (D.H.)
[2] Department of Data Science, Princess Sumaya University for Technology (PSUT), Amman 11941, Jordan, m.azzeh@psut.edu.jo (M.A.), a.aref@psut.edu.jo (A.R.)
[3] Department of Cybersecurity, Princess Sumaya University for Technology (PSUT), Amman 11941, Jordan, q.abualhaija@psut.edu.jo (Q.A.A.-H.)
[4] Department of Basic Sciences, Princess Sumaya University for Technology (PSUT), Amman 11941, Jordan, y.abuhour@psut.edu.jo (Y.A.)
* Correspondence: q.abualhaija@psut.edu.jo

**Abstract:** Hate Speech Detection in Arabic presents a multifaceted challenge due to the broad and diverse linguistic terrain. With its multiple dialects and rich cultural subtleties, Arabic requires particular measures to address hate speech online successfully. To address this issue, academics and developers have used natural language processing (NLP) methods and machine learning algorithms adapted to the complexities of Arabic text. However, many proposed methods were hampered by a lack of a comprehensive dataset/corpus of Arabic hate speech. In this research, we propose a novel multi-class public Arabic dataset comprised of 403,688 annotated tweets categorized as extremely positive, positive, neutral, or negative based on the presence of hate speech. Using our developed dataset, we additionally characterize the performance of multiple machine learning models for Hate speech identification in Arabic Jordanian dialect tweets. Specifically, the Word2Vec, TF-IDF, and AraBert text representation models have been applied to produce word vectors. With the help of these models, we can provide classification models with vectors representing text. After that, seven machine learning classifiers have been evaluated: Support Vector Machine (SVM), Logistic Regression (LR), Naive Bays (NB), Random Forest (RF), AdaBoost (Ada), XGBoost (XGB), and CatBoost (CatB). In light of this, the experimental evaluation revealed that, in this challenging and unstructured setting, our gathered and annotated datasets were rather efficient and generated encouraging assessment outcomes. This will enable academics to delve further into this crucial field of study.

**Keywords:** arabic hate speech; natural language processing (NLP); machine learning; arabic hate speech detection; arabic hate speech corpus

## 0. Introduction

In recent years, the spread, diversity, and ease of use of social media platforms (e.g. Facebook, Twitter, etc.) have facilitated the rapid dissemination of information and the quick growth of virtual communities [1]. Social media has changed typical daily routines of individuals, traditional business operations, as well as the interaction patterns within various communities [2]. Despite the benefits that resulted from these advances, individuals and communities became vulnerable to new forms of harm and verbal aggression that were not common before. Hate speech has gained prominence as a form of discourse that targets individuals or groups on the basis of race, religion, gender, sexual orientation, or other characteristics [3]. The number of content items on which Facebook took action due to hate speech worldwide between the 4th quarter of 2017 and the 1st quarter of 2023 is presented in Figure 1. Despite the decrease in numbers as governments worldwide relaxed COVID-19-related constraints, the number in the first quarter of 2023 is higher than the corresponding interval of 2020 and more than double the number of the corresponding interval of 2019 [4].

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 7 September 2023 doi:10.20944/preprints202309.0497.v1



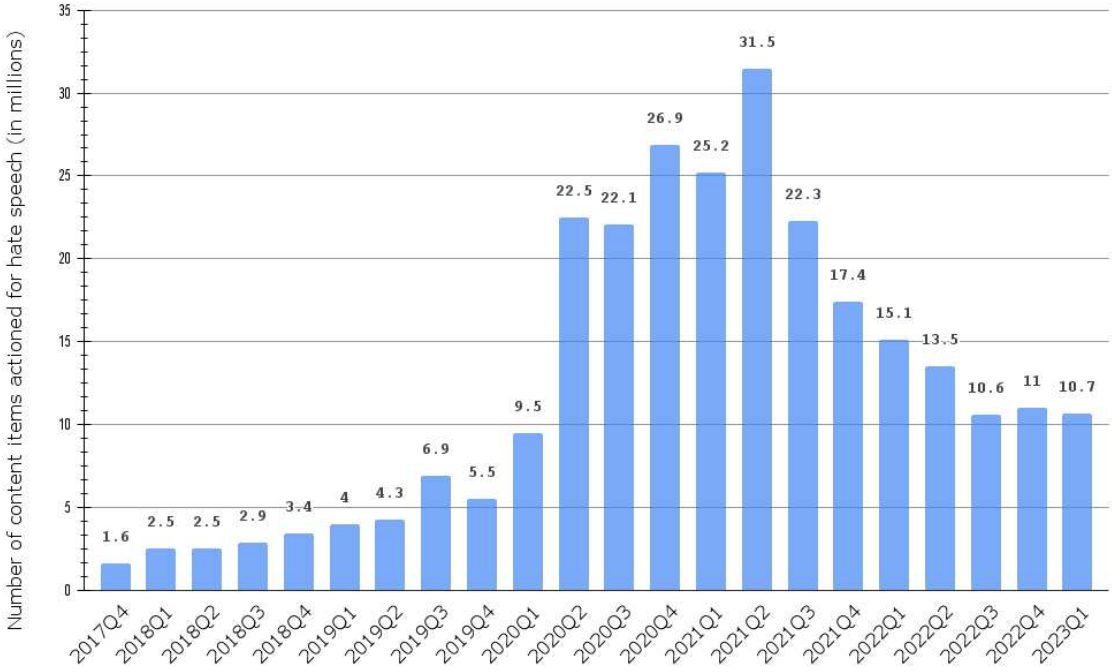


**Figure 1.** Number of content items actioned for hate speech on Facebook worldwide between 4th quarter 2017 and 1st quarter 2023.

There are different forms of hate speech, such as harassment, cyberbullying, offense, and abuse [5,6]. Harassment refers to persistent unwanted behavior that causes distress or fear, often involving repetitive and intrusive actions [7]. Cyberbullying specifically occurs in the digital realm, encompassing the use of technology to harass, intimidate, or demean others [8]. Offense refers to actions or expressions that cause displeasure or upset, while abuse involves the misuse of power or authority to harm or control others [9].

Many recent studies have shown the relationship of hate speech to the increase in hate crimes worldwide [10]. It also showed its connection to the exposure of targeted individuals to discrimination, violations, and denial of their human rights [11].

Social media can be very helpful for connecting people, increase self-esteem [12], as well as being a platform for information exchange and self-expression [13]. Other use of social media in societies includes, but not limited to, building communities, and help in emergencies [14]. On the other hand, social media may have negative impact on mental health as it may lead to stress, increased sadness and isolation [12], and addiction , as well as the possibility of having a negative impact of privacy and security, facilitating fraud [14], the spread of misinformation and hate speech. Social media has been used successfully in business for marketing and identifying and engaging the talents [14]. Other use of social media in business includes, but not limited to, customer support, facilitate communications between employees, and employee self development [13]. Furthermore, social media found to have a positive value in higher education particularly as teaching and learning tool [15]; it can increase peers' interactivity, and online knowledge sharing behaviour which has a positive impact on students' engagement that can lead better academic performance [16]. Also, the use of social media found to increase e-learning platform acceptance by students [17].

The propagation of hate speech online continuously challenges both policy-makers and research community; due to difficulties in limiting the evolving cyberspace, the need to empower individuals to express their opinions, and the delay of manual checking [18] .

In an effort to reduce its risks and possible devastating effects on the lives of individuals, families, and communities, the NLP community has shown an increasing interest in developing tools that help

 doi:10.20944/preprints202309.0497.v1



in the automatic detection of hate speech on social media platforms [19] as the detection of hate speech can be, generally, modeled as a supervised learning problem [20]. Several studies investigated the problem and contrasted various processing pipelines using varoious sets of features, classification algorithms (e.g., Naives Bayes, Support Vector Machine (SVM), deep learning architectures, and so on) [18].

Fairly generic features, such as bag of words or embeddings, found to result in a reasonable classification performance , and character-level schemes found to outperform token-level approaches [20]. It is reported in the literature that even though information derived from text can be useful for detecting hate speech, it may be beneficial to use some meta-information or information from other media types (e.g. images attached to messages) [18].

While several studies in the literature investigated anti-social behaviors such as, abusive or offensive language cyberbullying and heat speech; a limited number of researches have contributed to hate speech detection in Arabic in general [21]. At the time of writing, we are not aware of any study that attempts to detect the hate speech for the specif Arabic dialect used in Jordan. Compared to English, Arabic is considerably an under-resourced language when it comes to NLP. The existence of different dialects combined with the richness and complexity of Arabic morphology add up more challenges to Arabic NLP research [22].

The main contributions of this work are as follows:

1. Construct a public Arabic-Jordanian dataset of 403,688 annotated tweets labeled according to the appearance of hate speech as very positive, positive, neutral, and negative.
2. Comparing the performances of machine learning models for Hate speech detection of Arabic Jordanian dialect tweets.

The remainder of this paper is organized as follows. The related work to Jordanian dialect datasets and Arabic Hate speech detection are reviewed in Section 1. Section 2 details our methodology for constructing the new dataset, the preprocessing steps, and statistics. Section 3 describes in detail the architecture of classification models, the conducted experiments, and the results. Section **??**, discusses and analyzes the results. Finally, Section 4 concludes our work and discusses future directions.

## 1. Literature Review

With the rapid spread of social media platforms, the freedom level has been elevated so that many people can give their opinions with advice or criticism without borders. People with shy and conservative personalities have been given the chance to speak up and give their opinions with no fear of interruption or hesitation. The problem is that many people have abused this freedom by not taking into consideration the courtesy of speech and descent manners. Hate speech, including cyberbullying, offensive talk, sarcasm, and harassment, are just a few examples of freedom abuse on social media [5].

This problem has motivated researchers to create methods to detect and stop such violations that have a large negative influence on our societies and youth and children in particular. In this section, selective literature is introduced and discussed to illustrate the methods conducted in this area.

Surely, the problem of hate speech has been considered in several scopes: science, sociology, psychology, and even criminology. This research will concentrate on the technical efforts conducted in this area, regarding Natural Language Processing (NLP) and Artificial Intelligence (AI), to detect such behavior.

### *1.1. Hate Speech and Related Concepts*

NLP is one of the common disciplines that is needed in the area of hate speech detection. Posts, tweets, comments, reviews, and most social contributions on social media are inserted as text. People from all over the world can express their feelings with their language and even dialect. No language standards are enforced on such platforms, and thus, NLP tools have become essential in representing, understanding, and analyzing these inputs.

AI algorithms, either Machine Learning (ML) or Deep Learning (DL) algorithms, have been extensively conducted as classification algorithms to detect hate speech in text extracted from social media [19,23].

The most vital issue when tackling this problem is to work on a high-quality hate speech corpora. In literature, two streams are taken into consideration. Many researchers tend to use public corpora that are directed to hate speech in general or in a certain type of hate. Such corpora can be hard to find, especially in low-resources languages. Thus, most of the literature that adopts this stream works on English corpora as in [24–26]. Public Arabic hate speech corpora can also be found but rarely concentrate on certain Arabic dialects. [27–29], for example, have used in their research the OffensEval 2020 dataset, which shared task competition organizers have provided. In another case, authors of [30] have proposed an Arabic hate speech corpus that they reused in further experiments in [31]. Sections 1.2 and 1.3 highlight literature that created hate speech corpora in different languages.

The next issue in this problem is representing the text (posts, tweets, comments, etc.) in proper text presentation (word embedding technique), enabling AI classifiers to handle them as proper inputs and thus produce the desired outputs. Any NLP task needs such text presentation methods. In literature, several word embedding techniques are used and, in some cases, compared in the same paper. Examples of such techniques are TF-IDF [28], word2vec and some of its variations such as AraVec [32–34] , and Fasttext [30–33].

Using these corpora to detect hate speech on social media platforms is a classification problem that needs labeled data. Labeling of each text sample should be applied using either manual or automatic annotation processes. Number of classes varies from one research to another. Many papers use the binary classes by only labeling the samples with two labels. Hate or Not hate is the most common binary label used in literature, such as [5,32,33,35–37]. Others used different labels for binary classification, such as clean or offensive [30,38], hateful or normal [39]. Some researchers were more precise in identifying the labels according to the type of hate speech detected. For example, in [40], Islamic Radicalism is the type of hate to detect, and thus the binary labels are extremist or non-extremist. In [8], the target was to detect whether cyberbullying terms exist in Facebook posts; thus, the binary labels used were cyberbullying or non-cyberbullying.

Three-labeled corpora have been introduced, with a third label that either indicates a neutral label [34] or undecided [41]. Also, the three labels have been used to distinguish between hate and abusive classes in addition to clean or normal, as in [30,42,43].

Other research used multi-labeled corpora that include fine-grained labels that identify the type of hate detected more specifically. In [44], the arHate Dateset has been created, with labels: racism, against religion, gender inequality, violence, offensive, and bullying. These labels have been selected to distinguish the precise type of hate speech. Additional labels were added to indicate the existence of hate speech other than the ones mentioned previously, using labels normal positive and normal negative. Other examples are [30,45–47], which used multi-labels to annotate their corpora, that in some cases reached eight different labels, according to how many details desired to be expressed in the labels.

Several classifiers have been used in the literature to apply the classification task. Most research compared different models to find the most proper one(s) for the created or the public corpora tested.

The classifiers used were categorized into ML and DL algorithms. In [28], both ML and DL classifiers have been applied and compared. Fifteen traditional ML classifiers, such as SVM, RF, XGBoost, DT, LR, etc., have been used. On the other hand, DL classifiers have been used, such as CNN and RNN. When compared, it has been found that the best classifier was the hybrid CNN and RNN classifier.

In [48], SVM and LR have been used and compared with a BERT-based model where the BERT model yielded the best results. In their research, a novel approach has been conducted by including emojis found in the tweets in the hate speech detection.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 7 September 2023 doi:10.20944/preprints202309.0497.v1



Authors of [25] used the BERT model to propose their own model. They added extra layers on BERT, consisting of CNN and LSTM.

The ensemble concept has not been neglected in research since in [31], the authors have created an ensemble model consisting of CNN and BiLSTM classifiers. They have compared this ensemble model, with other individual models, and it outperformed the others.

In Sections 1.2 and 1.3, research that proposed hate speech corpora is discussed and illustrated.

### *1.2. Arabic Hate Speech corpora and detection Systems*

Arabic, as a low-resource language, lacks the availability of specialized hate speech corpora. As aforementioned, research has been found and discussed in the previous sub-section, highlighting some research involved in this area. Nevertheless, Arabic dialects' hate speech datasets are not easily found in the literature.

In this section, a sample of research that created Arabic hate speech corpora is discussed. Table 1 summarizes the main aspects of this sample.

Social media platforms have been considered the sanctuary of different types of people in society to express their feelings. Many people post their social news and events, either happy or sad, to the public. Nevertheless, this publicity can encourage some indecent people to reflect their negative feelings of hate, sarcasm, bullying, and others. Thus, social media platforms are considered the main resources of datasets, corpora, that consist of samples that can be trained and tested for the hate speech detection task.

In the literature, it has been found that Facebook [5,44,45], Twitter, Instagram and YouTube [5] are some of the main sources of such data. As illustrated in Table 1, most of the research used Twitter as the social media source; this indicates that this platform provides the data more easily to researchers than other platforms, such as Facebook. Another reason of researchers prefer to collect data from Twitter is that tweets mostly consist of short text. While other platforms, such as Instagram or YouTube, consist of data in the form of images and videos, which is harder to process. Also, some platforms, such as Facebook, Telegram, and Reddit, may have text content, but in most cases, the text is long and can take longer time to process.

The Arabic language has many challenges when processed and tackled. Yet, standard Arabic has its rules and grammar that can make the text understanding and analysis easier. Arabic dialects, on the other hand, propose a hard problem for AI to distinguish and understand. Thus, collecting Arabic dialect data has been a hot research topic that Arabian authors have considered when conducting NLP tasks, specifically hate speech detection.

As illustrated in Table 1, many researchers collected data that use Arabic letters without concentration on dialects, such as [5,32,34,35,41]. In other cases, researchers concentrated on certain dialects that refer to a certain region or country within the Arabian countries. This helps researchers, when scraping social media, to search for keywords that are more related to this dialect. Levantine [43] and Gulf [30,38] are examples of dialects used by people in a wide region of the Arab world. So, when a researcher needs to collect data in the Levantine dialect, for example, they should add to their query the desired locations, including Jordan, Palestine, Syria, and Lebanon. If a researcher concentrates on a certain country, the location query only includes this country. Saudi [36,42], Tunisian [39], and Egyptian [45] are examples of such dialects. As for the Jordanian dialect, our work is considered the first to tackle data in this dialect, as far as we know.

Other query questions are heavily used by researchers when scraping social media platforms, is the period of time. This can allow the researchers to study public opinions in a period of time when certain political or social events have happened.

**Table 1.** Summary of Literature on Arabic Hate Speech corpora.

| Ref# | Dialect | Source | Dataset Size | Labeling Process | Classes | Best Classifier | Results |
|---|---|---|---|---|---|---|---|
| [5], 2020 | Mixed | Facebook, Twitter, Instagram, YouTube | 20,000 | Manual | Hate, Not hate | RNN | Acc: 98.7%, F1-score: 98.7%, Recall: 98.7%, Precision: 98.7% |
| [42], 2020 | Saudi | Twitter | 9,316 | Manual | hateful, abusive, or normal | CNN | Acc: 83%, F1-score: 79%, Recall: 78%, Precision: 81%, AUC: 79% |
| [41], 2021 (AraCOVID19-MFH) | Mixed | Twitter | 10,828 | Manual | Yes, No, Cannot decide | arabert Cov19 | F1-score: 98,58% |
| [43], 2021 (ArHS) | Levantine | Twitter | 9,833 | Manual | Hate or Normal; Hate, Abusive or Normal | Binary Class: CNN, Ternary class: BiLSTM-CNN and CNN, Multi-class: CNN-LSTM and the BiLSTM-CNN | F1-score: 81%, F1-scode: 74%, F1-score: 56% |
| [34], 2020 | Mixed | Twitter | 3,696 | | Hate, Neutral, or normal | LSTM+ CNN, with word embedding Aravec (N-grams and skip grams) | F1-score: 71.68% |
| [44], 2022 (arHateDataset) | mixed | Variaty | 4,203 | | racism, Against religion, gender inequality, violence, offense, bullying, normal positive and normal negative | RNN architectures: DRNN-1: binary classification, DRNN-2: multi-labelled classification | Validation accuracy: 83.22%, 90.30% |

**Table 1.** *Cont.*

| Ref# | Dialect | Source | Dataset Size | Labeling Process | Classes | Best Classifier | Results |
|---|---|---|---|---|---|---|---|
| [5], 2020 | Mixed | Facebook, Twitter, Instagram, YouTube | 20,000 | Manual | Hate, Not hate | RNN | Acc: 98.7%, F1-score: 98.7%, Recall: 98.7%, Precision: 98.7% |
| [35], 2023 | Mixed | Twitter (public datasets) | 34,107 | Unifying annotation in datasets | Hate, no hate | AraBERT | Accuracy: 93% |
| [38], 2021 | Standard and Gulf | Twitter | Training: 9,345, Unlabelled: 5M, Testing: 4,002 | Semi-supervised Learning | Clean, or offensive | CNN + Skip gram | F1-score: 88.59%, Recall: 89.60%, Precision: 87.69% |
| [32], 2020 | Mixed | Twitter | 3,232 | Manual | Hate or Not hate | CNN-FastText | Acc: 71%, F1-score: 52%, Recall: 69%, Precision: 42% |
| [36], 2020 | Saudi | Twitter | 9,316 | | Hate or Not hate | CNN | Acc: 83%, F1-score: 79%, Recall: 78%, Precision: 81% |
| [39], 2022 | Tunisian | Twitter | 10,000 | Manual | Hateful or Normal | AraBERT | F1-Score: 99% |
| [30], 2020 | Gulf | Twitter | 5,361 | Manual | 2-classes: Clean or Offensive/Hate, 3-classes: Clean, Offensive or Hate, 6-classes: Clean, Offensive, Religious Hate, Gender Hate, Nationality Hate or Ethnicity Hate | CNN + mBERT | 2-classes: 87.03 %, 3-classes: 78.99%, 6-classes: 75.51% |
| [40], 2022 | Mixed | Twitter | 3,000 | Manual | Extremist or Non-extremist | SVM | Acc: 92%, F1-score: 92%, Recall: 95%, Precision: 89% |

 doi:10.20944/preprints202309.0497.v1



Since scraping the social media platforms and annotating them with proper labels is not easy, it can be noticed that the size of such corpora is not considered large. Most corpora listed in Table 1 have sizes less than 10,000 annotated text, while only three exceeded this number. Thus, collecting and annotating over 400,000 tweets in our work can be considered a vital contribution compared to other corpora proposed in the literature.

To evaluate the collected corpora, researchers have conducted hate speech detection algorithms on them. It can be noticed how most literature have concentrated on using DL algorithms, especially RNN and its variations; LSTM and GRU. In addition to DL transformer-based models, such as BERT, AraBERT, mBERT and others. This refers to the special features of text data over other types of data such as tabular ones. Extracting features of text depend on the relationships and association between words in the same text, and not necessarily adjacent words. Thus, such classifiers can capture such features more efficiently than others. Nevertheless, ML classifiers have proven to be efficient, in case of using proper word embedding techniques to represent and extract the important features from a text. Table 1, also summarizes the best classifiers used in the literature and their results.

### *1.3. Hate Speech Datasets for other Languages*

As aforementioned, many English hate speech corpora have been created to conduct the hate speech detection task. Recent surveys review such literature which proposed high-quality hate speech corpora that can be used by researchers for further investigation [23,49]. Nevertheless, low-resource language corpora other than Arabic can be hard to find.

Table 2 illustrates information about previous research that collected corpora in low-resource languages, such as Turkish [8,46], Kurdish [37], and Bangali [33].

Sizes, labeling process, classes, and best classifiers are displayed in the table, summarizing the important aspects of this literature.

It is worth mentioning that in high-resource languages, such as English, the researchers tend to concentrate on fine-grained classes that distinguish the types of hate speech. Since the number of keywords indicating these classes can be classified more easily. Consequently, this enables the researchers to create complex classifiers that consist of multiple layers that may yield high performance [47].

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 7 September 2023 doi:10.20944/preprints202309.0497.v1



**Table 2.** Summary of Selected Related Literature.

| Ref# | Language | Source | Dataset Size | Labeling Process | Classes | Best Classifier | Results |
|---|---|---|---|---|---|---|---|
| [8], 2023 | Turkish | Facebook | 5,000 | Manual | Cyberbullying or Non-Cyberbullying (balanced) | BERT | F1-score: 92.8% |
| [33], 2021 | Bengali | YouTube and Facebook Comments | 30,000 | Manual | Hate or Not hate | SVM | Acc: 87.5% |
| [46], 2022 | Turkish | Twitter | Istanbul Convention Dataset (1,033), Refugee Dataset (1,278) | Manual | 5-classes: No Hate speech, Insult, Exclusion, Wishing Harm or Threatening Harm | BERTurk | F1-score: Istanbul Convention Dataset (71.52%), Refugee Dataset (72.34%) |
| [37], 2022 | Kurdish | Facebook Comments | 6,882 | Manual | Hate or Not hate | SVM | F1-score: 68.7% |
| [47], 2022 (ETHOS) | English | YouTube and Reddit comments | Binary (998), Multi-label (433) | Auto and Manual | 8-classes: violence, directed vs generalized, gender, race, national origin, disability, sexual orientation, or religion | Binary: DistilBERT, Multi-label: NNBR (BiLSTM + Attention Layer + FF layer | F1-score: Binary: 79.92%, Multi-label: 75.05% |

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 7 September 2023 doi:10.20944/preprints202309.0497.v1



## 2. Methodology

This section details the comprehensive methodology used to construct, annotate, and evaluate the Jordanian Hate Speech Corpus (JHSC) for detecting hate speech focused on the Jordanian dialect. Our approach includes rigorous data collection, careful pre-processing, manual annotation, exploratory data analysis, and performance assessment using hate speech detection models. The methodology used reflects the robustness, reliability, and applicability of the JSSC in developing research analyzing hate speech in the context of the Arabic language and dialects. Figure 2 illustrates the general methodology used to create JSSC and model hate speech based on it.

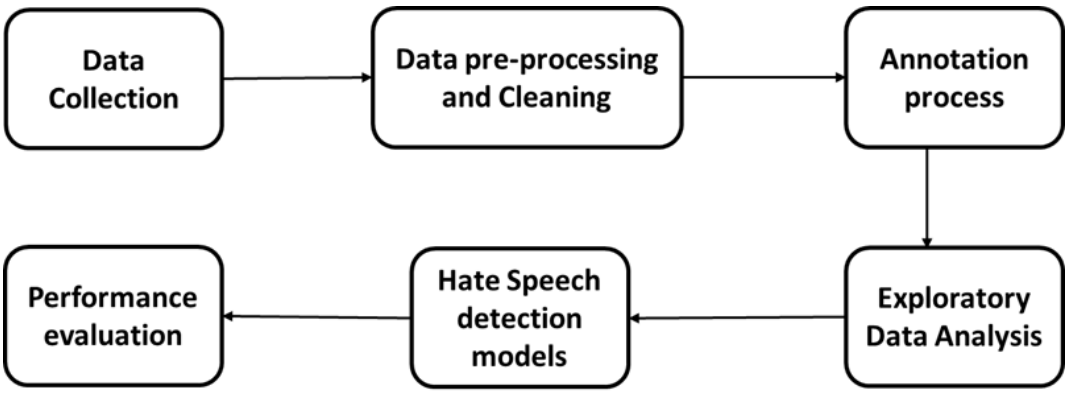


**Figure 2.** Methodology for creating JHSC and model hate speech based on it.

### *2.1. Data Collection*

The initial phase of constructing the Jordanian Hate Speech Corpus (JHSC) involved the collection of Arabic Jordanian dialect tweets from the Twitter platform. The tweets were collected from the beginning of 2014 to the end of 2022. To ensure the authenticity of the collected data, the following steps were applied:

1. Language Filter: The search parameters were further refined by specifying the Arabic language, ensuring that only tweets written in Arabic were retrieved.
2. Location Filter: After scrapping a random sample of tweets, it was found that most tweets do not have the location field that was supposed to be populated in users' profiles. To overcome this issue, Twitter's advanced search techniques were used to include location-based filters. The "search" techniques focused on Jordan's main cities and regions which cover 12 governorates of Jordan and include 20 cities and regions as listed in Table 3.
3. Systematic temporal approach: The data collection process was organized over a period of time extending from the beginning of 2014 to the end of 2022. A monthly segmentation strategy was adopted where tweets for each year were extracted individually and systematically on a month-by-month basis. This approach ensured the stability of the scrapping process and the systematic accumulation of a large number of tweets spread over a longer period of time. Subsequently, the distinct groups from each month and year were combined into one data set. The initial data set contained 2,034,005 tweets in the Jordanian Arabic dialect.

**Table 3.** Jordan's main cities and regions.

| | | | | |
|---|---|---|---|---|
| Abu Alanda | Ajloun | Al Karak | Al Mafraq | Al salt |
| Amman | Aqaba | Baqaa | Irbid | Jarash |
| Karak | Maan | Madaba | Mafraq | Mutah |
| Ramtha | Salt | Tafilah | Zarqa | Marka |

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 7 September 2023 doi:10.20944/preprints202309.0497.v1



### 2.2. Data pre-processing and Cleaning

To reduce noise in the data, several steps have been performed to clean and process the dataset, data pre-processing and cleaning steps are illustrated in Figure 3. First, all duplicate and "retweeted" tweets were deleted, as recommended by [50,51]. Next, the non-Arabic tweets were removed from the dataset since we focused on Arabic Jordanian tweets. Then, unnecessary tokens such as user tags, numbers, emails, URLs, HTML tags, and hashtags were removed because they might reduce the performance of the classifier ([52,53]).

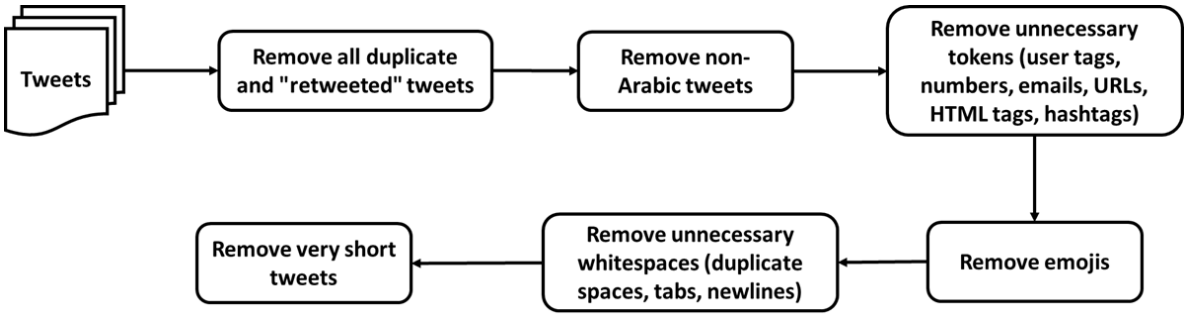


**Figure 3.** Data pre-processing and Cleaning steps.

Although emoji show feelings, they were removed from the dataset, because keeping the emoticons in the dialect Arabic tweets reduces the performance of the classifier ([52,53]), and this is due to the way Arabic sentences are written from right to left, which leads to the reversal of emoticons, as well as due to misunderstanding between brackets in the quote and in emoticons. After that, all whitespaces, such as duplicate spaces, tabs, and newlines, were removed from the dataset.

Finally, the very short tweets with two or less words were removed from the data set. It is worth mentioning that the stemming algorithms were not applied to the dataset because they do not work well with Arabic dialect words [53]. After applying the pre-processing and cleaning steps, the dataset now has 1,824,220 tweets.

### 2.3. Data Annotation

The annotation process is a pivotal stage in the creation of the Jordanian Hate Speech Corpus (JHSC). It includes careful manual tagging of each tweet with sentiment categories specifically geared toward identifying instances of hate speech. This process contributes to the development and evaluation of hate speech detection models.

#### 2.3.1. Annotation process stages

The process of Annotation tweets was done in two stages: lexicon-based annotation stage and manual annotation stage.

- Stage one - Lexicon-based annotation

In this stage, an Arabic hate lexicon from related research was used. This lexicon contains 357 terms that are considered hate or offensive terms [54], a sample from the lexicon term listed in Table 4. In this stage, all tweets that contain any term from this lexicon were extracted to a separate sub-dataset. The new sub-dataset contains 557,551 tweets which is around 30% of the original dataset. The new sub-dataset was then processed through stage 2 of annotation.

**Table 4.** Sample from bad-words lexicon.

| | | | | | |
|---|---|---|---|---|---|
| اباد | طرد | نجس | جاهل | ابليس | حوثي |
| بقر | كلب | حرق | دواعش | حريم | خنزير |

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 7 September 2023 doi:10.20944/preprints202309.0497.v1



- Stage two - Manual annotation stage

In this stage, the sub-dataset was labeled with four labels for sentiment: negative, neutral, positive, and very positive. The meaning and examples of each label are mentioned in Table 5.

The manual annotation process is designed to ensure accuracy and agreement between annotators. This stage was performed through the following tasks:

- Task one - Annotation Guidelines
  To enhance the reliability of annotations, a comprehensive annotation guideline was established. This guideline outlined specific criteria and linguistic indicators for each hate speech class, guiding annotators toward consistent and accurate labeling decisions.

- Task two - Hiring annotators team
  The sub-dataset was manually annotated for hate speech by a team of annotators. Figure 4, illustrates the steps conducted to perform this process.

**Table 5.** Tweets Samples.

| Label Meaning | Tweet | Tweet Meaning | Label |
|---|---|---|---|
| Tweets that have a clear indicator that the opinion is positive | يا نساء العالم كل التحية والاحترام بهذا اليوم ولاكن نكن كل الاحترام والتقدير إلى تاج نساء العالم نساء العربي | This tweet contains respect and appreciation for women all over the world and Jordanian women in specific | negative |
| Tweets that are not offensive or hateful. | بضل الواحد يخطط شهر انه كيف بده يدرس المادة | This tweet is written by a student talking about his study plans | neutral |
| Tweets that are offensive but do not contain hateful content. | اطلع يا كلب يا ابن الكلب يا حاميه يا قتله هاي | This tweet contains bad words (swearing) | positive |
| Tweets that contain hateful content directed at a specific group of people. | غريم الاردنيين الداعشي الكلب أبو بلال التونسي | This tweet contains bad words (swearing) that target specific names for known terrorists, and violent words such as murder | very positive |

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 7 September 2023 doi:10.20944/preprints202309.0497.v1



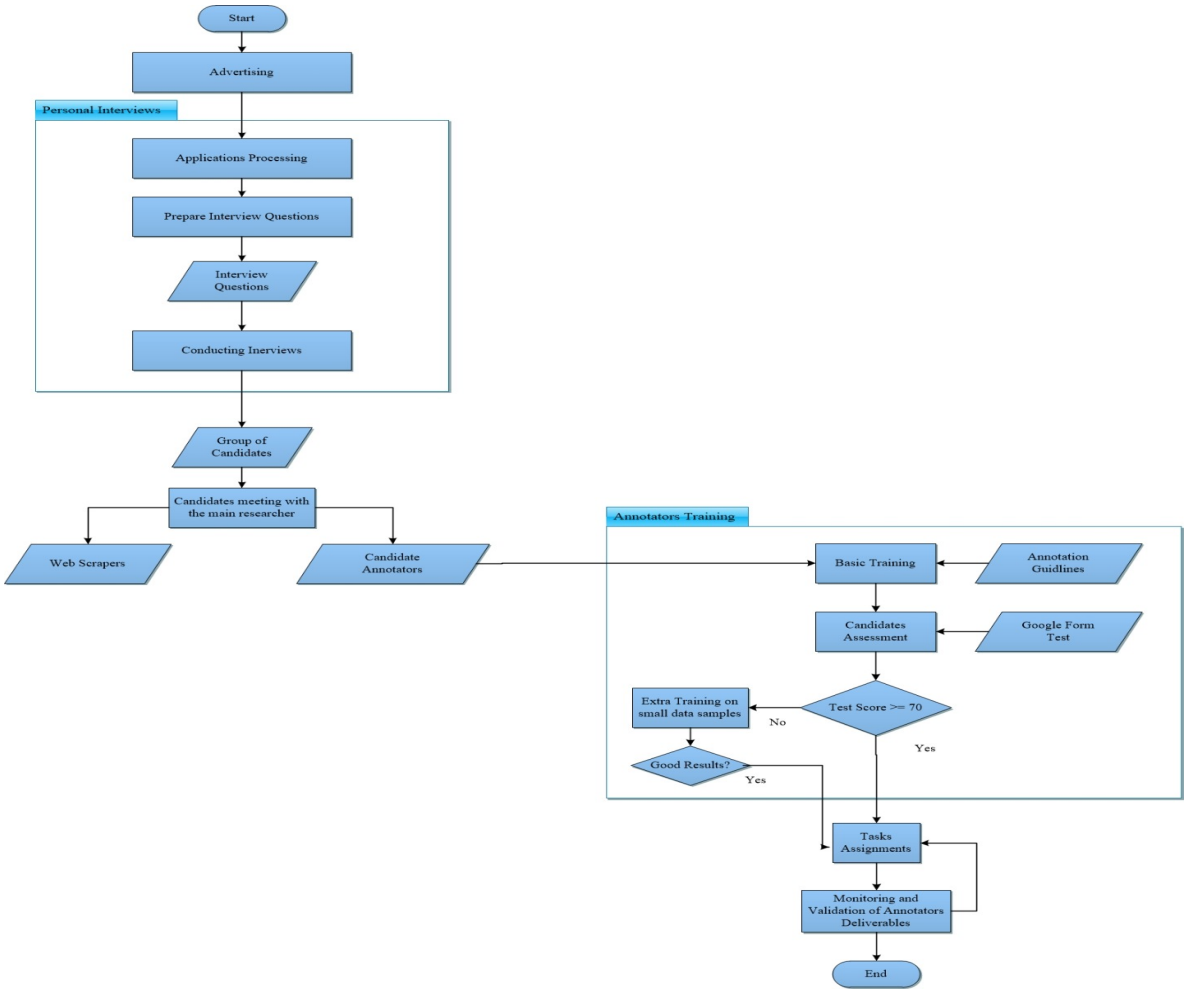


**Figure 4.** Annotation Team Selection.

The process started with an advertisement that has been published on LinkedIn. The purpose was to find qualified personnel, mainly students who could participate in the scraping and annotation part of the project. Figure 5 displays a screenshot of this advertisement.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 7 September 2023 doi:10.20944/preprints202309.0497.v1



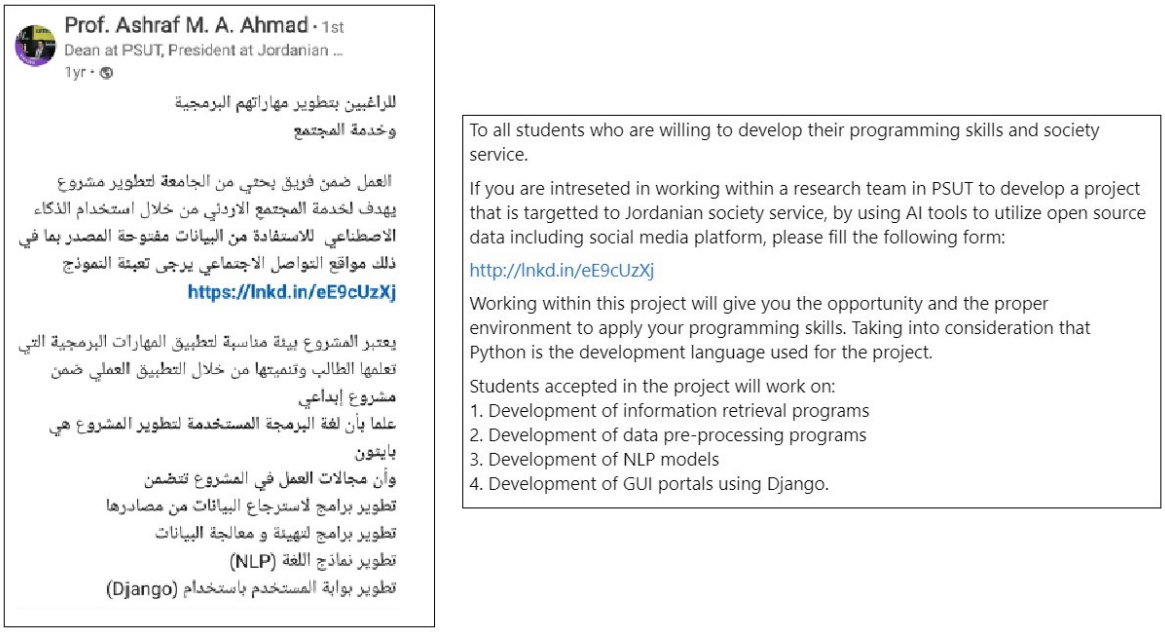


**Figure 5.** Announcement for building Annotators team.

The filled applications have been reviewed and thirty applicants have been interviewed. Twenty of them were selected after passing these interviews. A meeting has been conducted by the main researcher with the interviewed annotators, to clarify the project requirements and the expectations from their side. Part of the candidates have been directed to work on social media scraping, while the others took a quick training to understand the annotation task required. Before starting the annotation task, the candidates took a test that assessed their understanding of the annotation guidelines presented to them during the training and the annotation process. A link to the test is provided in [55]. Candidates who passed the test, with a score of 70) have proceeded with the annotation process. As a start, to validate the annotation guidelines, the annotators, who were native Jordanian Arabic speakers, participated in the following phases:

1. The annotators were given a training set of 100 tweets annotated by a human expert.
2. The annotators then independently applied the guidelines to another test set of 100 tweets.
3. The annotators' annotations were compared with the annotations of the experts. Differences were addressed through discussion. The guidelines have also been modified as necessary.

In addition to the above, the annotators were monitored closely during the annotation process to ensure the quality of the sub-dataset. The inter-annotator agreement is computed to confirm the quality.

### 2.3.2. Inter-annotator agreement

Inter-annotation agreement (IAA) is a measure of how well many annotators can make the same annotation decision on the same data when doing the annotation task independently. It is an important metric in many natural language processing (NLP) tasks, such as text classification, sentiment analysis, and named entity recognition. Fleiss' kappa is an IAA statistical measure that takes into account the number of annotators and the number of classes. Fleiss' kappa was computed from a sample of 500 tweets annotated by three annotators to choose one of four classes: positive, neutral, negative, and very negative. The kappa rate was 0.60 indicating that there was a moderate level of agreement between the annotators [56].

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 7 September 2023 doi:10.20944/preprints202309.0497.v1



*2.4. Exploratory Data Analysis*

The cleaned corpus has 1,824,220 tweets. Figure 6 shows the distribution of the number of tweets in the years from 2014 to 2022. It is worth mentioning that Figure 1 shows the turnout of Jordanians on Twitter in 2014, then how it decreased by half in the following year. It is also possible to note the relative stability in the past five years, despite the Corona epidemic that swept the world between 2019 and 2022. In general, it is known that the epidemic increased the percentage of participation on social media platforms, but the reason for the decline in the participation rate of Jordanians on Twitter can be attributed to the presence of other platforms for social communication, in addition to the tightening of penalties in the cybercrime law in Jordan.

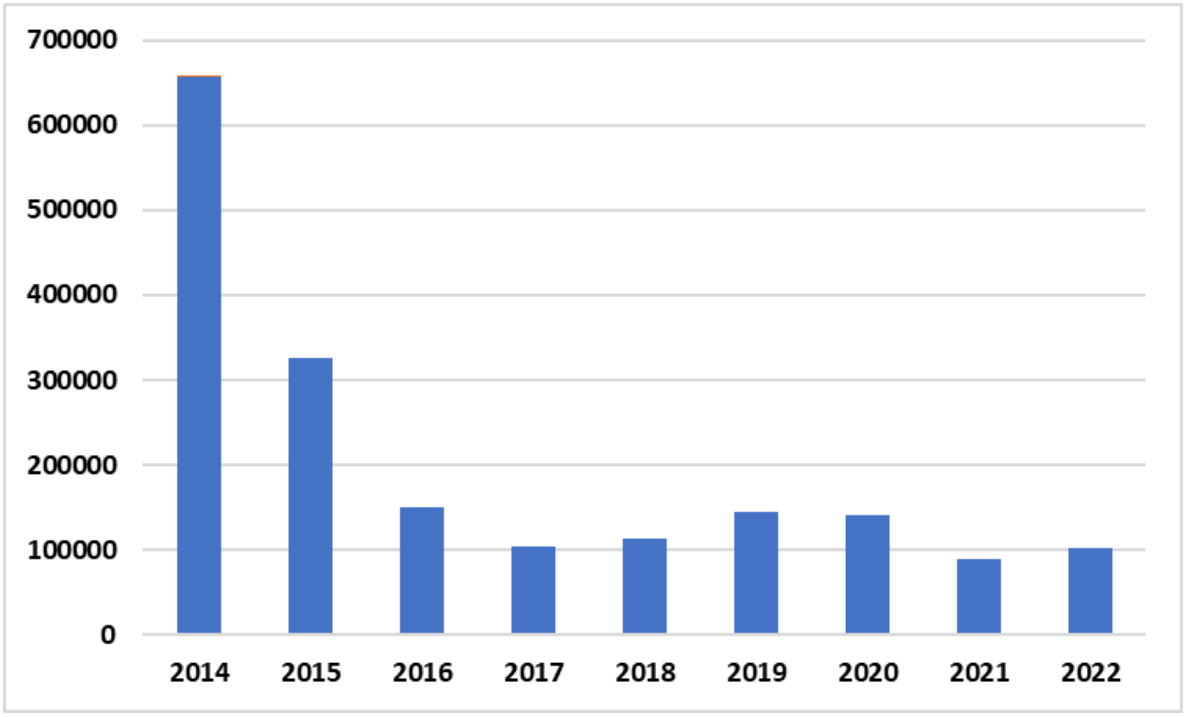


**Figure 6.** Distribution of collected tweets throughout the years 2014 to 2022.

The corpus is partitioned into two parts. Part one has 1,266,669 tweets, and it will be used to build the Jordanian dialect language model. Part two has 557,551 tweets used to construct the Hate speech Jordanian tweets dataset. Currently, this dataset consists of 403,688 annotated tweets, while the remainder is still undergoing the annotation process. Indeed, the dataset has 149,706 positive tweets, 126,297 offensive tweets, 7,034 very offensive tweets, and 120,651 neutral tweets.

*2.5. Feature Engineering and Model Construction*

2.5.1. Text Representation

Building Hate detection system based on machine learning and deep learning requires numerical input features. Converting words into numbers allows machines to perceive and decode linguistic patterns, which is fundamental in most NLP jobs. This process is referred to as text representation. Even if it is an iterative process, this one is crucial for selecting the features of any machine learning model or algorithm. Therefore, the input text must be first transformed to numerical features that can easily fit to machine learning algorithms.

Text representation can be mainly divided into three sections: discrete text representation, distributed text representation, and advanced language model, as shown in Figure 7. Under each category of text representation, there are various of techniques. In this paper we focus on three popular

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 7 September 2023 doi:10.20944/preprints202309.0497.v1



techniques: Term Frequency-Inverse Document Frequency (TF-IDF) [ ], Word2Vec [ ], and BERT text representation.

The idea behind TF-IDF is that each word's weight is determined by a word's frequency and how specific word is frequent in the whole corpus. It takes the count vectorizer (TF) and multiplies it by the IDF score. The resultant output weights for the words are low for very highly frequent words like stop-words. One of the advantages of TF-IDF is it simple and easy to understand, implement, but unfortunately, TF-IDF cannot capture the positional information of the word, and it is highly dependent on the corpus.

Word2Vec is a word embedding model that generates a vector representation of a word [ ]. Each word is represented by a defined vector size that captures its semantic and syntactic relationships with other words. The architecture of word2vec simply consists of input layer, one single hidden layer network, and output layer. The purpose of the network is to learn the word embedding vector for each word by learning the embedding and context weight matrices. There are two versions of Word2Vec: Continuous Bag of Words (CBOW), which is an efficient way to use for a small dataset; the main idea behind it is to predict the middle word in the context of surrounding words. Skip-Gram, in contrast to CBOW, predicts the surrounding context words from a single word, and it is suitable for large corpus but takes more training time as shown in Figure 2.3 [2]. The most important feature of word2vec is its ability to capture the relationships between words in terms of their syntactic and semantic relationships, but it struggling and does not do well with out-of-vocabulary words.

Most recently, advanced text representation techniques have been proposed based on the notion of deep contextualized text representation which allows the generated word vectors to capture the semantic meaning of the word in the text. The emerge of Transformer and Attention model has speed up the presence of advanced text representation such as BERT variants and GPT variants models. In this paper we used a version of BERT model that was trained over a large corpus of Arabic Language.

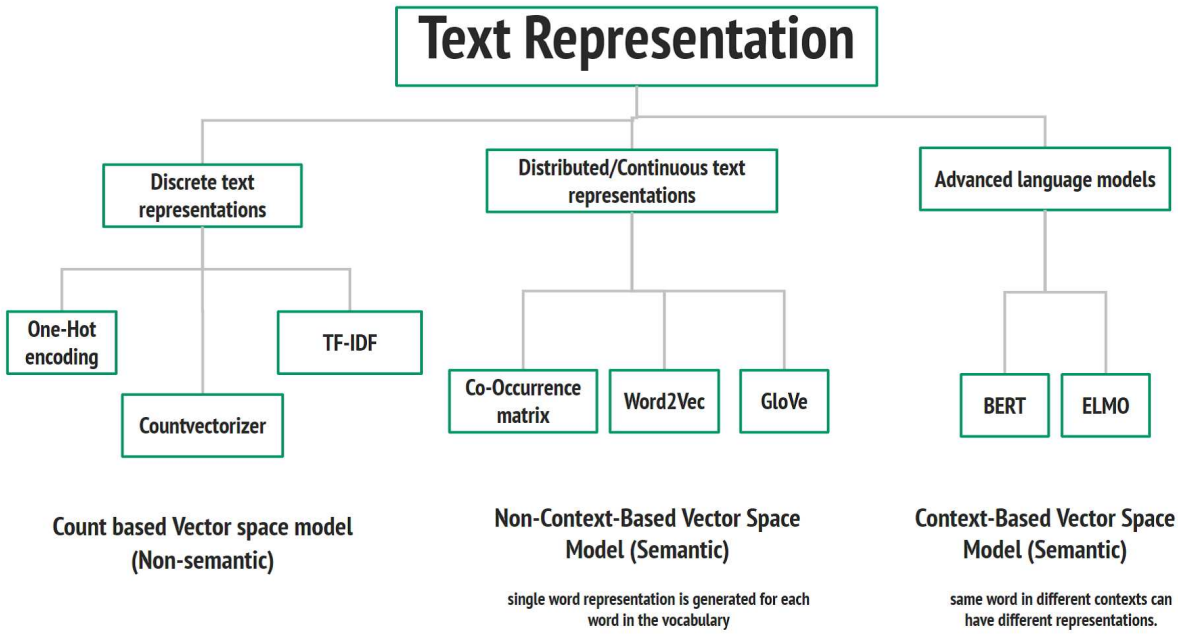


**Figure 7.** Text Representation Techniques.

### 2.5.2. Research Methodology

Figure 8 depicts the research methodology conducted in this paper. In the first part of the methodology, the collected texts have been revised and filtered by first removing retweets to avoid redundancy. Text cleaning is essential in preparing text data for use in NLP and machine learning models. It involves preprocessing the text to remove noise, fix structural issues, and standardize

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 7 September 2023 doi:10.20944/preprints202309.0497.v1



the format of the text; this can help improve the classification model's performance and make the text easier to work with. Text data is often messy and unstructured and can contain a variety of issues that can affect the performance of a classification model. These issues may include typos, misspellings, punctuation errors, and other irregularities that can confuse the model and make it difficult to understand the content of the text. Then, the URL addresses, emojis and other unwanted symbols have been removed. We used a regular expression in python to complete this job. Finally, the texts have been tokenized to prepare data for text representation.

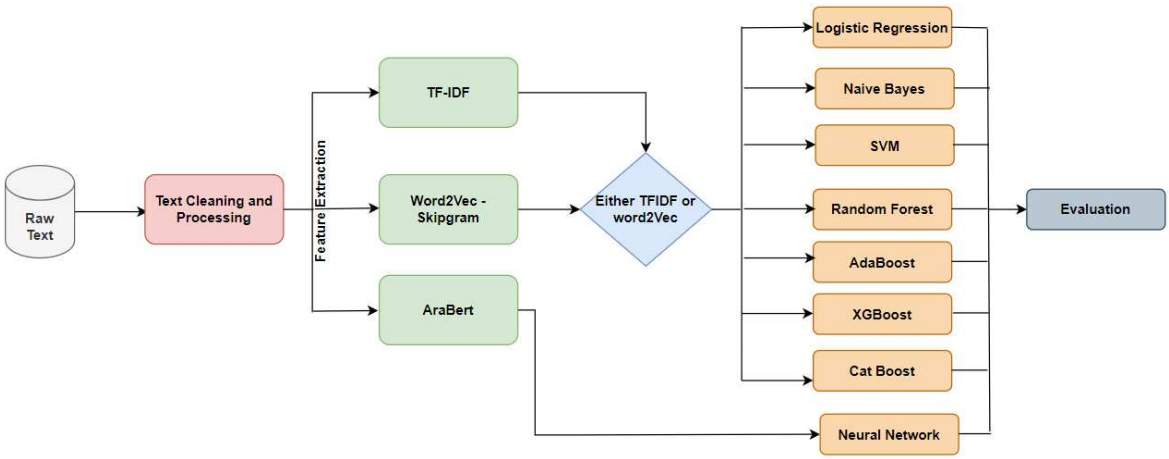


**Figure 8.** Research Methodology and the Experiment Framework.

In the second phase, the text of each message is then transformed into the numeric vector using the text representation models discussed in the previous section. We have applied TF-IDF and Word2Vec text representation techniques in addition we used AraBert transformer to produce text representation. But the latter will be only used with neural network model.

For AraBert model, we have used the pre-trained model as shown in Figure 9; we make fine tuning on our corpus as shown in Figure 10. During the pre-trained process, the transformer is trained over a large Arabic corpus. The output of this process is the pre-trained transformer mode which will be used later for fine-tuning based on our collected dataset.

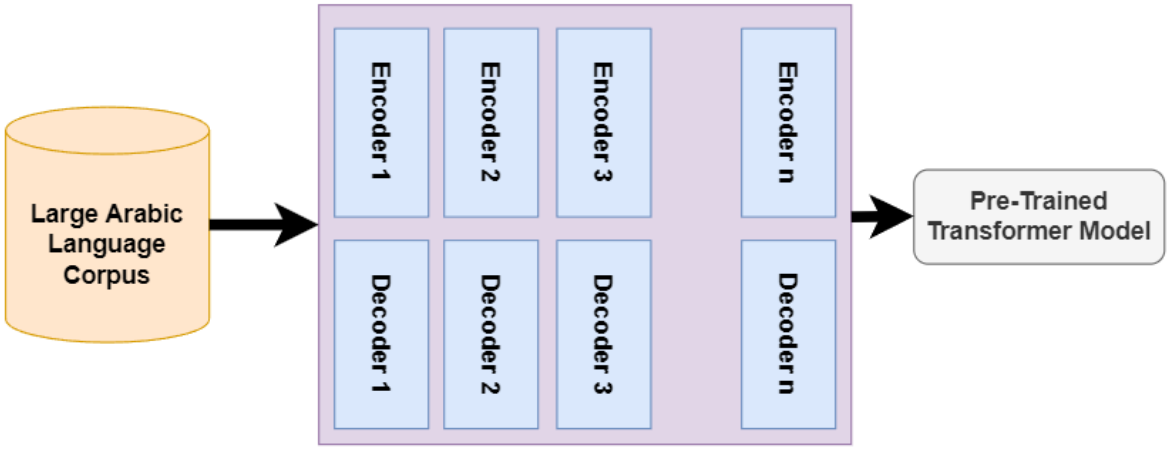


**Figure 9.** Pre-trained Process of Transformer.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 7 September 2023 doi:10.20944/preprints202309.0497.v1



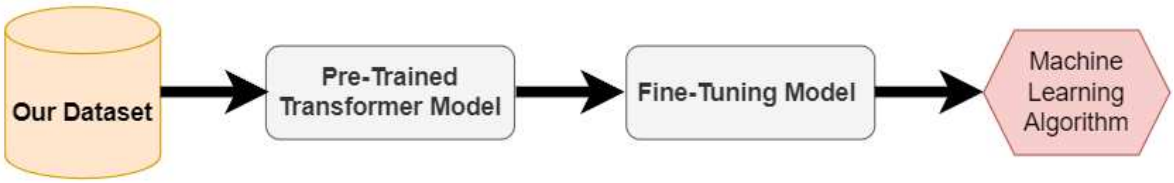


**Figure 10.** Fine Tuning Process.

During the fine-tuning process (see Figure 10), numerous settings may be changed, including the optimizer, learning rate, number of epochs, and dropout value. As part of the fine-tuning procedure, we tested various optimizers, including SGD optimizer, ADAM, and AdamW. We experimented with several learning rate values, including 1e-3, 1e-4, and 1e-5. We also experimented with them with several different epochs, ranging from 1 to 5. To prevent wasting time and storage, the terminating conditions were carefully chosen. We have tried Dropout values of 10e-2, 25e-2, and 50e-2, with each number yielding a somewhat different outcome.

Finally, the text representations of text in line with extracted features are entered into the NN model that was placed on the top of the pre-trained AraBert model. Two dense layers have been added to the NN model with ReLU activation functions. Also, a dropout layer was added to avoid overfitting during the training process, and the linear layer to find a correlation between input vectors and output labels. The ReLU layer will reduce the computation time required for model training. Finally, we divided the entire dataset to 70

### 2.5.3. Evaluation Measures

Choosing proper evaluation metrics for classification problems is tricky as every metric explain a specific part of the model performance. Wrong choices are likely to produce a poor explanation with deceived performance. Therefore, five evaluation metrics that capture different aspects of classification model predictions have been used. These metrics ensure a trade-off between the overall performance of the classification models. Since we have four class labels, we used weighted average to aggregate all evaluation results. The most popular evaluation metrics are Recall and Precision. Recall metrics, as shown in equation 1, can capture the proportion of hate speech that is correctly classified within hat speech. Precision, as shown in equation 2, is defined as the proportion of the hate speech tested as hate (see equation 2). F1 metric, as shown in equation 3, is used to combine the precision and recall metrics into a single metric that can work best with imbalanced data distribution. Finally, the accuracy metric shown in equation 4 reflects the proportion of all correctly classified examples. In addition to the above metrics, we used Area Under Curve (AUC), which estimates the area under the ROC curve formed by a set of Precision and Recall values and represented as a single value in the range [0, 1]. ROC curve presents the trade-off between recall and precision. The better model with high AUC is regarded as the superior model.

$$\text{Recall} = \frac{tp}{tp + fn} \tag{1}$$

$$\text{Precision} = \frac{tp}{tp + fp} \tag{2}$$

$$\text{F1} = 2 \times \left( \frac{Recall \times Precision}{Recall + Precision} \right) \tag{3}$$

$$\text{Accuracy} = \frac{tp + tn}{tp + tn + fp + fn} \tag{4}$$

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 7 September 2023 doi:10.20944/preprints202309.0497.v1



where tp (true positive) is the number of hate speech that is predicted as such. tn (true negative) is the number no hate speech that is predicted as such. fp (false positive) is the number of not hate speech predicted as hate speech. fn (false negative) is the number of hate speech predicted as not hate speech.

## 3. Experiments and Results

This section shows the empirical results of building hate speech detection. Three text representations models have been used to generate word vectors, namely: AraVec, TF-IDF and AraBert model. AraVec is a Word2Vec model trained over large Arabic corpus. AraBert is Bert alike transformer which has been trained over large Arabic corpus. These models can give us the text representation as vectors to fed into classification models. We have used seven machine learning classifiers: Support Vector Machine (SVM), Logistic regression (LR), Naive Bays (NB), Random Forest (RF), AdaBoost (Ada), XGBoost (XGB), and CatBoost (CatB)

### *3.1. Experimental setup and Hyperparameter tuning*

Table 6 shows the searching parameters and best parameters for each classifier. We have identified a list of values for each configuration parameters, then we used Grid search algorithm with five folds cross validation to selected best configurations for each classifier.

**Table 6.** Results of Negative Label.

| | Recall | | Precision | | F1 | |
|---|---|---|---|---|---|---|
| | W2V | TF-IDF | W2V | TF-IDF | W2V | TF-IDF |
| LR | 0.60 | 0.64 | 0.55 | 0.45 | 0.57 | 0.53 |
| NB | 0.39 | 0.30 | 0.58 | 0.53 | 0.47 | 0.38 |
| RF | 0.59 | 0.53 | 0.56 | 0.50 | 0.57 | 0.51 |
| SVM | 0.61 | 0.65 | 0.54 | 0.45 | 0.57 | 0.53 |
| AdaBoost | 0.56 | 0.59 | 0.54 | 0.46 | 0.55 | 0.52 |
| XGBoost | 0.58 | 0.54 | 0.57 | 0.50 | 0.58 | 0.52 |
| CatBoost | 0.59 | 0.55 | 0.57 | 0.51 | 0.58 | 0.53 |

### *3.2. Results*

To investigate the quality of the collected data in addition to the quality of annotation process, we conducted a comprehensive experimentation on building hate detection system using multiple machine learning algorithms and two main text representation techniques: Word2Vec (W2V) and TF-IDF in addition we used BERT based Arabic language called AraBert. As explained in the research methodology, we split the dataset into 70% training and 30% testing, then we used seven machine learning algorithms and training dataset to build different classifiers on features extracted from W2V and TF-IDF techniques. Finally, all constructed models have been evaluated on testing using multiple evaluation metrics.

To facilitate presenting the results, we organized all results into different tables based on the class labels in the dataset. Since we have four classes, we showed the evaluation results for each class label. Each table show the performance of each machine learning with each text representation technique. Then we added a table to summarize the overall results using weight average aggregation method. Table 6 shows the evaluation results for Negative class label. We omitted accuracy and AUC metrics because they are aggregated metrics and are not calculated individually for each class label. The bold text represents the best results between TF-IDF and W2V for each evaluation metric. The bold and red text represent the best machine learning model under each evaluation metric. Form the table, we can generally observe that W2V is more suitable for our text than TF-IDF. It is widely acknowledged that W2V can produce good text representation when the corpus contains over 25,000 vocabularies as in our case, therefore the machine learning algorithms that use W2V produce better results than those of TF-IDF. On the other hand, if we look at the machine learning algorithm, we can notice there is instability in terms of performance, such that we cannot identify one best model. However, for

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 7 September 2023 doi:10.20944/preprints202309.0497.v1



Recall metric we can see that SVM+TF-IDF is the best one, whereas for precision we can see that NB+W2V is the best one. This contradiction forces us to choose multiple options as good candidates. Finally, the best recall accuracy which is for (SVM+TF-IDF) suggests that the model can predict 65% of negative text as negative text, whereas the best precision score shows that 58% of the predicted texts are negative. The F1 metric can show compromises between Recall and Precision, which suggests that XGBoost and CatBoost with W2V are the best models for Negative class label.

Table 7 presents results for Neutral class label. Generally, the results are poor because the best recall or precision score is relatively low in. Interestingly, we can observe a stable result here more than the Negative class label. Also, we found that W2V is always produces good text representation for all machine learning models. If we look at the evaluation results between W2V and TF-IDF we can see there is big difference which suggest that TF-IDF is not appropriate for such kind of hate speech corpus. With respect to machine learning model, we cannot identify one best mode, but multiple ones according to the evaluation metrics. For example, NB+W2V can work well under Recall metric, whereas CatBoost+W2V can work well under Precision metric. If we take F1 metric as compromised solution, we can see that both NB and CatBoost with W2V are the best models.

**Table 7.** Results of Neutral Label.

| | Recall | | Precision | | F1 | |
|---|---|---|---|---|---|---|
| | W2V | TF-IDF | W2V | TF-IDF | W2V | TF-IDF |
| LR | 0.35 | 0.21 | 0.42 | 0.37 | 0.38 | 0.27 |
| NB | 0.47 | 0.27 | 0.37 | 0.36 | 0.41 | 0.30 |
| RF | 0.39 | 0.36 | 0.42 | 0.37 | 0.40 | 0.36 |
| SVM | 0.34 | 0.21 | 0.43 | 0.37 | 0.38 | 0.26 |
| AdaBoost | 0.31 | 0.23 | 0.40 | 0.35 | 0.35 | 0.28 |
| XGBoost | 0.38 | 0.34 | 0.43 | 0.37 | 0.40 | 0.35 |
| CatBoost | 0.38 | 0.34 | 0.44 | 0.38 | 0.41 | 0.36 |

Table 8 presents results for Positive class label. We can see the same trend as for the Neutral class label, but with different best machine learning models. First, we can confirm that the W2V is good text representation among all models, and CatBoost is the most accurate and stable model under three evaluation metrics. The overall results of Positive Label are good in comparison to the Neutral label and show good performance.

Finally, the evaluation results for 'Very Positive' class label are very poor as shown in Table 9. One reason for that is the nature of the dataset which is relatively imbalanced, which means that there is a big difference among the number of samples in each class label. Figure 1 shows the class the distribution of our dataset. We can notice there is imbalanced distribution between class labels. The 'very Positive' class label is the minor one. Therefore, the performance of machine learning over this label was very poor as shown in Table 9. Also, there is no stable results across all evaluation metrics, therefore it is difficult to judge which machine learning model is the most superior one.

**Table 8.** Results of Positive Label.

| | Recall | | Precision | | F1 | |
|---|---|---|---|---|---|---|
| | W2V | TF-IDF | W2V | TF-IDF | W2V | TF-IDF |
| LR | 0.53 | 0.37 | 0.47 | 0.39 | 0.50 | 0.38 |
| NB | 0.47 | 0.27 | 0.42 | 0.36 | 0.45 | 0.30 |
| RF | 0.52 | 0.36 | 0.48 | 0.37 | 0.50 | 0.36 |
| SVM | 0.52 | 0.36 | 0.48 | 0.38 | 0.50 | 0.37 |
| AdaBoost | 0.52 | 0.41 | 0.44 | 0.38 | 0.48 | 0.40 |
| XGBoost | 0.56 | 0.43 | 0.49 | 0.41 | 0.52 | 0.42 |
| CatBoost | 0.57 | 0.44 | 0.49 | 0.42 | 0.53 | 0.43 |

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 7 September 2023 doi:10.20944/preprints202309.0497.v1



**Table 9.** Results of Very Positive Label.

| | Recall | | Precision | | F1 | |
|---|---|---|---|---|---|---|
| | W2V | TF-IDF | W2V | TF-IDF | W2V | TF-IDF |
| LR | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| NB | 0.08 | 0.01 | 0.07 | 0.02 | 0.07 | 0.01 |
| RF | 0.02 | 0.04 | 0.37 | 0.34 | 0.04 | 0.07 |
| SVM | 0.00 | 0.00 | 0.26 | 0.00 | 0.01 | 0.00 |
| AdaBoost | 0.00 | 0.00 | 0.06 | 0.00 | 0.00 | 0.00 |
| XGBoost | 0.03 | 0.02 | 0.44 | 0.52 | 0.06 | 0.04 |
| CatBoost | 0.03 | 0.01 | 0.44 | 0.65 | 0.06 | 0.01 |

To get insights from the above results we aggregate all evaluation results using weighted average that consider the class distribution with final calculation as shown in Table 10. We can see that W2V is generally best text representation for our corpus. All machine learning models behave relatively accurate with good performance. Amongst them, CatBoost is the most stable and accurate model.

**Table 10.** Aggregated Results for all labels using weighted average.

| | Recall | | Precision | | F1 | | Accuracy | | AUC | |
|---|---|---|---|---|---|---|---|---|---|---|
| | W2V | TF-IDF | W2V | TF-IDF | W2V | TF-IDF | W2V | TF-IDF | W2V | TF-IDF |
| LR | 0.49 | 0.42 | 0.48 | 0.40 | 0.48 | 0.39 | 0.49 | 0.42 | 0.48 | 0.50 |
| NB | 0.43 | 0.38 | 0.46 | 0.41 | 0.44 | 0.37 | 0.43 | 0.38 | 0.48 | 0.47 |
| RF | 0.50 | 0.43 | 0.49 | 0.43 | 0.49 | 0.43 | 0.50 | 0.43 | 0.49 | 0.49 |
| SVM | 0.49 | 0.41 | 0.48 | 0.39 | 0.48 | 0.41 | 0.49 | 0.41 | 0.49 | 0.50 |
| AdaBoost | 0.47 | 0.40 | 0.46 | 0.41 | 0.46 | 0.40 | 0.47 | 0.41 | 0.41 | 0.46 |
| XGBoost | 0.50 | 0.43 | 0.50 | 0.44 | 0.50 | 0.43 | 0.50 | 0.44 | 0.48 | 0.50 |
| CatBoost | 0.51 | 0.44 | 0.51 | 0.44 | 0.50 | 0.44 | 0.51 | 0.44 | 0.47 | 0.50 |

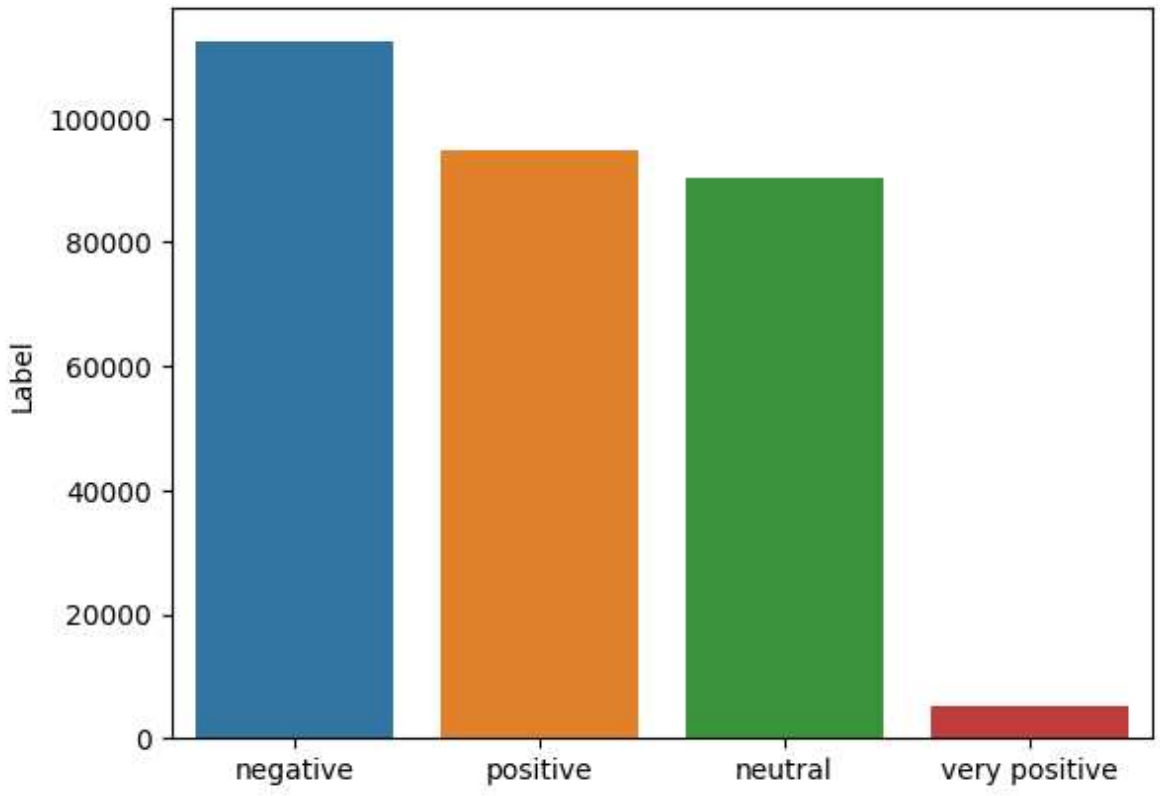


**Figure 11.** Class Distribution of class Label.

 doi:10.20944/preprints202309.0497.v1



Concerning the AraBert Model, we finetuned this transformer on our Arabic hate speech corpus, then we build a neural network model based on the CLS embedding. It is important to note the transformer usually used their own tokenizer and they produce their text representation as output through CLS embedding. Then this embedding vector is connected to fully connect Neural network. The model has been evaluated over testing corpus using the same evaluation measures as shown in Table 11. We can see the AraBert has the capability to learn text representation better than W2V and TF-IDF techniques and also produces good results in comparison to the weighted average results of the machine learning models.

**Table 11.** Results of AraBert Finetuned Model.

| metric | Recall | Precision | F1 | Accuracy | AUC |
|---|---|---|---|---|---|
| value | 0.61 | 0.68 | 0.63 | 0.62 | 0.68 |

To conclude, we can confirm that the collected data and annotation process were very appropriate and the obtained evaluation results show good performance for this complex and unstructured domain. We also, should not overlook to the complexity of processing Arabic text especially in Processing the natural Arabic language. For example, the word spelling can differ from one sentence to another, which changes the meaning and there are many different Arabic dialects, even in the same country, which makes it harder to understand the meaning of the sentence, and final the word diacrasy which can also change the meaning.

## 4. Conclusion and Future Work

In this study, we address the intricate challenge of Hate Speech Detection in Arabic, a language with a wide variety and nuanced cultural characteristics. This study intends to aid in the fight against hate speech in Arabic that is spread online. A notable resource in this field is the creation of a fresh multi-class Arabic dataset with over 400,000 annotated tweets that have been sentimentally classified. Additionally, using text representation techniques, including WordVec, TF-IDF, and AraBert, and seven machine learning classifiers, we evaluated the effectiveness of several machine learning models in detecting hate speech in tweets written in the Arabic Jordanian dialect. Our empirical findings indicated our dataset's usefulness and precisely how hate speech could be identified in this difficult, unstructured environment. Although this work makes significant advancements in the Arabic Hate Speech Detection field, several areas still might be used for more investigation. In the future, we want to increase the size and diversity of our dataset, examine multilingual methods, improve contextual analysis, create real-time detection systems, look into user-specific detection, and address bias and fairness concerns. By promoting a safer online environment, these initiatives will help develop more effective and culturally relevant solutions for addressing hate speech in Arabic.

**Funding:** This research was funded by the Ministry of Higher Education and Scientific Research/Jordan, grant number ICT-Ict/1/2021.

**Data Availability Statement:** The dataset (Corpus) associated with this research is publicly available. at https://data.mendeley.com/datasets/mcnzzpgrdj/1.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Kapoor, K.K.; Tamilmani, K.; Rana, N.P.; Patil, P.; Dwivedi, Y.K.; Nerur, S. Advances in social media research: Past, present and future. *Information Systems Frontiers* **2018**, *20*, 531–558.
2. Ngai, E.W.; Tao, S.S.; Moon, K.K. Social media research: Theories, constructs, and conceptual frameworks. *International Journal of Information Management* **2015**, *35*, 33–44. https://doi.org/https://doi.org/10.1016/j.ijinfomgt.2014.09.004.

3. Yalçınkaya, O.D. Instances of Hate Discourse in Turkish and English. *Turkish Studies-Language & Literature* **2022**, *17*.
4. Community Standards Enforcement Report. https://transparency.fb.com/reports/community-standards-enforcement/hate-speech/facebook/. Accessed: 2023-09-05.
5. Omar, A.; Mahmoud, T.M.; Abd-El-Hafeez, T. Comparative performance of machine learning and deep learning algorithms for Arabic hate speech detection in osns. In Proceedings of the Proceedings of the International Conference on Artificial Intelligence and Computer Vision (AICV2020). Springer, 2020, pp. 247–257.
6. Fortuna, P.; Soler, J.; Wanner, L. Toxic, hateful, offensive or abusive? what are we really classifying? an empirical analysis of hate speech datasets. In Proceedings of the Proceedings of the 12th language resources and evaluation conference, 2020, pp. 6786–6794.
7. Gilani, S.R.S.; Cavico, F.J.; Mujtaba, B.G. Harassment at the workplace: A practical review of the laws in the United Kingdom and the United States of America. *Public Organization Review* **2014**, *14*, 1–18.
8. Coban, O.; Ozel, S.A.; Inan, A. Detection and cross-domain evaluation of cyberbullying in Facebook activity contents for Turkish. *ACM Transactions on Asian and Low-Resource Language Information Processing* **2023**, *22*, 1–32.
9. Husain, F. Arabic offensive language detection using machine learning and ensemble machine learning approaches. *arXiv preprint arXiv:2005.08946* **2020**.
10. Nguyen, T. Merging public health and automated approaches to address online hate speech. *AI and Ethics* **2023**, pp. 1–10.
11. Chakraborty, T.; Masud, S. Nipping in the bud: detection, diffusion and mitigation of hate speech on social media. *ACM SIGWEB Newsletter* **2022**, *2022*, 1–9.
12. Zsila, Á.; Reyes, M.E.S. Pros & cons: impacts of social media on mental health. *BMC psychology* **2023**, *11*, 201.
13. Siddiqui, S.; Singh, T.; et al. Social media its impact with positive and negative aspects. *International journal of computer applications technology and research* **2016**, *5*, 71–75.
14. Akram, W.; Kumar, R. A study on positive and negative effects of social media on society. *International journal of computer sciences and engineering* **2017**, *5*, 351–354.
15. Sobaih, A.E.E.; Moustafa, M.A.; Ghandforoush, P.; Khan, M. To use or not to use? Social media in higher education in developing countries. *Computers in Human Behavior* **2016**, *58*, 296–305.
16. Ansari, J.A.N.; Khan, N.A. Exploring the role of social media in collaborative learning the new domain of learning. *Smart Learning Environments* **2020**, *7*, 1–16.
17. Alghizzawi, M.; Habes, M.; Salloum, S.A.; Ghani, M.; Mhamdi, C.; Shaalan, K. The effect of social media usage on students'e-learning acceptance in higher education: A case study from the United Arab Emirates. *Int. J. Inf. Technol. Lang. Stud* **2019**, *3*, 13–26.
18. Jahan, M.S.; Oussalah, M. A systematic review of Hate Speech automatic detection using Natural Language Processing. *Neurocomputing* **2023**, p. 126232.
19. Husain, F.; Uzuner, O. A survey of offensive language detection for the Arabic language. *ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP)* **2021**, *20*, 1–44.
20. Schmidt, A.; Wiegand, M. A survey on hate speech detection using natural language processing. In Proceedings of the Proceedings of the fifth international workshop on natural language processing for social media, 2017, pp. 1–10.
21. Al-Hassan, A.; Al-Dossari, H. Detection of hate speech in social networks: a survey on multilingual corpus. *Computer Science & Information Technology (CS & IT)* **2019**, *9*, 83.
22. Albadi, N.; Kurdi, M.; Mishra, S. Are they Our Brothers? Analysis and Detection of Religious Hate Speech in the Arabic Twittersphere. In Proceedings of the 2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM). IEEE, 2018, pp. 69–76.
23. Yi, P.; Zubiaga, A. Session-based cyberbullying detection in social media: A survey. *Online Social Networks and Media* **2023**, *36*, 100250.
24. Aldjanabi, W.; Dahou, A.; Al-qaness, M.A.; Elaziz, M.A.; Helmi, A.M.; Damaševičius, R. Arabic offensive and hate speech detection using a cross-corpora multi-task learning model. In Proceedings of the Informatics. MDPI, 2021, Vol. 8, p. 69.
25. Mozafari, M.; Farahbakhsh, R.; Crespi, N. A BERT-based transfer learning approach for hate speech detection in online social media. In Proceedings of the Complex Networks and Their Applications VIII:

Volume 1 Proceedings of the Eighth International Conference on Complex Networks and Their Applications COMPLEX NETWORKS 2019 8. Springer, 2020, pp. 928–940.

26. Awal, M.R.; Cao, R.; Lee, R.K.W.; Mitrović, S. Angrybert: Joint learning target and emotion for hate speech detection. In Proceedings of the Pacific-Asia conference on knowledge discovery and data mining. Springer, 2021, pp. 701–713.
27. Haddad, B.; Orabe, Z.; Al-Abood, A.; Ghneim, N. Arabic offensive language detection with attention-based deep neural networks. In Proceedings of the Proceedings of the 4th workshop on open-source Arabic corpora and processing tools, with a shared task on offensive language detection, 2020, pp. 76–81.
28. Abuzayed, A.; Elsayed, T. Quick and simple approach for detecting hate speech in Arabic tweets. In Proceedings of the Proceedings of the 4th workshop on open-source Arabic Corpora and processing tools, with a shared task on offensive language detection, 2020, pp. 109–114.
29. Hassan, S.; Mubarak, H.; Abdelali, A.; Darwish, K. Asad: Arabic social media analytics and understanding. In Proceedings of the Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations, 2021, pp. 113–118.
30. Alsafari, S.; Sadaoui, S.; Mouhoub, M. Hate and offensive speech detection on Arabic social media. *Online Social Networks and Media* **2020**, *19*, 100096.
31. Alsafari, S.; Sadaoui, S.; Mouhoub, M. Deep learning ensembles for hate speech detection. In Proceedings of the 2020 IEEE 32nd International Conference on Tools with Artificial Intelligence (ICTAI). IEEE, 2020, pp. 526–531.
32. Aref, A.; Al Mahmoud, R.H.; Taha, K.; Al-Sharif, M.; et al. Hate speech detection of Arabic shorttext. In Proceedings of the CS IT Conf. Proc, 2020, Vol. 10, pp. 81–94.
33. Romim, N.; Ahmed, M.; Talukder, H.; Saiful Islam, M. Hate speech detection in the bengali language: A dataset and its baseline evaluation. In Proceedings of the Proceedings of International Joint Conference on Advances in Computational Intelligence: IJCACI 2020. Springer, 2021, pp. 457–468.
34. Faris, H.; Aljarah, I.; Habib, M.; Castillo, P.A. Hate Speech Detection using Word Embedding and Deep Learning in the Arabic Language Context. In Proceedings of the ICPRAM, 2020, pp. 453–460.
35. Khezzar, R.; Moursi, A.; Al Aghbari, Z. arHateDetector: detection of hate speech from standard and dialectal Arabic Tweets. *Discover Internet of Things* **2023**, *3*, 1.
36. Alshaalan, R.; Al-Khalifa, H. Hate speech detection in saudi twittersphere: A deep learning approach. In Proceedings of the Proceedings of the fifth Arabic natural language processing workshop, 2020, pp. 12–23.
37. Saeed, A.M.; Ismael, A.N.; Rasul, D.L.; Majeed, R.S.; Rashid, T.A. Hate speech detection in social media for the Kurdish language. In Proceedings of the The International Conference on Innovations in Computing Research. Springer, 2022, pp. 253–260.
38. Alsafari, S.; Sadaoui, S. Semi-supervised self-learning for arabic hate speech detection. In Proceedings of the 2021 IEEE International Conference on Systems, Man, and Cybernetics (SMC). IEEE, 2021, pp. 863–868.
39. Salomon, P.O.; Kechaou, Z.; Wali, A. Arabic hate speech detection system based on AraBERT. In Proceedings of the 2022 IEEE 21st International Conference on Cognitive Informatics & Cognitive Computing (ICCI* CC). IEEE, 2022, pp. 208–213.
40. Mursi, K.T.; Alahmadi, M.D.; Alsubaei, F.S.; Alghamdi, A.S. Detecting islamic radicalism arabic tweets using natural language processing. *IEEE Access* **2022**, *10*, 72526–72534.
41. Ameur, M.S.H.; Aliane, H. AraCOVID19-MFH: Arabic COVID-19 multi-label fake news & hate speech detection dataset. *Procedia Computer Science* **2021**, *189*, 232–241.
42. Alshalan, R.; Al-Khalifa, H. A deep learning approach for automatic hate speech detection in the saudi twittersphere. *Applied Sciences* **2020**, *10*, 8614.
43. Duwairi, R.; Hayajneh, A.; Quwaider, M. A deep learning framework for automatic detection of hate speech embedded in Arabic tweets. *Arabian Journal for Science and Engineering* **2021**, *46*, 4001–4014.
44. Anezi, F.Y.A. Arabic hate speech detection using deep recurrent neural networks. *Applied Sciences* **2022**, *12*, 6010.
45. Ahmed, I.; Abbas, M.; Hatem, R.; Ihab, A.; Fahkr, M.W. Fine-tuning Arabic Pre-Trained Transformer Models for Egyptian-Arabic Dialect Offensive Language and Hate Speech Detection and Classification. In Proceedings of the 2022 20th International Conference on Language Engineering (ESOLEC). IEEE, 2022, Vol. 20, pp. 170–174.

46. Beyhan, F.; Çarık, B.; Arın, İ.; Terzioğlu, A.; Yanikoglu, B.; Yeniterzi, R. A Turkish hate speech dataset and detection system. In Proceedings of the Proceedings of the Thirteenth Language Resources and Evaluation Conference, 2022, pp. 4177–4185.
47. Mollas, I.; Chrysopoulou, Z.; Karlos, S.; Tsoumakas, G. ETHOS: a multi-label hate speech detection dataset. *Complex & Intelligent Systems* **2022**, *8*, 4663–4678.
48. Althobaiti, M.J. Bert-based approach to arabic hate speech and offensive language detection in twitter: Exploiting emojis and sentiment analysis. *International Journal of Advanced Computer Science and Applications* **2022**, *13*.
49. Alkomah, F.; Ma, X. A literature review of textual hate speech detection methods and datasets. *Information* **2022**, *13*, 273.
50. Barbosa, L.; Feng, J. Robust sentiment detection on twitter from biased and noisy data. In Proceedings of the Coling 2010: Posters, 2010, pp. 36–44.
51. Alayba, A.M.; Palade, V.; England, M.; Iqbal, R. Arabic language sentiment analysis on health services. In Proceedings of the 2017 1st international workshop on arabic script analysis and recognition (asar). IEEE, 2017, pp. 114–118.
52. Refaee, E.; Rieser, V. An arabic twitter corpus for subjectivity and sentiment analysis. In Proceedings of the LREC, 2014, pp. 2268–2273.
53. Al-Twairesh, N. Sentiment analysis of Twitter: a study on the Saudi community. PhD thesis, King Saud University Riyadh, Saudi Arabia, 2016.
54. Mubarak, H.; Darwish, K.; Magdy, W. Abusive language detection on Arabic social media. In Proceedings of the Proceedings of the first workshop on abusive language online, 2017, pp. 52–56.
55. Anootation Exam: https://forms.gle/9e56L2j8vH9mNSiV9.
56. Landis, J.R.; Koch, G.G. The measurement of observer agreement for categorical data. *biometrics* **1977**, pp. 159–174.