**Ensemble Method for Levantine Hate Speech Detection in Twitter**

**Abstract**

Nowadays, people use Online Social Networks (OSNs) to express feelings and ideas and to communicate and share information. With the freedom space provided by such networks, some people tend to propagate hate speech and insults. An early Detection of such content is crucial for predicting conflicts and could prevent the emotions to become actions or to spread widely.

Hate speech detection work on the Arabic text is sparse and scant compared to other languages like English. Moreover, Arabic corpora of short text in Levantine dialect for hate speech is also scant.

In this paper, we constructed ensemble method for hate speech detection in short text written in Levantine dialect. We used multiple classification algorithms, including Naïve Bayes, Random Forest, and Support Vector Machines, and two ensemble methods, hard voting and soft voting, on our own dataset collected from Twitter. The experiment results showed that using ensemble method can improve the classification performance. The best result is achieved when using soft voting with F1 measure 00.00% on our dataset.

Keywords: Arabic; hate-speech; offensive language; social networks; natural language processing; text classification; supervised machine learning;

# Introduction

With the widespread and increasing use of Online Social Networks (OSNs), people use OSNs to express feelings and ideas and to communicate and share information. With the freedom space provided by such networks, some people tend to propagate hate speech and insults. The shared content may be

Hate speech, offensive language, harassing, racism, insults, and other types of online abusive behavior. Unfortunately, OSNs are a perfect platform for publishing hurtful content such as cyberbullying and hate speech [9].

With the huge amount of content published every minute (e.g. 456,000 tweets are sent on Twitter every minute [14]).

The Arabic language is the fourth used language on the Internet and ranked as the sixth used language on Twitter, and is ranked in the fifth position of spoken language in the world. More than 6% of the world’s population speak Arabic, and there is a remarkable growth in using OSNs in the Arab region.

Arabic text classification and Arabic NLP techniques are relatively hard to implement due to different factors [2] [3]:

* Arabic is rich and complex morphological language.
* Arabic language has different forms such as the dialectical Arabic and each Arabic country has different dialect.
* The colloquial Arabic has many misspellings that differs morphologically and phonologically.
* Arabic has complex orthography and morphosyntactic rules.

In this paper, a dataset is created that is targeting the problem of hate speech on Twitter for Levantine countries. The dataset was collected using several keywords such as racism, insults, and Islam. The dataset was labeled manually with two classes; (Hate and Normal). To capture the hidden relations of words of the dataset, we used word embedding techniques (Word2Vec and the AraVec) for extracting a set of words features. We used multiple classification algorithms, including Naïve Bayes, Random Forest, and Support Vector Machines, and two ensemble methods, hard voting and soft voting, on our dataset. The experiment results showed that using ensemble method can improve the classification performance. The best result is achieved when using soft voting with F1 measure 79.8% on our dataset.

The structure of the paper is as follows. In section 2 we define the hate speech detection problem, while in section 3 we discuss related work. We describe our dataset collected from twitter in section 4. We overview our study approach and elaborate on the proposed method in section 5. We then experimentally evaluate the performance of different approaches in Section 6, concluding the paper in Section 7, by summarizing the findings and proposing future work.

# Problem Definition

It is difficult for machine learning to detect written hate speech because of the short text samples, the lack of continuity, confusion with regard to the identity of the authors and the recipients, and the intention or unclear emotions, as well as the use of slang, mixed language, spelling errors, homogeneous retention, and distortion. Especially when depending on the context, the meanings of words can vary greatly using humor, sarcasm, hints and metaphor. Therefore, it is important to provide a clear and concise definition for hate speech. Following the definition provided by [1], Hate speech is any communicative acts that used to express hatred towards a person or a group on the basis of some characteristic such as race, ethnicity, gender, sexual orientation, nationality, religion, or other characteristic. This is very important when we deal with unlabeled dataset where human annotators are involved.

Hate speech detection task is formulated as a binary classification problem, given an input text T, to output True, if T contains Hate Speech and False otherwise. The model is built by learning from a training set and evaluated later on unseen testing data.

Specifically, the input text is represented by a machine-readable format with retention of informative characteristics. a machine learning algorithm takes this representation as input and assigns it to one of two classes with a certain confidence. During the training phase, the classifier is built with this discrimination information, and then applied on unseen data, in order to measure its generalization ability.

In this study, we focus on manually annotated dataset from Twitter posts. We evaluate the performance of several established text representations (e.g., Bag of words, word embeddings) and several classification algorithms (e.g., Naïve Bayes, Random Forest, and Support Vector Machines). Moreover, we examine whether the contribution of ensemble methods (hard voting and soft voting) to the Hate Speech classification process can provide top performance in the Hate Speech detection task.

# Related Work

Hate Speech detection has been a growing attention research topic over the past few years. Some studies have tried to address the problem on the OSNs, there is a very little focus for the Arabic language [5].

In this section, we provide a short review of the related work for Arabic Hate Speech detection. Examples of such tasks can be found in [6] where the authors explore various approaches to detect hate speech and offensive language which include deep learning, transfer learning and multitask learning, while [5] aim to validate the effectiveness of twelve machine learning algorithms and two deep learning against a dataset collected from the social network platforms (Facebook, Twitter, Instagram, and YouTube). In another work [7], the authors used a specific word list as a seed to create a training set, to experiment with and create an offensive language detector. Authors in [8] proposed a method to detect abusive language, they used SVM with n-gram features for the classification where they achieved an F1-score of 0.82. They collected their own dataset from YouTube comments. [10] created a dataset of religious hate-speech discussions on Twitter, they used this data to train an RNN based classifier for automatic detection of hate-speech, they achieved 0.84 Area under the ROC curve. The authors also used their dataset to create multiple hate-speech lexicons. In [11], authors collected a dataset using several keywords such as racism and implement a deep learning model. [12] implement a classifier that combines both CNN and RNN in a joint architecture) achieved 0.73 macro-F1 score on the dev set. [13] constructed their method for data preprocessing and balancing and presented a Convolutional Neural Network (CNN) and bidirectional Gated Recurrent Unit (GRU) models used. [15] introduced a Levantine Hate Speech and Abusive (L-HSAB) Twitter dataset with objective to be a benchmark dataset for automatic detection of online Levantine toxic contents.

## Dataset and Preprocessing

The dataset acquisition step is the most time-consuming part of the text classification process [25].

The acquisitioned dataset may contain unnecessary data such as non-Arabic characters, repeated characters, emoji, or URLs. Therefore, the data should be cleaned and filtered before training for best results. This step is done using the following technics:

* Remove stop words.
* Remove non-Arabic words.
* Cleaning: removing unknown characters, diacritics, punctuation, URLs, etc.
* Arabic Dialects Normalization.
* Elongation removal: removing the repeated letters.
* Remove duplicated samples.

The researchers in Hate Speech detection Literature have their own dataset, collected from different sources. Some of these studies annotated their data as Hate or not, while the others annotated their data as hate, abusive or clean. Most of the studies collected (or “borrowed”) a dataset from twitter [6-10-11-12-13-15-23], YouTube [8-24] or from different OSNs [5-21]. Authors in [22] collected a dataset from a known channel news “AlJazeera.net”. The following table characterize the datasets proposed by different studies: data source, number of samples, percent of hate samples (we considered the hate and abusive as hate):

|  |  |  |  |
| --- | --- | --- | --- |
| **Ref.** | **Size** | **Hate %** | **Description** |
| [5] | 20,000 | Balanced | Most popular accounts and pages |
| [6-12] | 10,000 | 4.5 | Arabic tweets containing the vocative particle “يا” |
| [8-24] | 15,050 | 38.65 | YouTube channels uploading videos about Arab world celebrities |
| [11] | 1,634 | 52 | Tweets containing some specific words |
| [13] | 10,000 | 24 | Same dataset in [6-12] but balanced |
| [15] | 5,846 | 37.56 | Levantine, politicians, social and political activists and TV anchors |
| [21] | 6,039 | 36.51 | Tunis dialect |
| [22] | 32,000 | 81 | Modern Standard Arabic (MSA) |
| [23] | 3,950 | 42.66 | Tweets referring to different religious groups |

## Text representations for Hate Speech

Text representations is the mapping of written human language into a collection of useful features which are understandable by a hate speech detection model. Following, the different text representation used models:

* N-grams [16].
* TF-IDF Weighting [17].
* Word Embeddings [20].

Different researchers have employed different variety of feature representation techniques for Arabic hate speech, N-grams-based [7-8-10-15-21], TFIDF-based [12-15] and word embedding [6-11-12-13-23-24].

## Classification approaches

Several classification algorithms have been deployed, such as Support Vector Machine, Logistic Regression and others. SVM was used in [5-8-12-13-15-21], while Logistic Regression was used in [12-13]. Naïve Bayes was used in [15-21], while CNN and LSTM were used in [6-11-13-23-24]. Authors in [5] use twelve algorithms (MultinominalNB, ComplementNB, BernouliNB, Decision Tree and others).

It is noticeable that the studies in Arabic hate speech detection are still scant. The algorithms used are mostly limited to SVM or combined LSTMN and CNN.

# Dataset

We present in the following sections the training and test data used in our experiments.

## Data Collection

The dataset is collected from Twitter using Twitter streaming Application Programming Interface (API) during discrete periods from November 2017 to November 2020. The collect task is constrained to Arabic language and to geographical coordinates (Syria and around). The collector algorithm ignores tweets from famous or news channels. The total number of collected tweets is 8948.

## Data Annotation

The annotation task requires labeling the tweets of our dataset as Hate or Normal. Our annotation instructions defined two labels as: “Normal” tweets which have no offensive, aggressive, insulting and profanity content; and “Hate” tweets which contain an abusive language or dedicate the offensive, insulting or aggressive speech towards a specific person or a group of people based on their descriptive identity (race, gender, religion, skin color, belief). Examples ???. The annotation task was assigned to three annotators. Annotators represent varied cultures, ages, genders, and jobs and they are Levantine native speakers.

## Data Preprocessing

Several research studies have emphasized that using text preprocessing enhances classification results [27]. Hence, we applied different preprocessing techniques to filter noisy and non-informative features from the tweet. The first step is to eliminate duplicated tweets and remove short ones which are less than two words. The next step is to remove non-Arabic letters and special characters from the tweets; emoticons are also removed. then, removal of diacritics and URLs, white spaces, hashtags, punctuations and stop-words also was done. Moreover, it is common for users to repeat a letter, especially a vowel, one or more times to emphasis or to express enthusiasm. The next step is to replace elongated words by their original forms.

The next table shows the count of tweets after each filtering process.

|  |  |
| --- | --- |
| Action | Count |
| Collect Data | 8948 |
| Eliminate duplicated tweets | 6254 |
| Remove very short and non-Arabic tweets | 6071 |
| Remove HAHAHA tweets | 6038 |

## Data Statistics

A detailed review of the statistics of final version of dataset is provided in Table ?. for example, the count of tweets labeled as “Normal” which contain mentions is 4222 equivalents to 71.51% of normal tweets, the count of truncated tweets labeled as “Hate” is 47 equivalents to 35% of hate tweets. FRM denotes the count of tweets which are **F**avorited, **R**etweeted or has **M**entions. And FRMTH denotes plus **T**runcated tweets and that has **H**ashtags.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Feature | **Normal** | | **Hate** | | **Total** | |
| **count** | **%** | **Count** | **%** | **count** | **%** |
| Totals | 5774 | 97.78 | 155 | 2.56 | 5929 | 100.00 |
| **M**entions | 4110 | 71.51 | 127 | 81.94 | 4237 | 71.46 |
| **H**ashtags | 1030 | 17.28 | 37 | 23.87 | 1067 | 18.00 |
| **R**etweet count | 1843 | 30.95 | 72 | 46.45 | 1915 | 32.30 |
| **F**avorite count | 1058 | 18.68 | 35 | 22.58 | 1093 | 18.43 |
| **T**runcated | 821 | 13.91 | 56 | 36.13 | 877 | 14.79 |
| **I**n reply to user | 2438 | 43.26 | 69 | 44.52 | 2507 | 42.28 |
| **P**laces | 276 | 4.86 | 4 | 2.58 | 280 | 4.72 |
| **FRM** | 4597 | 80.25 | 146 | 94.19 | 4743 | 80.00 |
| **FRMH** | 4875 | 84.98 | 151 | 97.42 | 5026 | 84.77 |

The following chart depicts the statistics of previous table.

It is noticeable that rate of hate tweets whose features have values is greater than of normal tweets. For example, that rate of hate tweets which have mentions is greater than normal tweets. This applies on “hashtags”, “retweet count”, “favorite count” and “truncated” features, and in less for “in reply to user”. This rate increases when combining more features (e.g., 0.97 when combining “mentions”, “hashtags”, “retweet count” and “favorite count” features).

We found that when building a dataset for hate speech, it is preferable to constrain tweets to those whose text have mentions or hashtags, have favorited or have retweeted. We also assume that tweets with no mentions, no hashtags not favorited and not retweeted are valuable even they are labeled as hate.

## Data Augmentation

Researches shows that classifiers trained on imbalanced dataset may tend to have a high number of false negatives.

# Proposed Model

## Text representations

The input text is represented by a machine-readable format with retention of informative characteristics. We use Word embedding which is a representation of each word in a text as a vector of a few hundred dimensions. The Word2Vec tool offers two different architectures: Continuous Bag of Words (CBOW) and Skip-gram. CBOW tries to output the target word through the input context, on the contrary, Skip-gram tries to output an appropriate context for a target word.

AraVec [33] provides a few different models of pretrained Arabic word embeddings. These embeddings are trained on one of 3 datasets: tweets, Wikipedia articles or web pages. For each dataset two models are built (CBOW and Skip-gram). [34] suggests that the AraVec word embedding models could be useful for the task of detecting offensive language in Arabic.

Our dataset of Levantine dialect is collected from Twitter; therefore, we used AraVec trained on Twitter dataset. Skip-gram architecture usually produces better semantic accuracy than the CBOW architecture [20], so we used the Skip-gram architecture.

## Classification Methods

Generated features from text representation are fed to a classifier that decides the presence of Hate Speech content. We use a variety of classification models, as outlined below.

Naive Bayes (NB) [28] is a simple probabilistic classifier, based on Bayesian statistics. NB serves as baseline for various machine learning tasks [29] and reduces time complexity on large datasets.

Logistic Regression (LR) [30] is another statistical model common in binary classification tasks. It produces a prediction via a linear combination of the matrix of input with a vector of weights, passed as input to a logistic function which returns a value between 0 and 1.

The Random Forest (RF) [31] is an ensemble learning technique used for both classification and regression tasks. It combines multiple decision trees during training by bootstrap-aggregated ensemble learning, aiming to reduce noise and overfitting by merging multiple weak learners.

Stochastic Gradient Descent (SGD classifier) tries to find the minimum for a function (e.g., the loss/error function). The SGD optimizer works iteratively by moving in the direction of the gradient. The direction of the minimum is in the direction where the values are decreasing which computed using gradients.

Support Vector Classifier (SVC) is a supervised learning model used for classification, regression and outlier analysis. It constructs a hyperplane or a set of hyperplanes in a high- or infinite- dimensional space to separate data.

XGB classifier, XGB stands for eXtreme Gradient Boosting, which is a boosting algorithm based on the gradient boosted decision tree algorithm. It applies a better regularization technique to reduce overfitting, and it is one of the differences of the gradient boosting.

CatBoost classifier, means categorical boosting and it is a high-performance open source library for gradient boosting algorithm.

## Ensemble Learning

Ensemble Learning refers to a method where many base models are combined to carry out the same task. These base models are usually referred to as weak learners. It works on the principle that a weak learner predicts poorly when alone. But when combined with other weak learners, they create a stronger learner which performs much better than the lesser learners. We can create a better classifier by aggregating the predictions of each classifier and predict the class that gets the most votes. This approach is called as Voting Classification.

Hard Voting Classifier: Aggregate predictions of each classifier and predict the class that gets most votes. This is called as “**majority – voting**” or “**Hard – voting**” classifier.

Soft Voting Classifier: In an ensemble model, all classifiers are able to estimate class probabilities, then we can predict the class with the highest probability, averaged over all the individual classifiers. Need to ensure that all classifiers can estimate class probabilities.

# Experiments and Results

# Conclusion and Future Work

This paper presented a dataset for Levantine Arabic hate speech detection in OSNs. The dataset containing 0000 tweets collected from Twitter and manually labeled by three Arabic annotators into two classes: Hate, and not hate. Several machine learning algorithms (e.g., Naïve Bayes, SVM, Random Forest, etc.) were applied to evaluate the performance of the dataset. We also show the correlation between hate tweets and some features such as retweet counts, favorite counts and existence of mentions.

In future work, we will extend the dataset by adding more features extracted from tweets themselves like retweet count, favorite count, number of mentions. Also, collect more data from different OSNs. We plan to enhance the recall by applying deep learning algorithms.

# References

[1] Warner W, Hirschberg J, “Detecting hate speech on the World Wide Web”. In *Proceedings of the Second Workshop on Language in Social Media* 2012 Jun 7 (pp. 19-26). Association for Computational Linguistics.

[2] Badaro, G., Baly, R., Hajj, H., El-Hajj, W., Shaban, K. B., Habash, N., Al-Sallab, A., and Hamdi, A. (2019). A survey of opinion mining in arabic: A comprehensive system perspective covering challenges and advances in tools, resources, models, applications, and visualizations. ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP),

18(3):27.

[3] Alrefai, M., Faris, H., and Aljarah, I. (2018). Sentiment analysis for arabic language: A brief survey of approaches and techniques. International Journal of Advanced Science and Technology, 119(1):13–24.

[4] Chrysoula Themeli, George Giannakopoulos, and Nikiforos Pittaras, (). A study of text representations for Hate Speech Detection

[5] Ahmed Omar, Tarek M Mahmoud and Tarek Abd El-hafez, (03-2020). Comparative Performance of Machine Learning and Deep Learning Algorithms for Arabic Hate Speech Detection in OSNs

[6] Ibrahim Abu-Farha and Walid Magdy, Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, pages 86–90, (2020). Multitask Learning for Arabic Offensive Language and Hate-Speech Detection

[7] Mubarak, H. and Darwish, K. (2019). Arabic offensive language classification on twitter. In International Conference on Social Informatics, pages 269–276. Springer.

[8] Alakrot, A., Murray, L., and Nikolov, N. S. (2018). Towards accurate detection of offensive language in online communication in arabic. Procedia computer science, 142:315–320.

[9] Founta, A.M., Chatzakou, D., Kourtellis, N., Blackburn, J., Vakali, A., Leontiadis, I.: A unified deep learning architecture for abuse detection. In: WebSci 2019 – Proceedings of the 11th ACM Conference on Web Science, pp. 105–114 (2019).

[10] Albadi, N., Kurdi, M., and Mishra, S. (2018). Are they our brothers? analysis and detection of religious hate speech in the Arabic twitter sphere. In 2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM), pages 69–76. IEEE.

[11] Hossam Faris1, Ibrahim Aljarah1 Maria Habib1 and Pedro A. Castillo2, In Proceedings of the 9th International Conference on Pattern Recognition Applications and Methods (ICPRAM 2020), pages 453-460 (2020). Hate Speech Detection using Word Embedding and Deep Learning in the Arabic Language Context.

[12] Abeer Abuzayed Tamer Elsayed, (May, 2020). The 4thWorkshop on Open-Source Arabic Corpora and Processing Tools with a Shared Task on Offensive Language Detection, pages 109-114 (2020). Quick and Simple Approach for Detecting Hate Speech in Arabic Tweets.

[13] Bushr Haddad, Zoher Orabe, Anas Al-Abood, Nada Ghneim, (2020). Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, pages 48–52, (2020). Arabic Offensive Language Detection with Attention-based Deep Neural Networks.

[14] Mar, B.: How Much Data Do We Create Every Day? The Mind-Blowing Stats Everyone Should Read, Forbes (2018). https://www.forbes.com/sites/bernardmarr/2018/05/21/howmuch-data-do-we-create-every-day-the-mind-blowing-stats-everyone-should-read/. Accessed 12 Dec 2019.

[15] Hala Mulki, Hatem Haddad, Chedi Bechikh Ali and Halima Alshabani, Proceedings of the Third Workshop on Abusive Language Online, pages 111–118 Florence, Italy, August 1, 2019. (2019). L-HSAB: A Levantine Twitter Dataset for Hate Speech and Abusive Language.

[16] Cavnar, W.B. and J.M. Trenkle. N-gram-based text categorization. in Proceedings of SDAIR-94, 3rd annual symposium on document analysis and information retrieval. 1994. Citeseer.

[17] Ramos, J. Using tf-idf to determine word relevance in document queries. in Proceedings of the first instructional conference on machine learning. 2003. Piscataway, NJ.

[18] Mikolov, T., et al. Distributed representations of words and phrases and their compositionality. in Advances in neural information processing systems. 2013.

[19] Le, Q. and T. Mikolov. Distributed representations of sentences and documents. in International conference on machine learning. 2014.

[20] Tomas Mikolov, Kai Chen, Greg Corrado and Jeffrey Dean, (2013). Efficient Estimation of Word Representations in Vector Space.

[21] Haddad, H., Mulki, H., Oueslati, A.: T-HSAB: A Tunisian hate speech and abusive dataset. Springer International Publishing (2019).

[22] Mubarak, H., Darwish, K., Magdy, W.: Abusive language detection on Arabic social media. In: Proceedings of the First Workshop on Abusive Language Online, pp. 52–56. Association for Computational Linguistics, Stroudsburg (2017).

[23] Ghosh Chowdhury, A., Didolkar, A., Sawhney, R., Shah, R.R.: ARHNet – leveraging community interaction for detection of religious hate speech in Arabic, pp. 273–280 (2019).

[24] Mohaouchane, H., Mourhir, A., Nikolov, N.S.: Detecting offensive language on Arabic social media using deep learning. In: 2019 Sixth International Conference on Social Networks Analysis, Management and Security, pp. 466–471 (2019).

[25] Stieglitz, S., Mirbabaie, M., Ross, B., Neuberger, C.: Social media analytics – challenges in topic discovery, data collection, and data preparation. Int. J. Inf. Manag. 39(October 2017), 156–168 (2018) Stieglitz, S., Mirbabaie, M., Ross, B., Neuberger, C.: Social media analytics – challenges in topic discovery, data collection, and data preparation. Int. J. Inf. Manag. 39(October 2017), 156–168 (2018).

[26] Gilbert Badaro, Ramy Baly, Hazem M. Hajj, and Wassim El-Hajj, in ACM Transactions on Asian Language Information Processing · May 2019 (2019). A Survey of Opinion Mining in Arabic: A Comprehensive System Perspective Covering Challenges and Advances in Tools, Resources, Models, Applications, and Visualizations.

[27] Shaikh, S. and S.M. Doudpotta, Aspects Based Opinion Mining for Teacher and Course Evaluation. Sukkur IBA Journal of Computing and Mathematical Sciences, 2019. 3(1): p. 34-43.

[28] Russell, S., Norvig, P., Intelligence, A.: A modern approach. Artificial Intelligence. Prentice-Hall, Egnlewood Cliffs 25(27), 79{80 (1995).

[29] Lewis, D.D.: Naive (bayes) at forty: The independence assumption in information retrieval. In: European conference on machine learning. pp. 4{15. Springer (1998)

[30 ?????] Menard, S.W.: Applied logistic regression analysis. No. 04; e-book. (1995)

[31 ?????] Liaw, A., Wiener, M., et al.: Classification and regression by random forest. R news 2(3), 18{22 (2002).

[32] Abdelghani Dahou, Shengwu Xiong, Junwei Zhou, Mohamed Houcine Haddoud, and Pengfei Duan. 2016. Word embeddings and convolutional neural network for arabic sentiment classification. In Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers, pages 2418–2427.

[33] A. B. Soliman, K. Eissa, and S. El-Beltagy, "AraVec: A set of Arabic Word Embedding Models for use in Arabic NLP," presented at the 3rd International Conference on Arabic Computational Linguistics (ACLing 2017), Dubai, UAE, 2017.

[34] M. M. Ashi, M. A. Siddiqui, and F. Nadeem, "Pre-trained Word Embeddings for Arabic Aspect-Based Sentiment Analysis of Airline

Tweets," in *AISI*, 2018.