IOCA 2021

MDPI

Type of Article

AI-based Misogyny Detection from Arabic Levantine Twitter Tweets

Abdullah Y. Muaad[1,2], J. Hanumanthappa[1],* ,Mugahed A. Al-antari[2,3*], Bibal Benifa J V[4] and Channabasava Chola[4]

[1] Department of Studies in Computer Science, University of Mysore, Manasagangothri, Mysore 570006, India; abdullahmuaad9@gmail.com (A.Y.M.)
[2] Sana'a Community College, Sana'a 5695, Yemen
[3] Department of Computer Science and Engineering, College of Software, Kyung Hee University, Suwon-si 17104, Korea; en.mualshz@khu.ac.kr (M. A. A.)
[4] Department of Computer Science and Engineering, Indian Institute of Information Technology Kottayam, India
* Correspondence: hanumsbe@gmail.com (H.J.) and en.mualshz@khu.ac.kr (M. A. A.)

Abstract: Twitter is one of the social media platforms that is extensively used to share the public opinions. Arabic text detection system (ATDS) is a challenging computational task in the field of Natural Language Processing (NLP) using Artificial Intelligence (AI)-based techniques. The detection of misogyny in Arabic text got a lot of attention in recent years due to the racial and verbal violence against women on social media platforms. In this paper, an Arabic text recognition approach is presented for detecting misogyny from Arabic tweets. The proposed approach is evaluated using the Arabic Levantine Twitter Dataset for Misogynistic, and gained recognition accuracies of 90.0% and 89.0% for binary and multi-class tasks, respectively. The proposed approach seems to be useful in providing practical smart solutions for detecting Arabic misogyny on social media.

Keywords: Arabic language processing; Arabic Text Representation; Misogyny Detection

1. Introduction

People's express their thoughts, emotions, and feelings by means of posts on social media platforms. Recently, online misogyny considered as a harrasment is increased against Arab women on a daily basis[1][2]. An automatic misogyny detecting system is necessary for minimizing the prohibition of anti-women Arabic harmful content [2]. People are increasingly using social media platforms such as Twitter, Facebook, Google, and YouTube to communicate their various ideas and beliefs [3]. Misogyny on the internet has become a major problem that has expanded across a variety of social media platforms. Women in the Arab countries, like their peers around the world, are subjected to many forms of online misogyny. This is, unfortunately, is not compatible with the values of the islamic religion or with any other values or beliefs regarding women. Detecting such contents is crucial for understanding and predicting conflicts, understanding polarization among communities, and providing means and tools to filter or block inappropriate content [3]. The main challenges and opportunities in this field are the lack of tools with absence of resources in non-English (such as Arabic) dataset [4]. This research aims to develop a deep learning- based accurate approach to limit the misogyny problems. The lack of such studies in the Arabic perspective is an inspiration to investigate and find out practical smart solutions by designing and developing automatic identification misogyny system [5].

- The main contributions of this work are summarized as follows,
- The Arabic text is represented using the word and word embedding techniques.
- The state-of-art deep learning BERT technique is used to detect Arabic misogyny.

A comprehensive comparison study is conducted using different machine learning and deep learning techniques to achieve prominent and superior detection results.

2. Related Works

In 2020 Aggression and Misogyny Detection using BERT is proposed for three languages such as English, Hindi and Bengali[6]. The proposed model uses an attention mechanism over BERT to get the relative importance of words, followed by fully-connected layers, and a final classification layer which predicts the corresponding class[6]. The misogyny identification techniques offer satisfactory results, but the recognition of aggressiveness is still in its infancy for some languages [7]. Misogyny detection in Arabic language is still in its early stages, with only a few important contributions there[8]. In the last five years, there has been a growth in the number of researchers who are interested in automatic Arabic hate speech detection in social media. In the presented research Arabic text detection based on Misogyny is extensively studied. Starting with a comprehensive comparative study of neural network and transformer-based language models that are applied for Arabic fake news detection[9]. In terms of generalization, AraBERT v02 outperformed all other models evaluated. They advise using a gold-standard dataset annotated by humans in the future, rather than a machine-generated dataset, which may be less reliable[9]. In the same domain of detection, word2vec model is suggested to detect semantic similarity between words in Arabic, which can assist in the detection of plagiarism. The authors built the word2vec model using the OSAC corpus[10]. Here, the authors focus on creating a successful offensive tweet identification dataset. They quickly construct a training set from a seed list of offensive words. Given an autonomously generated dataset, represent a character n-gram and use a deep learning classifier to achieve a 90% F1 score [11]. A single learner machine learning approach and ensemble machine learning approach is investigated for offensive language detection in Arabic language[12]. In addition to this, transfer learning method and AraBERT is used for Arabic offensive detection datasets. The results report outperformance of Arabic monolingual BERT models over BERT multilingual models. Their results mention that limitation by effects of transfer learning on the performance of the classifiers, particularly for highly dialectic[13]. With augmentation of the data to improve text detection, the authors experimented with seven BERT-Based models and they augmented a task data set to identify the sentiment of a tweet or detect if a tweet is sarcasm [14]. Their experiments were based on fine-tuning seven BERT-based models with data augmentation to solve the imbalanced data problem. For both tasks, the MARBERT BERT-based model with data augmentation outperformed other models with an increase of the F-score by 15%. Regarding the influence of preprocessing in text detection, a simple but intuitive detection system based on the investigation of a number of preprocessing steps and their combinations is addressed [15]. Here, a comparison between LSVC and BiLSTM classifiers was conducted. The detection of misogyny in Arabic text was presented using the Arabic Levantine Twitter dataset for Misogynistic language (LeT-Mi) which is the first benchmark dataset for Arabic misogyny. They employ an MTL configuration to investigate its effect on the tasks. They present an experimental evaluation of several machine learning systems, including SOTA systems. The result for accuracy is equal to 88 and presented an approach based on stylistic and specific topic information for the detection of misogyny, exploring the several aspects of misogynistic Spanish and English user generated texts on Twitter Section (Heading 1) [16]. Finally, an approach based on character level for Arabic text utilizing convolutional neural network (CNN) is presented to solve many problems such as difficulties in preprocessing etc [17].

3. Proposed Model

3.1. ATDS Architecture

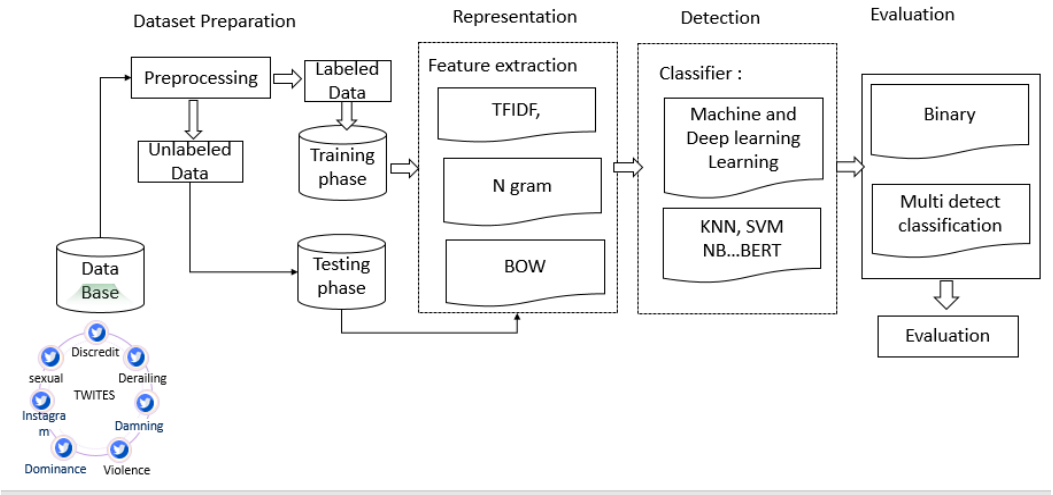

Figure 1. Architecture of the Arabic text detection system (ATDS): Abstract view. .

3.2. Preprocessing

Pre-processing technique is most commonly used for preparing raw data into specific input data format, which could be useful for machine learning and deep learning techniques. The main purpose of preprocessing is to clean the dataset regarding stop-words, punctuation, poor spellings, slang, and other undesired words abound in text data. This unwanted noise and language may have a negative impact on the recognition performance of the Arabic misogyny detection task. In this work, we eliminate all the non-Arabic words, stop words, and punctuations through the following steps,

a) Tokenization:
This process is used to convert the Arabic text (sentence) into tokens or words. Tokenized documents can be transformed into sentences, and sentences can be converted into tokens. Tokenization divides a text sequence into words, symbols, phrases, or tokens[18].

b) Normalization:
the normalization is to make all word in same from and there are many techniques such as stemming etc. Finally, normalization takes in by rules or regular expressions

c) Stop Word Elimination:
In the text preprocessing task, there are numerous terms that have no critical meaning but appear frequently in a document. It refers to words that do not help to increase performance because they do not provide much information for the sentiment classification task; therefore, stop words should be removed before the feature selection proces.

d) Stemming
One word can appear in many distinct forms, but the semantic meaning remains the same. Stemming is the process of replacing and removing suffixes and affixes to obtain the root, base, or stem word

e) Lemmatization
The goal of lemmatization is the same as stemming: to reduce words to their base or root words. However, in lemmatization, the inflection of words is not

simply cut off; rather, it leverages lexical information to turn words into their base forms[19].

3.3. Representation

After Arabic text preprocessing, the data is transformed to be in a specific structure style for representation purpose. To perform this, bag-of-words (BOW), term frequency-inverse document frequency (TFIDF) are used for data representation with traditional machine learning techniques. For deep learning techniques, we use a new technique called word embedding in bidirectional encoder representations from transformers (BERT). Instead of the basic language task, BERT is trained with two tasks to encourage bidirectional prediction and sentence-level understanding [21].

3.4. Text Detection

Detection of text and classify to true labelled class based on their content is known as classification. Several works have been reported here based on text classification using different algorithms as we will explain in part 5. There are many algorithm have been implemented as follow:

- Passive Aggressive Classifier

Passive-Aggressive algorithms are a family of Machine learning algorithms that are popularly used in big data applications. Passive-Aggressive algorithms are generally used for large-scale learning. It is one of the online-learning algorithms. In online machine learning algorithms, the input data comes in sequential order and the machine learning model is updated sequentially, as opposed to conventional batch learning, where the entire training dataset is used at once[20].

- Logistic Regression

Logistic regression is a statistical model that in its basic form uses a logistic function to model a binary dependent variable, although many more complex extensions exist[19].

- Random Forest Classifier

The term “Random Forest Classifier” refers to the classification algorithm made up of several decision trees. The algorithm uses randomness to build each individual tree to promote uncorrelated forests, which then uses the forest's predictive powers to make accurate decisions[19].

- Linear SVC

The support vector machine (SVM) classifiers is one of the commonly used algorithms for text classification due to its good performance. SVM is a non-probabilistic binary linear classification algorithm which performs by plotting the training data in multi-dimensional space. Then SVM categorizes the classes with a hyper-plane. The algorithm will add a new dimension if the classes can not be separated linearly in multi-dimensional space to separate the classes. This process will continue until training data can be categorized into two different classes[19].

- Decision Tree Classifier

Decision Trees are also used in tandem when you are building a Random Forest classifier which is a culmination of multiple Decision Trees working together to classify a record based on majority vote. A Decision Tree is constructed by asking a seris of questions with respect to a record of the dataset we have got[19].

- K Neighbors Classifier

KNN works by finding the distances between a query and all the examples in the data, selecting the specified number examples (K) closest to the query, then votes for the most frequent label (in the case of classification) or averages the labels (in the case of regression)[19].

- ARABERTv2

AraBERT is an Arabic pretrained language model based on Google's BERT architecture. AraBERT uses the same BERT-Base config[21]

4. Experimental analysis

4.1. Dataset

The dataset [1] is Unbalanced by limiting the number of articles in each specific category as summarized in Table 1.

Table1: Data Distribution Per Class in binary classification

Type of Misogyny	نوع كراهية النساء	No of Tweets
None	لاشي	3,061
misogyny	كراهية النساء	4,805

the author classifies his data as we mention below:

1. Damning (Damn): tweets under this class contain cursing content.
2. Derailing (Der): tweets under this class combine justification of women abuse or mistreatment.
3. Discredit (Disc): tweets under this class bear slurs and offensive language against women.
4. Dominance (Dom): tweets under this class imply the superiority of men over women.
5. Sexual Harassment (Harass): tweets under this class describe sexual advances and sexual nature abuse.
6. Stereotyping & Objectification (Obj): tweets under this class promote a fixed image of women or describe women's physical appeal.
7. Threat of Violence (Vio): tweets under this class have an intimidating content with threats of physical violence.
8. None: if no misogynistic behaviors exist.

Table 2. Data Distribution Per Class in multi classification.

Type of Misogyny	نوع كراهية النساء	No of Tweets
Discredit	تشويه السمعة	2,327
Stereo typing & objectification	الكتابة والصياغة المجسمة	290
Damning	اللعنة	256
Threat of violence	التهديد بالعنف	175
Derailing	الخروج عن السكة	59
Dominance	هيمنة	38
Sexual harassment	التحرش الجنسي	17
None	لاشي	3,388

4.2. Implementation Environment

To perform all experiments in this study, we use a PC with the following specifications: Intel R © Core(TM) i7-6850 K processor with 4 GB RAM, 3.360 GHz frequency. The algorithms such as Passive Aggressive Classifier, Logistic Regression, Logistic Regression, Random Forest Classifier, K Neighbors Classifier, and linear SVC are implemented herein using Python 3.8.0 programming with Anaconda [Jupyter notebook]. The Python-based ML libraries such as NLTK, pandas, and scikit-learn are utilized to investigate the performance metrics by the proposed methods at the same time tensorflow and keras in colab have been used to implement ARABERTv2. The results and discussions concerning various techniques incorporated are highlighted in the subsequent sections. The code will be available in our account in GitHub[1].

[1] https://github.com/abdullahmuaad8

4.3. Evaluation metrics

To assess our proposed system, we use the following indices,

Recall is calculated by dividing the number of true positive (TP) observations by the total number of observations (TP+FN).

Specificity is defined as the proportion of true positive (TP) observations to total positive forecasted values (TP+FP).

F1-score is the weighted average of recall and precision, which means that the F1-score includes both FPs and FNs.

Accuracy is defined as the simple ratio of accurately predicted observations to total observations.

The definition formula of all these metrics are defined as follow,

$$Recall/Sensitivity\ (Re) = \frac{TP}{TP + FN}, \quad (1)$$

$$Specificity\ (Sp) = \frac{TN}{TN + FP}, \quad (2)$$

$$F1 - score\ (F - M)\ = \frac{2 \cdot TP}{2 \cdot TP + FP + FN}, \quad (3)$$

$$Overall\ accuracy\ (Az) = \frac{TP + TN}{TP + FN + TN + FP}, \quad (4)$$

where TP, TN, FP, and FN are defined to represent the number of true positive, true negative, false positive, and false negative detections, respectively. To derive all of these parameters, a multidimensional confusion matrix is used.

5. Results and discussion

The results and discussions concerning various techniques incorporated are highlighted in this section, we describe our experiments on this data. We evaluate the performance of all algorithms on this data. We design our experiments at two levels (tasks):

1. Misogyny identification (Binary): Tweets contents are classified into misogynistic and non misogynistic. This requires merging the seven categories of misogyny into the misogyny class.

2. Categories classification (Multi-class): Tweets are classified into 8 categories: discredit, dominance, damning, derailing, sexual harassment, stereotyping and objectification, and threat of Violence, or non-misogynistic. In addition, we found that Linear SVC outperformed all compared models in terms of generalization for machine learning and BERTv2 for deep learning technique.

5.1. Binary Classification

The results of the misogyny identification task have been shown in table 3. In terms of accuracy,precision, recall and F-measure the Linear SVC model outperforms the others. We also can observe that the model outperforms all the other models except Random Forest Classifier that works better in terms of recall . At the same time we have been used one of the transfer learning called ARABERTv2 which gives excellent accuracy but the time was much compared to machine learning.

Table 3. Arabic misogyny detection Evaluation results for binary classification tasks.

Method	Az	Sp	Re	F-M
Passive Aggressive Classifier	81	84	86	85
Logistic Regression	81.50	81	90	86
Random Forest Classifier	62	62	**100**	76
Linear SVC	**83**	**85**	88	**86**
Decision Tree Classifier	70	74	78	76
K Neighbors Classifier	65	64	98	78

ARABERTv2	**90**	**-**	**-**	**-**

5.2. Multi Classification

The results of the misogyny identification task have been shown in Table 4. In terms of accuracy, the Linear SVC model outperforms the others. According to the results, the typical machine learning Random Forest Classifier model performance is poor. At the same time we have been using one of the transfer learning called ARABERTv2 which gives excellent accuracy but the time consumption was more as compared to machine learning methods.

Finally, we'd like to point out that the data set was unbalanced. For example, as shown in table 1, the class Sexual harassment has only 17 comments, which means that learning the pattern for these classes is very limited. As a result, we recommend that the number of comments in this data be increased as a future project

Table 4. Arabic misogyny detection Evaluation results for multiclass classification tasks.

Method	Az	Sp	Re	F-M
Passive Aggressive Classifier	72	72	72	72
Logistic Regression	69	68	68	68
Random Forest Classifier	40	39	39	39
Linear SVC	**74**	**73**	**73**	**73**
Decision Tree Classifier	56	56	56	56
K Neighbors Classifier	44	43	43	43
ARABERTv2	**89**	**-**	**-**	**-**

6. Conclusion

The problem of misogyny has become a major problem for Arab women. In this work, we introduce a model for detection of misogyny of Arabic text. We have carried out our work utilizing a data set called (Arabic Levantine Twitter Dataset for Misogynistic). Our results provide excellent accuracy equal to 83% using machine learning for detection and classification tasks. This article proves that, there are many open issues need to solve staring by limitation of benchmark dataset, lexicons of Arabic text in general and especially for misogyny of women at the same time the difficulty of nature of Arabic language in morphology and delicate. Then augmentation of data such as oversampling to solve unbalance of classes could get better performance. Finally, there is a need to study the correlation between hate speech, misogyny and the problem of mix language for future work.

References

[1] I. Abu Farha and W. Magdy, "Multitask Learning for {A}rabic Offensive Language and Hate-Speech Detection," Proc. 4th Work. Open-Source Arab. Corpora Process. Tools, with a Shar. Task Offensive Lang. Detect., no. May, pp. 86–90, 2020, [Online]. Available: https://www.aclweb.org/anthology/2020.osact-1.14.

[2] H. Mulki and B. Ghanem, "Let-Mi: An Arabic Levantine Twitter Dataset for Misogynistic Language," pp. 154–163, 2021, [Online]. Available: http://arxiv.org/abs/2103.10195.

[3] M. Alkhair et al., "An Arabic Corpus of Fake News : Collection , Analysis and Classification To cite this version : HAL Id : hal-02314246 An Arabic Corpus of Fake News : Collection , Analysis and Classification," 2019.

[4] M. S. Jahan and M. Oussalah, "A systematic review of Hate Speech automatic detection using Natural Language Processing," 2021, [Online]. Available: http://arxiv.org/abs/2106.00742.

[5] R. Alshalan and H. Al-Khalifa, "A deep learning approach for automatic hate speech detection in the saudi twittersphere," Appl. Sci., vol. 10, no. 23, pp. 1–16, 2020, doi: 10.3390/app10238614.

[6] N. Safi Samghabadi, P. Patwa, S. PYKL, P. Mukherjee, A. Das, and T. Solorio, "Aggression and Misogyny Detection using

{BERT}: A Multi-Task Approach," Proc. Second Work. Trolling, Aggress. Cyberbullying, no. May, pp. 126–131, 2020, [Online]. Available: https://www.aclweb.org/anthology/2020.trac-1.20.

[7] E. Fersini, D. Nozza, and P. Rosso, "AMI @ EVALITA2020: Automatic misogyny identification," CEUR Workshop Proc., vol. 2765, 2020, doi: 10.4000/books.aaccademia.6764.

[8] A. Hengle, A. Kshirsagar, S. Desai, and M. Marathe, "Combining Context-Free and Contextualized Representations for Arabic Sarcasm Detection and Sentiment Identification," 2021, [Online]. Available: http://arxiv.org/abs/2103.05683.

[9] M. Al-Yahya, H. Al-Khalifa, H. Al-Baity, D. Alsaeed, and A. Essam, "Arabic Fake News Detection: Comparative Study of Neural Networks and Transformer-Based Approaches," Complexity, vol. 2021, 2021, doi: 10.1155/2021/5516945.

[10] D. Suleiman, A. Awajan, and N. Al-Madi, "Deep learning based technique for plagiarism detection in Arabic texts," Proc. - 2017 Int. Conf. New Trends Comput. Sci. ICTCS 2017, vol. 2018-Janua, pp. 216–222, 2017, doi: 10.1109/ICTCS.2017.42.

[11] H. Mubarak and K. Darwish, "Arabic Offensive Language Classification on Twitter," Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 11864 LNCS, no. April 2020, pp. 269–276, 2019, doi: 10.1007/978-3-030-34971-4_18.

[12] F. Husain, "Arabic Offensive Language Detection Using Machine Learning and Ensemble Machine Learning Approaches," 2020, [Online]. Available: http://arxiv.org/abs/2005.08946.

[13] F. Husain and O. Uzuner, "Transfer Learning Approach for Arabic Offensive Language Detection System -- BERT-Based Model," 2021, [Online]. Available: http://arxiv.org/abs/2102.05708.

[14] A. Abuzayed and H. Al-Khalifa, "Sarcasm and Sentiment Detection In {A}rabic Tweets Using {BERT}-based Models and Data Augmentation," Proc. Sixth Arab. Nat. Lang. Process. Work., pp. 312–317, 2021, [Online]. Available: https://www.aclweb.org/anthology/2021.wanlp-1.38.

[15] M. Lichouri, M. Abbas, B. Benaziz, A. Zitouni, and K. Lounnas, "Preprocessing Solutions for Detection of Sarcasm and Sentiment for {A}rabic," Proc. Sixth Arab. Nat. Lang. Process. Work., pp. 376–380, 2021, [Online]. Available: https://www.aclweb.org/anthology/2021.wanlp-1.49.

[16] S. Frenda, B. Ghanem, and M. Montes-y-Gómez, "Exploration of misogyny in Spanish and english tweets," CEUR Workshop Proc., vol. 2150, pp. 260–267, 2018.

[17] Abdullah Muaad, Haznumanthappa Jayappa, Mugahed Al-Antari, Sungyoung Lee. ArCAR: A Novel Deep Learning Computer-Aided Recognition for Character-Level Arabic Text Representation and Recognition. Algorithms. 2021; 14 (7):216.

[18] Z. Alyafeai, M. S. Al-shaibani, M. Ghaleb, and I. Ahmad, "Evaluating Various Tokenizers for Arabic Text Classification," vol. 5, 2021, [Online]. Available: http://arxiv.org/abs/2106.07540

[19] K. Kowsari, K. J. Meimandi, M. Heidarysafa, S. Mendu, L. Barnes, and D. Brown, "Text classification algorithms: A survey," Inf., vol. 10, no. 4, pp. 1–68, 2019, doi: 10.3390/info10040150.

[20] J. Huang, "Detecting fake news with machine learning," J. Phys. Conf. Ser., vol. 1693, no. 1, 2020, doi: 10.1088/1742-6596/1693/1/012158.

[21] W. Antoun, F. Baly, and H. Hajj, "AraBERT: Transformer-based Model for Arabic Language Understanding," arXiv, 2020.