

Deep Learning for emotion analysis in Arabic tweets

ENAS ABDEL HAKIM KHALIL (✉ enaskhalil@gmail.com)

National Research Centre <https://orcid.org/0000-0002-3050-2656>

Enas .M.F. El Houby

National Research Centre

Hoda .k. Mohamed

Ain Shams University Faculty of Engineering

Research

Keywords: Deep Learning, BiLSTM, multilabel emotion classification, word embedding, Aravec, ARLSTEM

Posted Date: June 30th, 2021

DOI: <https://doi.org/10.21203/rs.3.rs-631537/v1>

License:   This work is licensed under a Creative Commons Attribution 4.0 International License. [Read Full License](#)

Abstract

Expressing our emotions using text and emojis expressions became widespread through social media such as Facebook, Instagram, Twitter, Weibo, and LinkedIn. Nowadays, both organizations and individuals are interested in using social media to analyze people's opinions and extract sentiments and emotions. We proposed a model for multilabel emotion classification, using a bidirectional Long Short-term Memory BiLSTM deep network. It is evaluated on the Arabic tweets' dataset provided by SemEval 2018 for the E-c task. Several preprocessing steps, including ARLSTEM with some modifications, replacing emojis with corresponding text meaning from a manually built lexicon, and feature vector representation using Aravec word embedding is applied. The novelty in our research that it examines the effect of hyperparameter tuning on model performance, and it uses BiLSTM in all of its deep neural network layers. The proposed model achieves a comparable performance with state-of-the-art models using different machine learning and deep learning techniques. The system achieves about 9% enhancement in validation accuracy compared with the last best model in the same task using Support Vector classifier SVC; it outperforms the other deep neural networks (UNCCTeam) based on fully connected layers in micro F1 metric of about 4.4%.

1. Introduction

With the rapid growth of different web applications as E-commerce platforms and substantial social media comments in various fields, an urgent need to deal with this massive amount of web data and automatically extract helpful information arises. Sentiment Analysis models are of great importance in this task. Sentiment analysis, sometimes interchangeably called opinion mining, is a computational field within Natural Language Processing (NLP) concerned with people's sentiments and opinions towards objects such as services, persons, products, events, organizations, and topics [1]. Thanks to the availability of high-performance computational computers, which allows using different machine learning techniques, especially deep learning ones, to build high-performance, robust automatic sentiment analysis models. While sentiment analysis aims to detect positive, neutral, or negative opinions from text, emotion analysis is one of the most common sentiment analysis tasks, which aims to see and recognize types of feelings through text expression.

Emotions are mainly expressed using language; they are complex and nuanced, although they are commonplace and familiar. Humans can understand hundreds of different emotions. According to the basic emotion model [2], [3], and [4], some emotions, such as joy, fear, and sadness, are more fundamental than others and can be expressed in different ways. Our degree of utterances, for example, can reveal that we are so sad, slightly angry. The term affect refers to various categories of emotions as joy, fear, arousal, and valence. Natural language applications are vital in knowing the affectual states of people, including intensities and categories of their emotions. The detection of emotions, in turn, is of great importance in many fields as public health, marketing, disaster management, public policy, and political issues [5]. Two typical categorical representations for emotions exist Ekman representation [2] which includes anger, happiness, disgust, surprise, fear, and sadness, and Plutchik model [3], [4], which includes Ekman's six emotions in addition to two labels: trust and anticipation. Emotion recognition systems from facial expressions and images have been widely used [6], [7], [8], and [9]. Hand gesture recognition is used as a part of Human action recognition (HAR)[10]. Emotion recognition models can be also explored based on human-

computer interaction [11],[12], and [13]. Emotion-rich textual data from social networks can be processed for a variety of real-world applications, including [14], [15]; [16], [17],[18], and[19].

The research interest in Arabic sentiment analysis increases drastically due to the vast number of Arabic language users on the internet. However, emotion recognition from Arabic text still needs enormous efforts to develop more accurate emotion mining models in both MSA and dialectal Arabic using a large-scale emotion lexicon.

The fast rise of social media platforms (e.g., Twitter, Facebook) drew the researchers' attention to the technique of "Affect detection from text," which enables users to communicate their sentiments, emotions, and ideas via text. SemEval, an international workshop on semantic evaluation developed from SensEval (Evaluation Exercises for the Semantic Analysis of Text, Organized by ACL-SIGLEX) and aimed to evaluate semantic analysis systems, produced an excellent value task, "The SemEval-2018 Task 1: Affect in Tweets" [5] in its 12th workshop on semantic evaluation in 2018. This task includes five subtasks in which automatic systems infer a person's emotional state from their tweet for each task in English, Arabic, and Spanish. This paper performs multilabel emotion classification in the Arabic language using SemEval-2018 Task1-datasets. In achieving this goal, several preprocessing steps, including removing non-Arabic words, removing digits, removing stop-word, and applying a robust stemmer ARLSTM, are implemented. Then building the feature vectors using Aravec (A set of Arabic Word Embedding Models for use in Arabic NLP) [20] for tweet representation; these embedded vectors are then fed to a multilayer Bidirectional LSTM network.

The remainder of the paper is organized as follows: Sect. 2 discusses a related body of work, including several Arabic text emotion analysis methods. Section 3 describes the approach we propose for evaluating emotional content in tweets. Section 4 summarises our findings and examines the most significant findings. Finally, Sect. 5 brings this research to a close and makes recommendations for future work.

2. Related Work

Learning the users' emotions is essential in many applications, such as social robots used as communication assistance for education and entertainment [9] This article describes an acceptable and natural interaction between Social Robots and children. The thermal facial reaction of youngsters, i.e., the nose tip temperature signal, was recorded and classified in real-time using the Mio Amico Robot during an experimental session. The categorization was performed by comparing the thermal signal analysis-classified emotional state to the emotional state recorded by Face reader 7. An empathic robot in [7] to recognize human emotions through facial expressions and automatically respond to these specific emotional states produced a state-of-the-art accuracy rate of 95.58%. Using Convolutional Neural Network CNN and a bank of Gabor filters in different experiments for feature representation and employs SVMs and MLPs as classifiers.

Customer feedback detection, as in [13] where a multimodal affect recognition system developed to classify whether a customer likes or dislikes a product examined at a counter, by analyzing the consumer's facial expression, hand gestures, body posture, and voice after testing the product. Hand gesture recognition is a component of Human action recognition (HAR) and is widely used in scientific research; it is critical for interacting with deaf individuals. [10] proposes an approach divided into transfer learning through Alex Nets

and hyperparameter tuning through ABC, GA, and PSO algorithms. The methodology produced effective outcomes with an average accuracy of 98.09 percent, beating the best work in the medical sector in [21]; computational analysis techniques are used to measure the emotional facial expression of people who have Parkinson's disease (PD). Since PD experiences hypomania, which often causes a reduction in facial expression, it is important to examine an experimental pilot work for masked face detection in PD. This experiment achieved an accuracy of 85% on the testing images using a deep learning-based model.

A novel methodology for incorporating human emotion into intelligent computer systems is presented [14]. It has been proposed as a method to elicit emotional information from users. A hybrid cloud intelligence model using an adaptive fuzzy method with a high degree of interpretability achieves a satisfactory performance accuracy of 81.39% using Facebook's sentiment analysis API.

The use of emotion lexicons is of great importance in the emotion classification task. Many lexicons in different languages have been built as [16],[22], [23], [19], [24], [25], [26] in English, Polish and French. In Arabic, many efforts have been recently made in building emotion lexicons as ArSEL (Arabic Sentiment and Emotion Lexicon)[18]., ArSEL has been constructed automatically by using three lexical resources: epecheMood, English WordNet and ArSenL. These lexicons helped in improving the sentiment classification model accuracy.

Multilabel emotion classification is a hot topic in emotion analysis tasks since it represents real-life situations where the human may express a mixture of emotions in his text simultaneously. For example, the text may express happiness, love, optimism, or maybe sadness and pessimism, so it is more beneficial to build such models with more than one output emotion for each input text. The following few paragraphs below browse some recent efforts in Arabic multilabel emotion classification:

EMA (Emotion Mining in Arabic) [16] performs Emotion and Sentiment mining on Arabic tweets. First, preprocessing steps are performed, first applying normalization rules adopted by [27], including removing diacritics or taskeel and hamza removal, then removing elongations and non-Arabic letters. Next, most frequent emojis have been replaced with the corresponding Arabic word using a manually created lexicon to replace each emoji. Finally, using ARLSTEM [28] for stemming. Then in the feature selection stage, the author tried different features separately, but the word embedding from AraVec proved to be the best feature. The tweet is finally classified either as neutral or as one or more of 11 emotions (anger, disgust, anticipation, joy, love, optimism, fear, pessimism, sadness, trust, surprise). Linear SVC performed best among all classifiers tested, with a test accuracy of 0.489.

TW-Star [29] uses different preprocessing stemming (Stem), lemmatization (Lem), stop words removal (Stop), and common emoji recognition (Emo). The preprocessed tweets are then classified by a multilabel classifier based on Binary Relevance (BR) using the Support Vector Machines (SVM) with Term Frequency Inverse Document Frequency (TF-IDF) features. Several experiments with different combinations of preprocessing achieved the best results of accuracy 0.465 using a combined of (Emo + Stem + Stop).

TeamUNNC [30] performs tokenization, removal of white spaces, and treating punctuations as individual words. AraVec [20] combined with Affective Tweets Weka-package

list as ⚡️ instead of ⚡ are not considered in our updated list of stop words for not to increase ambiguity.

- Creating of emoji Lexicon:

A lexicon having the most shared tweet emojis is manually created, where each emotion is transcribed to its corresponding Arabic word. The lexicon consisted of 118 entry words that replaced emojis. Emojis are replaced with related meanings to emotions.

- Stemming:

Further normalization step is stemming. It concerns reducing the word to its standard form. The stemming process here is performed using Robust Arabic Light Stemmer ARLSTEM [28]. With social medial data (tweets data), using a word stem is more valuable than using its lemma because tweets are mostly in dialectal Arabic, not in Modern Standard Arabic MSA form. At the same time, MSA data is used to train the majority of Arabic morphological analyzers [17]. The ARLstem normalizes the word by removing diacritics. Although this process causes ambiguity in word semantics, it is interesting since it facilitates the stemming process. ARLstem replaces hamzated Alif with Alif, Alif Maqsura with Yaa, and removes Waaw at the beginning. Prefixes from the words' beginning and suffixes from the word's end are trimmed. Stemming also includes transforming the word from the feminine form to the masculine form, Stem the verb prefixes and suffixes, or both. We exclude "⚡️" and the derivations as "⚡️", "⚡️", "⚡️", "⚡️" from any text normalization or stemming and replace all this word derivations with the word "⚡️" and this minor modification is found to have a good impact on the performance, as the count of occurrence of this word is found to represents about 16% of the number of words in our data set

2. Feature extraction:

Word embedding generates the feature vectors. For word embedding, we employ AraVec [28]. AraVec is a large-scale dataset (approximately 205,000 words) that comprises different Arabic dialects and is trained on the Twitter data domain. Word embedding is proved to be decisive as it overcomes the sparsity problem in n-grams models and simplifies semantics by giving identical representations for words that may exist in the same context. The pre-trained model "Twitter-CBOW/tweets_cbow_300" loaded by gensim libraries in python has been used,

in which a 300 dimensions real numbers vector represents the word; the tweet embeddings are calculated by taking the average of its all words embedding. The average embedded vector of each tweet is then inputted to the classifier and classified either as neutral or as one or more of 11 emotions (disgust, anger, fear, pessimism, anticipation, joy, love, optimism, sadness, trust, surprise).

3. Network Architecture:

We build a deep learning model of Recurrent Neural Network RNN, mainly BiLSTM layers. The model is built using Keras libraries under the tensorflow2.3.0 platform and python 3.8. A Bidirectional LSTM, or simply BiLSTM, is a sequence model with two LSTMs: forwarding and backward direction input. BiLSTM increases the performance of the neural network and improves the context (e.g., knowing the following and

the preceding word in a sentence). As a result, it usually learns faster than the one-directional approach, although it depends on the task. The basic structure of BiLSTM is shown in figure 2.

The proposed model contains three bidirectional LSTM layers with 300, 200, and 50 inputs, respectively, with a Relu activation function in each. A dropout layer follows each Bilstm, with three Drop out layers, one after each BiLSTM layer. Following the first BiLSTM is a repeater layer. The last layer in the model is a Dense layer with 11 outputs corresponding to the eleven emotions. The activation is a sigmoid function so that it gives a probability of each emotion. We approximate the values to 0, 1 class for each of the eleven outputs representing the eleven emotions. After several experiments, the parameters are tuned as follows: The learning rate is adjusted to a value of 0.001, optimizer algorithm "Adam" with loss "mse" and "accuracy" metrics, the network structure is shown in figure 3.

4. Experimental Results

Our proposed model has been evaluated using the following metrics[3]:

- Multilabel accuracy (or Jaccard index): "Multilabel accuracy is defined as the size of the intersection of the predicted and actual label sets divided by the size of their union." It is computed for each tweet t and then is averaged over all tweets in the dataset T :

$$Accuracy = \frac{1}{|T|} \sum_{t \in T} \frac{Gt \cap \hat{P}t}{Gt \cup \hat{P}t} \quad (1)$$

Where Pt is the set of the predicted labels for tweet t , Gt is the set of the actual(gold) labels for tweet t , and T is the set of tweets

- Precision: It is the number of true positive results divided by the number of all positive results, including those not identified correctly,

$$PPV = \frac{TP}{TP+FP} \quad (2)$$

Precision is also known as positive predictive value (PPV).

Recall: The number of true positive results is divided by the number of all samples that should have been identified as positive; recall is also known as sensitivity in diagnostic binary classification.

$$Recall(also\ called\ sensitivity\ or\ true\ Positive\ Rate\ TPR) \ TPR = \frac{TP}{P} = \frac{TP}{TP + FN} \quad (3)$$

F1 score is the harmonic mean of precision and recall

$$F1\ score = 2 \cdot \frac{PPV \times TPR}{PPV + TPR} \quad (4)$$

- Micro-averaged metrics are different from the overall accuracy when the classifications are multi-labeled, so it is essential to clarify the difference between Micro- and macro-averages (for whatever metrics like

Loading [MathJax]/jax/output/CommonHTML/jax.js age, the metric is independently computed for each class. Then

the average is calculated (so, all classes are treated equally), while micro-average combines the contributions of all classes to compute the average metric. Thus, Micro-average is preferred in a multi-class classification setup, especially in imbalance class cases (i.e., having many more examples of one class than other classes).

- Micro-averaged F-score is computed as follows:

$$\text{Micro-avgPrecision}(\text{micro-P}) = \frac{\sum_{e \in E} \text{number of tweets correctly assigned to emotion classe}}{\sum_{e \in E} \text{number of tweets assigned to emotion classe}} \quad (5)$$

$$\text{Micro-avgRecall}(\text{micro-R}) = \frac{\sum_{e \in E} \text{number of tweets correctly assigned to emotion classe}}{\sum_{e \in E} \text{number of tweets in emotion classe}} \quad (6)$$

$$\text{Micro-avgF} = \frac{2 \times \text{micro-P} \times \text{micro-R}}{\text{micro-P} + \text{micro-R}} \quad (7)$$

- Where E is the given set of eleven emotions
- Macro-averaged F-score is calculated as follows:

$$\text{Precision}(P_e) = \frac{\text{number of tweets correctly assigned to emotion classe}}{\text{number of tweets assigned to emotion classe}} \quad (8)$$

$$\text{Recall}(R_e) = \frac{\text{number of tweets correctly assigned to emotion classe}}{\text{number of tweets in emotion classe}} \quad (9)$$

$$F_e = \frac{2 \times P_e \times R_e}{P_e + R_e} \quad (10)$$

$$\text{Macro-avgF} = \frac{1}{|E|} \sum_{e \in E} F_e \quad (11)$$

The experiments are conducted with cross-validation, with the number of splits $k = 10$, each time one split is taken as a test, another nine splits is for training, and the number of repeats = 3 giving a total of 30 experiments. The effect of changing hyperparameters is investigated and recorded through Tables 1 to 5.

Table 1 gives different experimental results and shows the effect of the emoji lexicon on performance; the number of Epochs is set to 20, Batch size = 32, three dropout rates 0.5,0.5,0.5, and Adam optimization. As it is shown in the table, the results using lexicon are better.

The change in dropout rate values affects the results slightly, but an observable reduction in accuracy when not using dropout as reported in Table 2, using a number of epochs = 20, batch size = 32, learning rate = 0.001, and Adam optimization.

Evaluation of model using different batch sizes is also reported in Table 3. For our data set size, the optimal batch size is 32 or 64 although, further experiments using higher epoch value proved that the batch size of 32 is more optimal for this model.

In Table 4 a comparison between two batch sizes 32,64 is implemented using a higher number of epochs,50 epochs which proved that the batch size of 32 is more optimal for our model.

The effect of changing the number of epochs is reported in Table 5, the best results obtained using 50 epochs.

Table 6 tests two optimizers with lexicon emoji, dropout rates0.5,0.5,0.5, epochs = 50, learning rate 0.001, and a batch size = 32; Adam optimizer shows better accuracy than Nadam. For all experiments, it is found that the best Learning rate value is set to 0.001.

Table 7 gives the impact of whether or not to exclude the word "عربي" and the derivations as "عربي", "عربي", "عربي", "عربي", "عربي" from any text normalization or stemming. Leaving these words without any preprocessing has a positive impact on the accuracy of about 0.8%

Table 8 shows a comparison between the proposed model (with Adam optimizers, lexicon emoji, dropout rates0.5,0.5,0.5, epochs = 50, learning rate 0.001, and batch size = 32) with other models. The system achieves about 9% enhancement in validation accuracy compared with the last best model in the same task using Support Vector classifier SVC; it outperforms the other deep neural networks (UNCC) based on fully connected layers in micro F1 about 4.4%.

5. Conclusions And Future Work

A method is proposed for building a deep learning model for multilabel emotion classification in Arabic tweets. Using SemEval2018 Task1 dataset. We follow several preprocessing steps, including normalization, stemming, replacing the most common emojis with their corresponding meanings using a manually created lexicon of emojis; for features selection, word embedding proved to be the best technique. Aravec pre-trained word embedding model with CBOW is used to build 300 dimension word vectors for each word in our data then the average embedded word vector is calculated for each tweet. Then bidirectional LSTM model is used for classification. The proposed method produced comparable results with SVM, RF, and fully connected deep NN; it achieved 9% improvement in validation accuracy than best obtained by SVM one. BiLSTM increases the amount of information through a two-way network and improves the context used by the network. The effect of hyperparameter tuning is studied experimentally since the grid search approach is not supported by Keras

libraries in the LSTM model. Further improvements in preprocessing like removing ambiguity results from stemming the nouns ending with, es , ies usually as a $\text{}$, and words ending with s , ies as $\text{}$ plural, and applying more restricted grammatical rules will help so much in model performance enhancement. The effect of using different deep learning models as convolution neural networks will also be investigated.

Table 1
Results of the proposed model with and without

Emoji lexicon	Macro-F1	Micro-F1	Micro Precision	Micro Recall	Multilabel accuracy (Jaccard index)
No lexicon	0.402	0.58	0.693	0.499	0.409
lexicon	0.421	0.606	0.711	0.528	0.435
<i>emoji lexicon</i>					

Table 2
Results of proposed models with different dropouts

Dropouts	Macro-F1	Micro-F1	Micro Precision	Micro Recall	Jaccard index
No dropout	0.447	0.596	0.643	0.556	42.5
0.3,0.3,0.3	0.452	0.604	0.646	0.567	0.433
0.5,0.5,0.5	0.421	0.606	0.711	0.528	0.435

Table 3
Results of the proposed model using different batch sizes

Batch size	Macro-F1	Micro-F1	Micro_Precision	Micro_Recall	Jaccard-index
8	0.415	0.599	0.705	0.522	0.428
16	0.42	0.603	0.703	0.528	0.432
32	0.421	0.606	0.711	0.528	0.435
64	0.418	0.606	0.717	0.525	0.435
128	0.41	0.601	0.724	0.515	0.43
256	0.41	0.599	0.727	0.509	0.427
512	0.382	0.58	0.73	0.481	0.409

Table 4
Results of the proposed model with different epochs

epochs	Macro-F1	Micro-F1	Micro Precision	Micro Recall	Multilabel accuracy (Jaccard index)
10	0.402	0.591	0.724	0.501	0.420
20	0.421	0.606	0.711	0.528	0.435
50	0.445	0.615	0.681	0.561	0.444
100	0.451	0.609	0.666	0.562	0.438

Table 5
The effect of using two different values of batch size with epoch value 50

Batch size	Macro-F1	Micro-F1	Micro Precision	Micro Recall	Multilabel accuracy (Jaccard index)
32	0.445	0.615	0.681	0.561	0.444
64	0.442	0.611	0.677	0.558	0.441

Table 6
The effect of changing the optimizer with epochs = 50

Optimizer	Macro-F1	Micro-F1	Micro_Precision	Micro_Recall	Multilabel accuracy (Jaccard index)
Adam	0.445	0.615	0.681	0.561	0.444
Nadam	0.444	0.612	0.674	0.561	0.441

Table 7
the effect stemming or not the word "□□□□" and derivations, using the model with 50

Stemming "□□□□"	Macro-F1	Micro-F1	Micro Precision	Micro Recall	Multilabel accuracy (Jaccard index)
No	0.445	0.615	0.681	0.561	0.444
Yes	0.441	0.609	0.677	0.553	0.438

Table 8
Results of our proposed model and other state of the art models

Model	Preprocessing	Features	Classification algorithm	Validation Accuracy	Test Accuracy	Micro F1	Macro F1
EMA	Normalization, a manual emoji lexicon + ARLSTEM.	AraVec	SVC	0.488	0.489	0.618	0.461
TW-Star	Emo + Stem + stop	TF-IDF	SVM		0.465	0.597	0.446
UNCC	Tokenization white spaces removal	AraVec + Affective Tweets features	a fully connected neural network		0.446	0.572	0.447
Proposed Model	Normalization + a manual emoji lexicon + ARLSTEM.	AraVec [20]	Bidirectional LSTM	0.58	0.444	0.615	0.445
SVM-Unigrams		Unigrams	SVM		0.38	0.516	0.384
Amrita		Doc2Vec	RF		0.254	0.379	0.25
TeamCEN	unimportant and symbol Space removal	Global Vector (Glove)	RF, SVM		1.8	2.8	1.5

List Of Abbreviations

BiLSTM , ARLSTEM , Aravec; ARLSTEM, NLP, HAR, PD, ArSEL, EMA, BR,TF-IDF,SVM, SGD, RF,RNN, CNN,MSA

Declarations

Ethics approval and consent to participate: Not applicable

Consent for publication: Not applicable

Availability of data and materials: The data that support the findings of this study are publicly available, at https://competitions.codalab.org/competitions/17751#learn_the_details-datasets

Competing interests :The authors declare that they have no competing interests.

Funding: the research has no funding

Authors' contributions: All the authors contributed to the structuring of this paper. All authors read and approved the final manuscript.

1. Liu B. Sentiment analysis: Mining opinions, sentiments, and emotions. Cambridge University Press; 2015.
2. Ekman P. An argument for basic emotions. *Cognition emotion*. 1992;6(3–4):169–200.
3. Plutchik R. *A general psychoevolutionary theory of emotion*, in *Theories of emotion*. Elsevier; 1980. pp. 3–33.
4. Plutchik R. The psychology and biology of emotion. HarperCollins College Publishers; 1994.
5. Mohammad S, et al. *Semeval-2018 task 1: Affect in tweets*. in *Proceedings of the 12th international workshop on semantic evaluation*. 2018.
6. Trad C, et al. *Facial action unit and emotion recognition with head pose variations*. in *International Conference on Advanced Data Mining and Applications*. 2012. Springer.
7. Ruiz-Garcia A, et al. A hybrid deep learning neural approach for emotion recognition from facial expressions for socially assistive robots. *Neural Comput Appl*. 2018;29(7):359–73.
8. Wegrzyn M, et al. Mapping the emotional face. How individual face parts contribute to successful emotion recognition. *PloS one*. 2017;12(5):e0177239.
9. Filippini C, et al., *Facilitating the child–robot interaction by endowing the robot with the capability of understanding the child engagement: The case of mio amico robot*. *International Journal of Social Robotics*, 2020: p. 1–13.
10. Ozcan T, Basturk A. Transfer learning-based convolutional neural networks with heuristic optimization for hand gesture recognition. *Neural Comput Appl*. 2019;31(12):8955–70.
11. Constantine L, et al. *A framework for emotion recognition from human computer interaction in natural setting*. in *22nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD 2016), Workshop on Issues of Sentiment Discovery and Opinion Mining (WISDOM 2016)*. 2016.
12. Hibbeln MT, et al. How is your user feeling? Inferring emotion through human-computer interaction devices. *Mis Quarterly*. 2017;41(1):1–21.
13. Patwardhan AS, Knapp GM. Multimodal affect analysis for product feedback assessment. arXiv preprint. 2017;arXiv:1705.02694.
14. Karyotis C, et al. A fuzzy computational model of emotion for cloud based sentiment analysis. *Inf Sci*. 2018;433:448–63.
15. Giatsoglou M, et al. Sentiment analysis leveraging emotions and word embeddings. *Expert Syst Appl*. 2017;69:214–24.
16. Abdul-Mageed M, Ungar L. *Emonet: Fine-grained emotion detection with gated recurrent neural networks*. in *Proceedings of the 55th annual meeting of the association for computational linguistics (volume 1: Long papers)*. 2017.
17. Badaro G, et al. *Ema at semeval-2018 task 1: Emotion mining for arabic*. in *Proceedings of The 12th International Workshop on Semantic Evaluation*. 2018.
18. Badaro G, et al. Arsel: A large scale arabic sentiment and emotion lexicon. *OSACT*. 2018;3:26.
19. Badaro G, et al. *EmoWordNet: Automatic expansion of emotion lexicon using English WordNet*. in *Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics*. 2018.

20. Soliman AB, Eissa K, El-Beltagy SR. Aravec: A set of arabic word embedding models for use in arabic nlp. *Procedia Computer Science*. 2017;117:256–65.
21. Sonawane B, Sharma P. *Review of automated emotion-based quantification of facial expression in parkinson's patients*. *The Visual Computer*, 2020: p. 1–17.
22. Mohammad SM, *Word affect intensities*. arXiv preprint arXiv:1704.08798, 2017.
23. Bandhakavi A, et al. Lexicon generation for emotion detection from text. *IEEE Intell Syst*. 2017;32(1):102–8.
24. Poria S, et al. *Fuzzy clustering for semi-supervised learning–case study: Construction of an emotion lexicon*. in *Mexican International Conference on Artificial Intelligence*. 2012. Springer.
25. Janz A, Piasecki M. *plWordNet as a basis for large emotive lexicons of Polish..*
26. Abdaoui A, et al. Feel: a french expanded emotion lexicon. *Language Resources Evaluation*. 2017;51(3):833–55.
27. Shoukry A, Rafea A, *Preprocessing Egyptian Dialect Tweets for Sentiment Mining*. 2012.
28. Abainia K, Ouamour S, Sayoud H. A novel robust Arabic light stemmer. *J Exp Theor Artif Intell*. 2017;29(3):557–73.
29. Mulki H, et al. *Tw-star at semeval-2018 task 1: Preprocessing impact on multi-label emotion classification*. in *Proceedings of The 12th International Workshop on Semantic Evaluation*. 2018.
30. Abdullah M, Shaikh S. *Teamuncc at semeval-2018 task 1: Emotion detection in english and arabic tweets using deep learning*. in *Proceedings of the 12th international workshop on semantic evaluation*. 2018.
31. Unnithan NA, et al. *Amrita_student at SemEval-2018 Task 1: Distributed Representation of Social Media Text for Affects in Tweets*. in *Proceedings of The 12th International Workshop on Semantic Evaluation*. 2018.
32. George A, HB BG, Soman K. *Teamcen at semeval-2018 task 1: global vectors representation in emotion detection*. in *Proceedings of the 12th international workshop on semantic evaluation*. 2018.
33. Pennington J, Socher R, Manning CD. *Glove: Global vectors for word representation*. in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*. 2014.

Footnotes

[1] <https://www.nltk.org/>

[2] The python command to download Stopwords list of NLTK package is

```
<< nltk.corpus import stopwords>>, and to assign arabic_stopwords
```

```
<< arabic_stopwords = stopwords.words("arabic")>>
```

[3] <https://competitions.codalab.org/competitions/17751>

Figures

Loading [MathJax]/jax/output/CommonHTML/jax.js

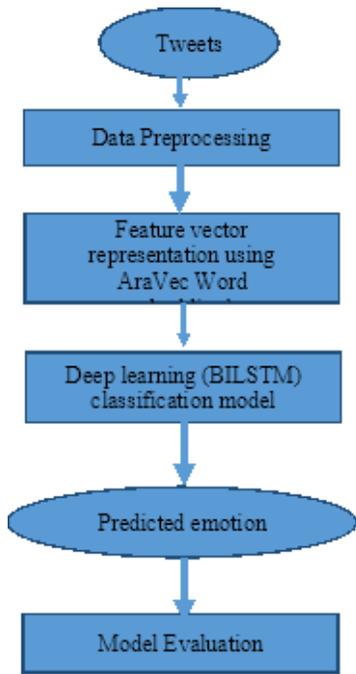


Figure 1 proposed Framework

Figure 1

proposed Framework

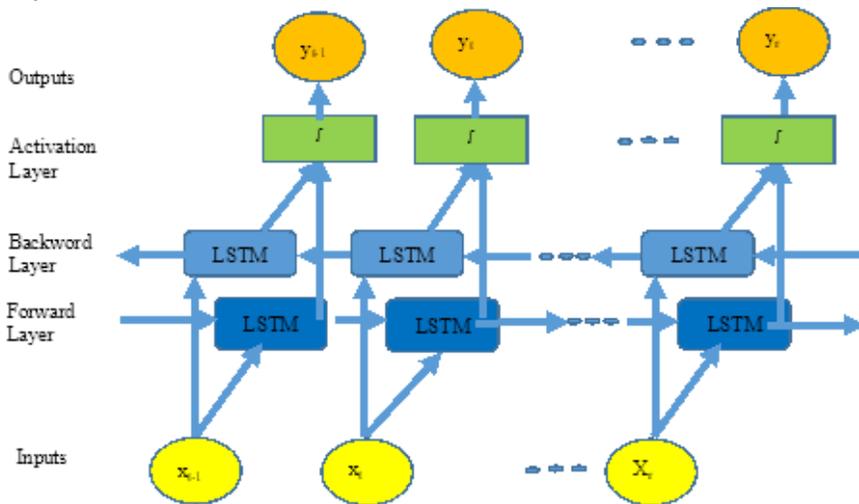


Figure2. The basic structure of the BLSTM network. The LSTM nets at the bottom indicate the forward feature. The above nets are used backward. Both networks connect to an activation layer to produce outputs

Figure 2

The basic structure of the BLSTM network. The LSTM nets at the bottom indicate the forward feature. The above nets are used backward. Both networks connect to an activation layer to produce outputs

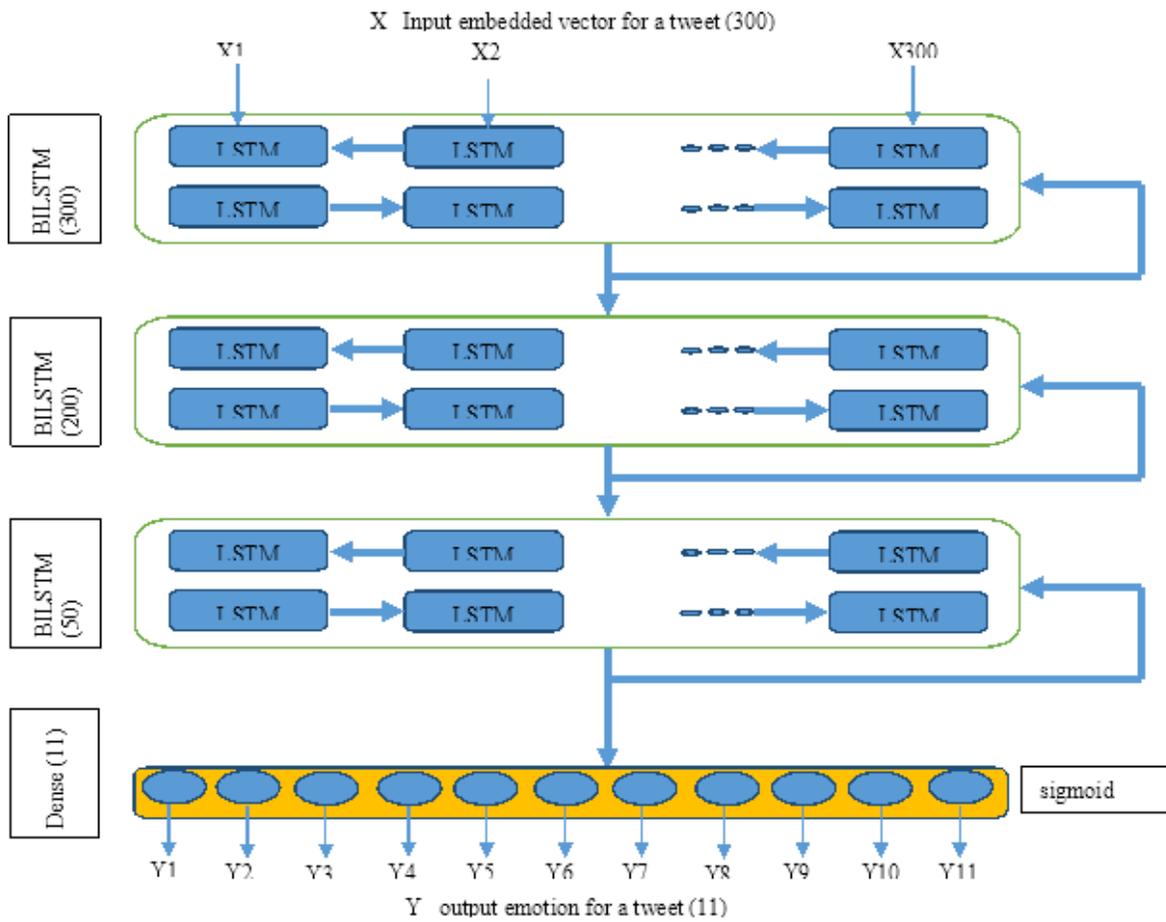


Figure 3 Deep learning (BILSTM) classification model

Figure 3

Deep learning (BILSTM) classification model