A Novel Hybrid Network for Arabic Sentiment Analysis using fine-tuned AraBERT model

Nassera HABBAT*, Houda ANOUN, and Larbi HASSOUNI

RITM Laboratory, CED ENSEM Ecole Superieure de Technologie Hassan II University, Casablanca, Morocco
nassera.habbat@gmail.com, houda.anoun@gmail.com, lhassouni@hotmail.com

Abstract: The pre-trained word embedding models become widely used in Natural Language Processing (NLP), but they disregard the context and sense of the text. We study in this paper, the capacity of pre-trained BERT model (Bidirectional Encoder Representations from Transformers) for the Arabic language to classify Arabic tweets using a hybrid network of two famous models; Bidirectional Long Short Term Memory (BiLSTM) and Gated Recurrent Unit (GRU) inspired by the great achievement of deep learning algorithms. In this context, we fine-tuned the Arabic BERT (AraBERT) parameters and we used it on three merged datasets to impart its knowledge for the Arabic sentiment analysis. For that, we lead the experiments by comparing the AraBERT model in one hand in the word embedding phase, with a statics pre-trained word embeddings method namely AraVec and FastText, and on another hand in the classification phase, we compared the hybrid model with convolutional neural network (CNN), long short-term memory (LSTM), BiLSTM, and GRU, which are prevalently preferred in sentiment analysis. The results demonstrate that the fine-tuned AraBERT model, combined with the hybrid network, achieved peak performance with up to 94% accuracy.

Keywords: Deep learning algorithms; Arabic sentiment analysis; Word embedding; AraBERT, Hybrid network.

1. Introduction

During the COVID-19 pandemic, when people had to stand away, social networks were frequently used by people to express their views, thoughts, feelings and emotions about the pandemic situation. Therefore, it is easy to collect data and identify negative and positive emotions, which is a very popular research topic. In the literature, sentiment analysis mainly focused on the English language while there are few studies on Sentiment Analysis of Arabic texts in relation to other languages, and this is due to the difficult nature of the Arabic language. However, the number of Arabic Internet users has recently increased significantly, and therefore Arabic Sentiment Analysis (ASA) has become an active research area. Indeed, the characteristics of the derivational and inflectional of Arabic complicate its analysis.

The principal idea of the ASA task is to affect predefined classes to Arabic text based on its content. Text representation is a key step that affects ASA performance, and the contextual embedding models are effective for learning universal sentence representations because they can take into account both context and word meaning.

The principal contribution of this work is tackling ASA using deep learning models based on CNN, LSTM, BiLSTM, GRU, and a proposed hybrid network, which combines BiLSTM and GRU taking the benefit of BiLSTM and GRU. Moreover, our study aims to compare AraVec, FastText, and AraBERT word embedding for text representation. We employed three datasets from different resources (around 223,000 tweets) to train the deep learning models. In addition, we used AraBERT for Arabic text preprocessing, and finally, we applied the best model to our collected Moroccan tweets (around 118,000 tweets) to investigate the sentiment of Moroccan people in the COVID-19 period.

We organized our paper as follows; Section 2 gives a brief literature review. The Arabic sentiment analysis progress is described in Section 3. The proposed model is presented in

Received: September 20th, 2021. Accepted: December 26th, 2021
DOI: 10.15676/ijeei.2021.13.4.3

Section 4. Section 5 exposes the results of the experiments. Finally, Section 6 summarizes this study and outlines the future work.

2. Related work

Huge improvement has been made in research on the field of NLP and sentiment analysis for many languages spoken in the world. In particular, research on NLP and sentiment analysis for the English language has already achieved significant results and progressed not only in adapting the latest theories in the areas of lexical analysis and machine learning (ML) but also in applications [1], [2].

In [2], the authors proposed an analysis of Toxic Comments in Wikipedia. They used several feature extraction techniques; such as Term Frequency-Inverse Document Frequency (TF-IDF), popular transformers models; BERT, XLNet tokenizer, and DistilBERT, and they used Glove, word2vec, and Fasttext in the representation of text. Concerning neural network architectures, they implemented four deep learning models: Feed Forward Neural network (FFNN), CNN, LSTM, and GRU. In addition, they implemented a combination of bidirectional GRU, LSTM and convolutional layer and two composed architectures. The 1st one was composed of an embedding layer, bidirectional LSTM layer, and a 1D convolution layer, whereas in the second one; they used GRU layer instead of LSTM layer. As a result of their experiments, they found that the most suitable representation is the Glove pre-trained embedding without standard pre-processing.

In another comparative study [1], the authors found that the model which performed the best result with 84.1% in accuracy is CNN with BERT pre-trained word embedding evaluated on a Twitter dataset, compared to Naïve Bayes, Logistic Regression, SVM, using count vectors and tf-idf as machine learning algorithms, and to LSTM, Stacked LSTM with 1D convolution with BERT as deep learning algorithms.

Whereas, in [3] the authors performed an analysis about COVID-19 on 596,784 tweets from India and the rest of the world from 20 January 2020 to 25 April 2020. They found that sentiment analysis using BERT model gives approximately 94% term of the validation accuracy. However, in [4]; the BiLSTM model with an attention mechanism attained the best result on the classification task with 72.09% in terms of F1 score compared to 1D-CNN, and a hybrid 1D-CNN + BiLSTM model with Fasttext and BERT, this analysis was done on 10,742 Albanian Facebook comments concerning the COVID-19 pandemic which were manually classified.

A combination of different deep learning models is used in some studies to increase the accuracy of text classification. In [5]; the authors proposed a hybrid model (Bi-LSTM+CNN +additional attention mechanism) that makes use of the advantages of BiLSTM and CNN. After evaluating their approach in IMDB dataset (movies reviews), they concluded that the proposed model outperforms LSTM, CNN, and individual multi-layer perception (MLP) with the highest accuracy of 91,41%. The accuracy of 82,14% was obtained using the same combination (BiLSTM + CNN) in [6] applying to Turkish tweets as a dataset. Another combination was evaluated in [7] concerning (LSTM + GRU) for problems based on the regression (e.g. prediction of stock price) and their model achieved 0.00098 in mean squared error (MSE).

Research work on other languages, such as Arabic, concerning sentiment analysis is making good progress. In [8] the authors present a survey that describes in detail the past and recent progress of multilingual sentiment analysis in formal and informal languages used on online social platforms, using different deep learning models. For ASA, another comprehensive overview of research works was conducted by [9].

In [10], the authors proposed a model composed of a CNN and stacked independent LSTM to analyze Arabic SemEval-2016; a manual-labeled dataset for the Hotels field, and their model achieved 58,08% in F1 score. In the same context, [11] proposed an Arabic deep learning model to perform a sentiment analysis on the largest corpus of 63,000 books reviews. This model consists of one layer CNN, two layers LSTM, SVM classifier, and the FastText skip-gram technique is used for word embedding. As result, their model achieved competitive performance in terms of recall, precision, and F1-score with 92.14%, 89.10%, and 90.44% respectively.

A recent trend in language modeling is that of contextual integration models such as ELMo, ULMFiT, and BERT. BERT [12] proved to be very effective at language understanding including the Arabic language; in [13]; the authors applied pre- trained AraBERT to various natural language comprehension tasks such as Sentiment Analysis, Question Answering, and Named Entity Recognition and compared it with multilingual BERT. AraBERT attained the best performance in different Arabic NLP tasks.

3. Arabic Sentiment Analysis progress

Text classification based on deep learning using NLP is a popular subject nowadays. This paper proposes some models to classify emotions from Moroccan tweets.

Data preprocessing is a necessary step after dataset collection. Then we presented deep learning algorithms namely CNN, LSTM, BiLSTM, GRU, and a proposed hybrid network, after that, we compared the classifiers using different word embedding models; AraVec, FastText, and AraBERT. The proposed model, which consists of a hybrid network using AraBERT as word embedding, will be presented in the next section.

The main purpose of this process is to enhance the accuracy of emotion detection. We train all the models with various word embeddings mentioned in Figure 1. Finally, the best models are obtained by comparing the results in terms of evaluation metrics. The details of these models are explained in this section.

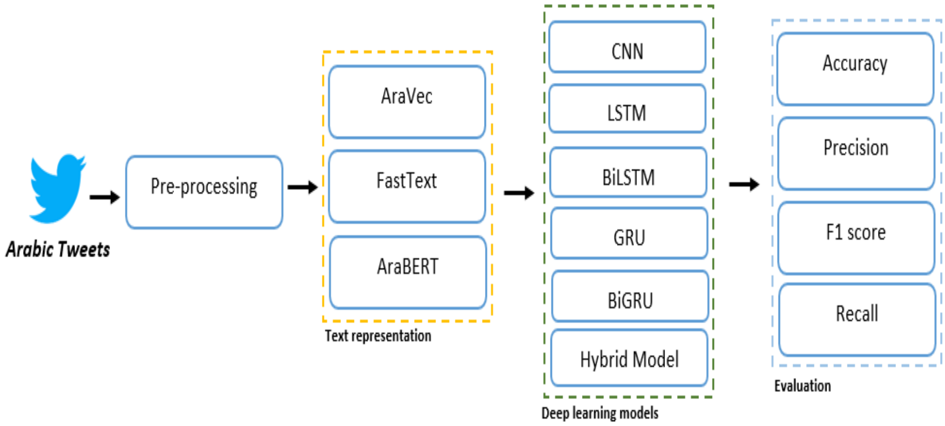

Figure 1. Proposed methodology

A. Text pre-processing:

This step is one of the most important processes for obtaining a clean version of a given dataset, in our study; we applied the following preprocessing steps:

- Normalization. This process consists of transforming a text into a standard form. English and Arabic punctuation, non-letters, hyperlinks, and symbols are removed, and diacritics such as (Damma, Fatha, Kasra, Tanwin Damm, Tanwin Fath, and Tatwil) as shown in Table 1. We used two tools to pre-process the tweets; the Natural Language Toolkit (NLTK) [14], and regular expressions using python.

Table 1. The normalization process applied to three Arabic expressions.

Word	لااااااااا	الأب	نعم ... صحيح
Normalized form	لا	الاب	نعم صحيح

- Data transformation. This step consists of converting data to correspond to the fine-tuned AraBERT model. We used AraBERTv1 [15] to decompose the sentences into sections (tokenization) because AraBERTv1 tokenizer is relevant for Arabic as it based on Farasa Segmenter [16]. Then, the special [CLS] symbol is added at the start of every text and the token [SEP] to separate sentences and added in the end, after that, we get BERT's format, and finally, based on the vocabulary of pre-trained AraBERT, each token is mapped to an index.

B. Text representation:

Pre-trained vector models allow improving the global accuracy, especially with the fast application of neural networks in the NLP field. In this field, word embedding has been widely used. It tends to capture the semantic and syntactic relationship of the text and map the vocabulary words to the real vector.

In this part, we present the static Word Embeddings used in our study, especially AraVec and Fastext:

- **AraVec** [17]: It is an open-source pre-trained distributed word embedding for Arabic NLP tasks. AraVec dispose of various text representation models made on two Arabic content fields; Arabic Wikipedia and tweets items, resulting in more than one billion tokens. We used the AraVec word embedding pre-trained on Twitter-CBOW, 300 dimensions with reference to Word2vec (Mikolov et al., 2013.) which is one of the most commonly used methods for learning word embeddings based on shallow neural network.
- **FastText** [19]: it is a library developed by the team of Facebook research to compute the representation of the word. it is a pre-trained word vector trained in a wide range of languages (157 languages). We used in this study, a FastText 300d skip-gram trained on Wikipedia for the Arabic language, By default, the word vector takes into account three to six-character n-grams which are suited for Arabic text, as every word in the Arabic language has a root of three letters.

C. Deep learning models:

Among the most powerful machine learning algorithms we find Neural Networks. They have been very successful in Artificial Intelligence (AI), namely in the computer vision domain (e.g. voice recognition and sound processing). In this part, we will discuss how to use neural networks to process text data in more detail.

C.1. Convolutional Neural Network (CNN):

CNN [20] is a type of deep neural networks, it holds a neurons layer used for the convolution process. Neurons reply to the input of the activations of the vicinity neurons depending on the specific convolution kernel size (kernel also known as a filter). Convolution moves the convolution kernel to a complete values set.

To minimize the number of outputs, prevent the network from overfitting in convolutional networks, reduce the computational complexity, and also remove excessing data, pooling layers are designed. The sampling layers are generally used directly before the convolution layers, as the double data is produced when the convolution kernels are changed by the single inputs.

In our study, as shown in Figure 2a we construct a 1D convolutional network, this model contains a first layer, which is the convolutional one with different parameters (as the filter=512). These feature detectors are transmitted through the matrix of word embeddings using the ReLU function to indicate determined features. The convolution outputs are pooled utilizing the Max and Global pooling layers. Then, we used dropout to stop overfitting and raise its score to 0.8. Finally, in the output layer, we employed the sigmoid activation function to classify texts into predefined categories.

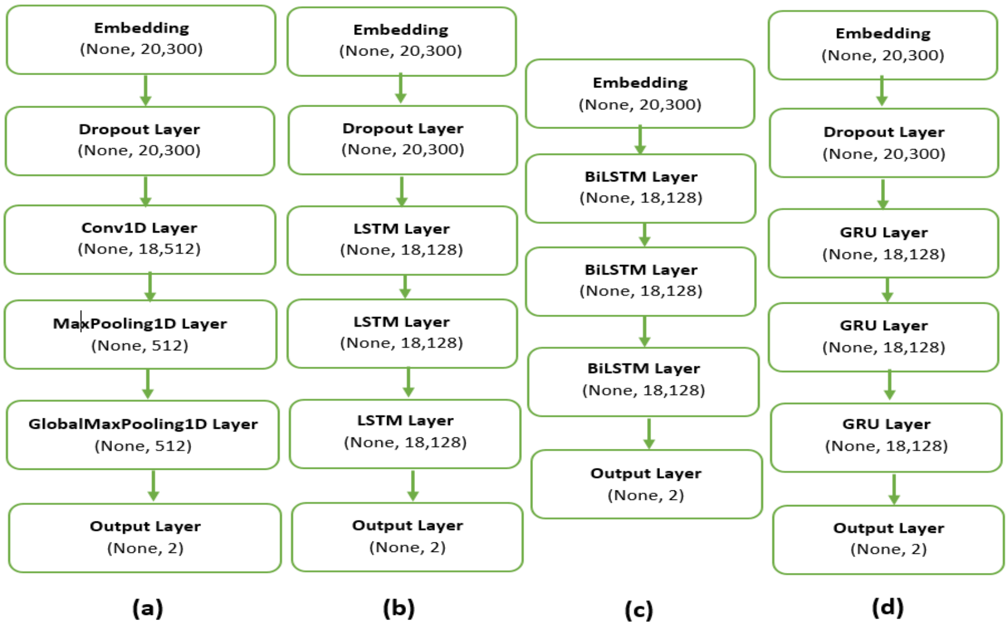

Figure 2. Used deep learning architectures. (a) CNN. (b) LSTM (c) BiLSTM (d) GRU.

C.2. Long-Short Term Memory (LSTM):

LSTM [21] is a recurrent neural network (RNN), Its structure is more complex, which allows it for dealing with the shifty gradient problem. This model is designed to work on sequences to maintain long-term information and context.

Unlike other types of neural networks which are composed of interconnected neurons, LSTM consists of memory blocks linked in layers. This block contains a gateway, which is used to manage the information flow as well as the output and state of the block. The gateway can know which input in the sequence is important and which data must be retained.

As shown in Figure 2b, we built an LSTM with 3 sequential layers and one layer producing the final result containing 128 units. We give as input a text representation matrix containing the embedding vectors generated using different word embedding described in the previous part. The dropout score is set at 0.6. Then, the result vector of the previous layer is transmitted to a fully connected network. After all, the sigmoid activation function is employed in the output layer to get the appropriate class.

C.3. Bidirectional LSTM:

BiLSTM [22] is a special LSTM network. The idea of BiLSTM is to aggregate the previous and future input information of a given time step into the LSTM model.

BiLSTM uses a bidirectional LSTM layer to find patterns that can be found by browsing the input data history forward and backward. The forward sequence as input is processed in the first layer, and then the second layer is used to treat the backward sequence. Since the output layer can access the gone and future context of each point in the sequence.

Although the LSTM is a feed-forward network and reads the sequences from left to right, the Bi-LSTM unifies two layers on the same output in opposite directions (forward and backward).

As shown in Figure 2c, we constructed a BiLSTM implementation, which includes three bidirectional sequential layers and a final generation layer containing 128 units. A representation matrix containing embedding vectors is given, and the result vectors of the previous layer are transferred to the fully connected network. Finally, in the output layer the appropriate classes are obtained using the sigmoid activation function.

C.4. Gated Recurrent Unit:

GRU [23] is an RNN network architecture amply used, and because the design of both LSTM and GRU is the same, GRU can be considered a variant of the LSTM. Using update and reset gates, GRU solve the vanishing gradient problem. And the update gate allows the model to define the amount of information before (from the precedent time point).

As shown in Figure 2d, we constructed a GRU with three consecutive layers, one of which generated the final result with 128 units. We provide a representation matrix that contains the embedding vector. The dropout score is set to 0,6. Then the result vector of the upper layer is transmitted to the fully connected network. The sigmoid activation function is employed in the output layer to obtain the appropriate class.

4. Proposed method

In this part, we will define the proposed method composed of AraBERT contextual word embedding model as input of hybrid network, which is a combination of BiLSTM and GRU.

A. Contextualized Word Embedding

We analyze in this part, the influence of context word embedding on sentiment analysis tasks. In particular, we used the AraBERT model [13], it is an Arabic pre-training BERT transformer model, which is a deep, unsupervised bidirectional language representation that can create word embeddings to represent the semantics of words in their context. It is pre-trained in datasets from Arabic news websites for articles; about 1 billion tokens in 3.5 million articles from Open Source International Arabic News (OSLAN) Corpus, and 1.5 billion words in 5 million articles from 10 major sources of news from eight countries.

We present in Figure 3, the best-as-a-service technology, which pulls out the activation of one or further layers without fine-tuning the parameters of AraBERT. It calculates the average pool of the second to the last hidden layer of all tokens. The output representation becomes the input of different used classification models that we will present in the next part.

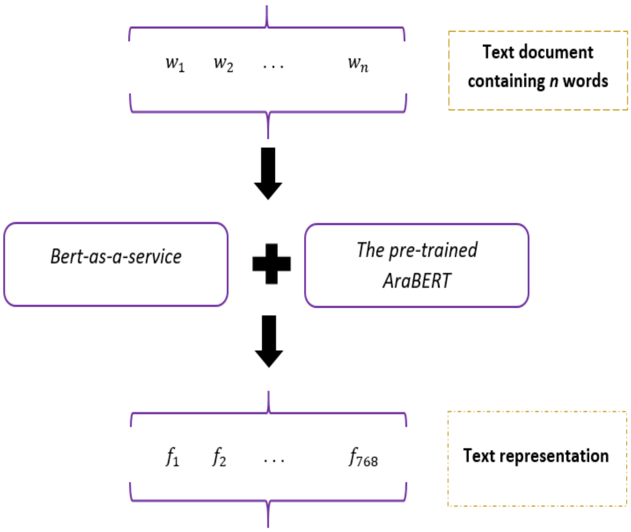

Figure 3. Feature extraction of the pre-trained AraBERT.

We used in our study, the pretrained 'bert-base-arabertv02' Arabic embedding with 768 hidden dimensions, 12 attention heads and encoder layers, and 110 M parameters [24].

B. Hybrid network

In Figure 4 we expose our proposed model which is a combination of BiLSTM and GRU. Firstly, we transfer the input vector of the word embedding to the BiLSTM with a hidden layer. Then we move the output of Bi-LSTM layer to the GRU entrance. We had the output of the GRU at the second layer, which becomes the input of a dense network after passing by the dropout layer, and finally, we obtained the classes using the sigmoid activation function.

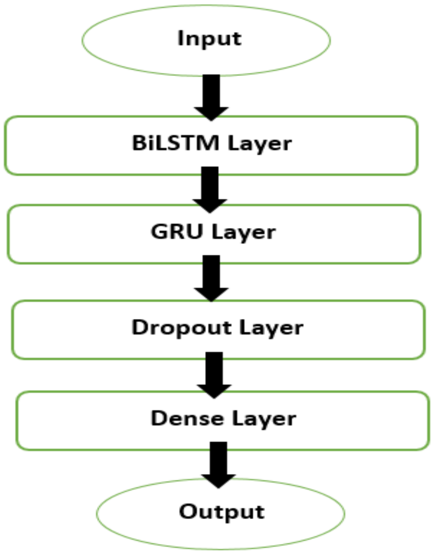

Figure 4. Hybrid model Architecture.

5. Experiments and results

This part describes the dataset used for experiments and presents the experimental configurations and parameters of deep learning algorithms, results, and related discussion.

A. Dataset description

In our analysis, we used two datasets published by a Kaggle competition [25] which contains around 193,000 Arabic tweets, in addition, we used around 49,000 tweets manually labeled into Positive and Negative as Moroccan dataset. Those datasets were merged and split into 80% Training and 20% Testing. After training and testing our models we applied the best model to a collected dataset, which consists of 118,154 tweets published by Moroccan users during the COVID-19 pandemic from 29 May 2020 to 23 May 2021 and stored into MongoDB database. The details of our used datasets are described in the following Table 2 :

Table 2. Constructed training and testing set statistics.

Datasets	Positive tweets	Negative tweets	Total
Kaggle dataset 1	22 761	22 514	45 275
Kaggle dataset 2	76 094	72 234	148 328
Moroccan Dataset	24 932	24 932	49 864
Total Train and Test dataset	123 787	119 680	223 467

B. Deep learning models settings:

The experiments are conducted on the TensorFlow (Version 2.4.1) and Keras (Version 2.4.3) framework running on Python (Version 3.8), and Adam (Adaptive Moment Estimation) optimizer was used to reduce the model error rate in the prediction, with 1e-3 in learning rate and 64 in batch size. We fine-tuned our model on 15 epochs on the data.

C. Evaluation metrics:

To get performance conclusions from different models, we employed the standard measurements used in the classification task, such as precision, accuracy, F-measure, and recall.

The calculation of these parameters is based on a False Statement. For instance, a sentence is declared positive even if the sentence is negative or a sentence is declared neutral even though the sentence is negative or positive. To uncover the False Statement, we must make a performance evaluation.

There is four classifications of performance evaluation:

- T_Pos: True Positive
- F_Pos: False Positive
- T_Neg: True Negative
- F_Neg: False Negative

C.1. Accuracy:

The accuracy parameter is set by the number of observations correctly predicted divided by the total observations.

$$\text{Accuracy} = \frac{\text{T\_Pos} + \text{T\_Neg}}{\text{T\_Pos} + \text{T\_Neg} + \text{F\_Pos} + \text{F\_Neg}}$$

C.2. Precision:

The precision parameter is a positive prognostic value. It is the percentage of total relevant results correctly predicted by the result of the algorithm.

$$Precision = \frac{T\_Pos}{T\_Pos + \text{F\_Pos}}$$

C.3. Recall:

The recall parameter is the average probability of complete data recovery.

$$Recall = \frac{T\_Pos}{T\_Pos + F\_Neg}$$

C.4. *F-measure:*

It is defined as the weighted average of the recall and precision of the test.

$$F_{Measure} = \frac{2 \times Recall \times Precision}{Recall + Precision}$$

D. Discussion

We expose in this part, the results of our experiments which consist of comparing the effect of various word embedding models on the classification of Arabic tweets, we compared those models using accuracy, precision, recall, and f-measure.

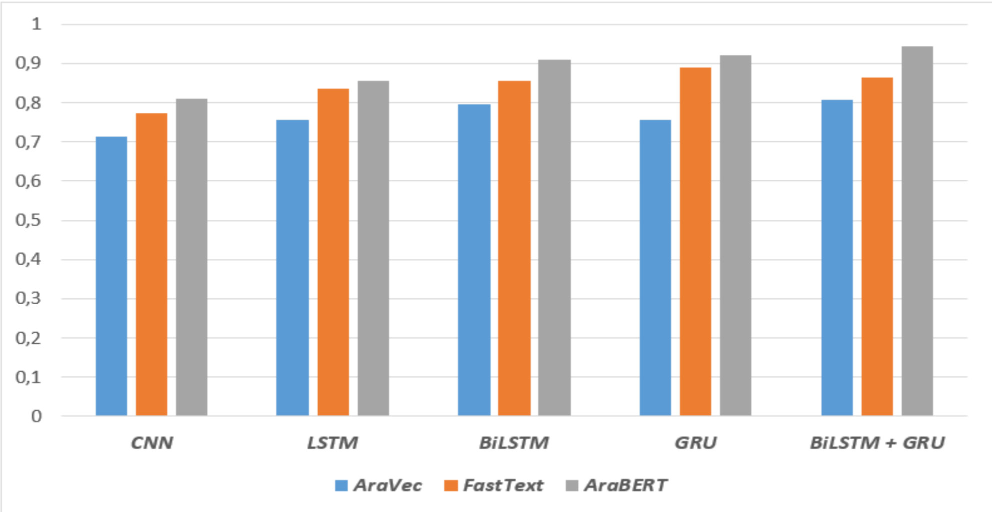

Figure 5. Accuracy of deep learning models using different word embedding.

As shown in Figure 5, The proposed combination (BiLSTM + GRU) outperforms other models using different word embeddings, and as presented in Table 3, the results suggest that the proposed model in combination with AraBERT as contextualized word embeddings are a good choice detecting the sentiment of Arabic tweets in the analyzed datasets with precision, recall, f-measure and accuracy percentages of 92.64%, 89.41 %, 85.35%, and 94.34 %, respectively, and cross-validation of BiLSTM +GRU+AraBERT is exposed in Table 4 where we made a 15-time cross-validation of the better efficient model on the training data.

The BiLSM and GRU model seems to be the best because BiLSTM has better performance and a faster learning rate, also the calculation of GRU is simpler than the calculation of LSTM, and then it executes faster calculations in addition to reducing memory.

Table 3. Performance evaluation of deep learning algorithms with various embeddings models.

Deep learning algorithms	*Word Embeddings*	*Accuracy*	*Precision*	*Recall*	*F1-measure*
CNN	*AraVec*	0,7131	0,7364	0,4064	0,2648
	FastText	0,7720	0,7951	0,4820	0,3874
	AraBERT	0,8094	0,8285	0,6157	0,5042
LSTM	*AraVec*	0,7546	0,7480	0,3065	0,2453
	FastText	0,8346	0,8573	0,3751	0,4987
	AraBERT	0,8562	0,8756	0,4621	0,5312
BiLSTM	*AraVec*	0,7954	0,8061	0,6129	0,3468
	FastText	0,8565	0,8236	0,7689	0,6423
	AraBERT	0,9094	0,9020	0,5720	0,8954
GRU	*AraVec*	0,7562	0,7612	0,5612	0,1756
	FastText	0,8908	0,9002	0,4060	0,5041
	AraBERT	0,9216	0,9187	0,8012	0,8274
BiLSTM + GRU	*AraVec*	0,8062	0,8237	0,7237	0,2052
	FastText	0,8645	0,8751	0,3412	0,5085
	AraBERT	**0,9434**	0,9264	0,8941	0,8535

Table 4. Cross-validation of the BiLSTM + GRU + AraBERT model.

Epochs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
Accuracy	0,8488	0,8599	0,8776	0,8897	0,8989	0,9398	0,9598	0,9699
Loss	0,1275	0,1048	0,1026	0,0931	0,0691	0,0697	0,0522	0,0475
Epochs	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**Avg**
Accuracy	0,9736	0,9837	0,9846	0,9853	0,9848	0,9858	0,9897	0,9434
Loss	0,0417	0,0411	0,0393	0,0367	0,0365	0,0345	0,0338	0,062

After getting the best model for detecting Arabic tweets, we applied it to our collected Moroccan tweets to investigate the sentiment of Moroccan people in the COVID-19 period, as shown in Figure 6, the sentiment analysis results reveal that Twitter users in Morocco posted more positive tweets.

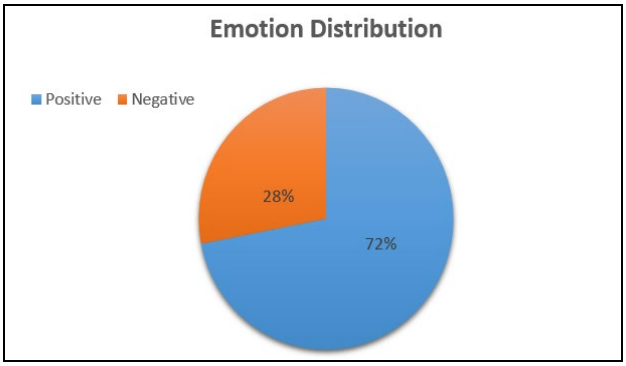

Figure 6. Sentiment analysis of Moroccan tweets

6. Conclusion and Future work

Emotion analysis is the analysis of social data to determine the audience's inclinations.
For Arabic, sentiment analysis is dependent on the terminology of the input phrase and independent of the in-depth study of semantic and syntactic rules, which is a challenge.

Therefore, in this work, we tried to compare and assess various sentiment analysis models on Arabic tweets. We experimentally assessed the performance of four deep learning models namely CNN, LSTM, BiLSTM, GRU, and a hybrid model (BiLSTM + GRU) with three text representation techniques (i.e. AraVec, FastText, AraBERT).

Among these models, the proposed model (BiLSTM + GRU) with the AraBERT model has obtained the best accuracy with 0,9434. The analysis of the results of deep learning models clearly shows that for our dataset, the performance of the hybrid network is better than other models for different word embedding, and their accuracy is more than other models' accuracy.

In future work, we intend to combine other models to increase the accuracy using different word embeddings (AraBERT, ELMO, …).

7. References

[1]. S. K. Lora, « A Comparative Study to Detect Emotions from Tweets Analyzing Machine Learning and Deep Learning Techniques », *Int. J. Appl. Inf. Syst.*, vol. 12, p. 8, 2020.

[2]. V. Maslej-Krešňáková, M. Sarnovský, P. Butka, et K. Machová, « Comparison of Deep Learning Models and Various Text Pre-Processing Techniques for the Toxic Comments Classification », *Appl. Sci.*, vol. 10, nº 23, p. 8631, déc. 2020, doi: 10.3390/app10238631.

[3]. M. Singh, A. K. Jakhar, et S. Pandey, « Sentiment analysis on the impact of coronavirus in social life using the BERT model », *Soc. Netw. Anal. Min.*, vol. 11, nº 1, p. 33, déc. 2021, doi: 10.1007/s13278-021-00737-z.

[4]. Z. Kastrati, L. Ahmedi, A. Kurti, F. Kadriu, D. Murtezaj, et F. Gashi, « A Deep Learning Sentiment Analyser for Social Media Comments in Low-Resource Languages », *Electronics*, vol. 10, nº 10, p. 1133, mai 2021, doi: 10.3390/electronics10101133.

[5]. B. Jang, M. Kim, G. Harerimana, S. Kang, et J. W. Kim, « Bi-LSTM Model to Increase Accuracy in Text Classification: Combining Word2vec CNN and Attention Mechanism », *Appl. Sci.*, vol. 10, nº 17, p. 5841, août 2020, doi: 10.3390/app10175841.

[6]. M. U. Salur et I. Aydin, « A Novel Hybrid Deep Learning Model for Sentiment Classification », *IEEE Access*, vol. 8, p. 58080-58093, 2020, doi: 10.1109/ACCESS.2020.2982538.

[7]. M. A. Hossain, R. Karim, R. Thulasiram, N. D. B. Bruce, et Y. Wang, « Hybrid Deep Learning Model for Stock Price Prediction », in *2018 IEEE Symposium Series on Computational Intelligence (SSCI)*, Bangalore, India, nov. 2018, p. 1837-1844. doi: 10.1109/SSCI.2018.8628641.

[8]. M. M. Agüero-Torales, J. I. Abreu Salas, et A. G. López-Herrera, « Deep learning and multilingual sentiment analysis on social media data: An overview », *Appl. Soft Comput.*, vol. 107, p. 107373, août 2021, doi: 10.1016/j.asoc.2021.107373.

[9]. A. Alharbi, M. Taileb, et M. Kalkatawi, « Deep learning in Arabic sentiment analysis: An overview », *J. Inf. Sci.*, vol. 47, nº 1, p. 129-140, févr. 2021, doi: 10.1177/0165551519865488.

[10]. S. Al-Dabet, S. Tedmori, et M. AL-Smadi, « Enhancing Arabic aspect-based sentiment analysis using deep learning models », *Comput. Speech Lang.*, vol. 69, p. 101224, sept. 2021, doi: 10.1016/j.csl.2021.101224.

[11]. A. H. Ombabi, W. Ouarda, et A. M. Alimi, « Deep learning CNN–LSTM framework for Arabic sentiment analysis using textual information shared in social networks », *Soc. Netw. Anal. Min.*, vol. 10, nº 1, p. 53, déc. 2020, doi: 10.1007/s13278-020-00668-1.

[12]. J. Devlin, M.-W. Chang, K. Lee, et K. Toutanova, « BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding », *ArXiv181004805 Cs*, mai 2019, Consulté le: janv. 14, 2021. [En ligne]. Disponible sur: http://arxiv.org/abs/1810.04805

[13]. W. Antoun, F. Baly, et H. Hajj, « AraBERT: Transformer-based Model for Arabic Language Understanding », p. 7.

[14]. S. Bird, E. Klein, et E. Loper, *Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit*. O'Reilly Media, Inc., 2009.

[15]. « arabert/arabert at master · aub-mind/arabert », *GitHub*. https://github.com/aub-mind/arabert (consulté le juill. 30, 2021).

[16]. MagedSaeed, *farasapy: A Python Wrapper for the well Farasa toolkit*. Consulté le: févr. 11, 2021. [En ligne]. Disponible sur: https://github.com/MagedSaeed/farasapy

[17]. A. B. Soliman, K. Eissa, et S. R. El-Beltagy, « AraVec: A set of Arabic Word Embedding Models for use in Arabic NLP », *Procedia Comput. Sci.*, vol. 117, p. 256-265, 2017, doi: 10.1016/j.procs.2017.10.117.

[18]. T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, et J. Dean, « Distributed Representations of Words and Phrases and their Compositionality », p. 9.

[19]. A. Joulin, E. Grave, P. Bojanowski, et T. Mikolov, « Bag of Tricks for Efficient Text Classification », *ArXiv160701759 Cs*, août 2016, Consulté le: juin 22, 2021. [En ligne]. Disponible sur: http://arxiv.org/abs/1607.01759

[20]. A. Zhang, Z. C. Lipton, M. Li, et A. J. Smola, « Dive into Deep Learning », p. 1021.

[21]. S. Hochreiter et J. Schmidhuber, « Long Short-Term Memory », *Neural Comput.*, vol. 9, nº 8, p. 1735-1780, nov. 1997, doi: 10.1162/neco.1997.9.8.1735.

[22]. M. Schuster et K. K. Paliwal, « Bidirectional recurrent neural networks », *IEEE Trans. Signal Process.*, vol. 45, nº 11, p. 2673-2681, nov. 1997, doi: 10.1109/78.650093.

[23]. K. Cho *et al.*, « Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation », *ArXiv14061078 Cs Stat*, sept. 2014, Consulté le: août 27, 2021. [En ligne]. Disponible sur: http://arxiv.org/abs/1406.1078

[24]. *google-research/bert*. Google Research, 2021. Consulté le: avr. 20, 2021. [En ligne]. Disponible sur: https://github.com/google-research/bert
[25]. « Kaggle: Your Home for Data Science ». https://www.kaggle.com/ (consulté le juill. 02, 2021).

Nassera HABBAT is a doctoral student at Hassan II University in Morocco. In 2015, she obtained from Caddi Ayyad University a master's degree in Information System Engineering and a bachelor's degree in Technology and Web Programming in 2013. Her research areas are Big data and Artificial Intelligence (AI).

Houda ANOUN obtained in 2007 from Bordeaux I University her Ph.D. in computational linguistics, and she received in 2003, her engineering degree in Software Engineering from ENSEIRB Bordeaux. Actually, she occupies the post of professor in the Department of Computer Science at Superior School of Technology in Hassan II University since 2009. She is currently interested in the field of IA, in particular big data, machine learning, and deep learning.

Larbi Hassouni prepared his Ph.D. diploma at the University of Aix Marseille III in France in 1987, and in 1983, he got his engineer degree from the “Marseille Central School”. As a part of his research work, a list unification software is developed by using the LeLisp language of INRIA to help realize the inference engine of the digital circuit diagnosis expert system. In addition, he developed a digital circuit behavior symbol simulator using C language, FRL language, and LeLisp language to help develop a "formal proof" tool to correct the design of material generated by Hardware Design Language (HDL). At present, his research mainly focuses on Data Science.