IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received January 24, 2020, accepted February 18, 2020, date of publication February 24, 2020, date of current version March 3, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2975906

Hybrid Feature Model for Emotion Recognition in Arabic Text

NOURAH ALSWAIDAN AND MOHAMED EL BACHIR MENAI
Department of Computer Science, College of Computer and Information Sciences, King Saud University, Riyadh 11543, Saudi Arabia
Corresponding author: Nourah Alswaidan (nourah_swaidan@yahoo.com)

This work was supported by the Research Center of the College of Computer and Information Sciences, King Saud University.

ABSTRACT In recent years, research into developing state-of-the-art models for Arabic natural language processing tasks has gained momentum. These models must address the added difficulties related to the nature and structure of the Arabic language. In this paper, we propose three models, a human-engineered feature-based (HEF) model, a deep feature-based (DF) model, and a hybrid of both models (HEF+DF) for emotion recognition in Arabic text. We evaluated the performance of the proposed models on the SemEval-2018, IAEDS, and AETD datasets by comparing the performances of those models on each emotion label. We also compared the model performances with those of other state-of-the-art models. The results show that the HEF+DF model outperformed the DF and HEF models on all datasets. The DF model performed better than the HEF model on the SemEval-2018 and AETD datasets, while the HEF model performed better than the DF model on the IAEDS dataset. The HEF+DF model outperformed the state-of-the-art models in terms of accuracy, weighted-average precision, weighted-average recall, and weighted-average F-score on the AETD dataset and in terms of accuracy, macro-averaged precision, macro-averaged recall, and macro-averaged F-score on the IAEDS dataset. It also achieved the best macro-averaged F-score and the second-best Jaccard accuracy and micro-averaged F-score on the SemEval-2018 dataset.

INDEX TERMS Arabic natural language processing, deep learning, emotion recognition, small dataset.

I. INTRODUCTION

Because we rely on computers to perform our daily tasks, the need for improved human-computer interactions has increased. Text is the main medium of human-computer interactions in various forms: text messages, emails, product reviews, web blogs, and other social media platforms, including Facebook, Twitter, and YouTube. Automating emotion recognition can benefit the field of human-computer interaction as well as other fields, including virtual reality, e-learning, psychology, business, data mining, information filtering systems, and robotics. The computer's lack of common-sense knowledge makes it difficult for computers to understand emotion; thus, emotion recognition from text is both difficult and also an important natural language processing task (NLPs).

Emotion recognition from text refers to the task of automatically assigning emotion to text selected from a set of predefined emotion labels. There are few published studies on emotion recognition in Arabic text. In general, NLP in Arabic is not as advanced as NLP in English. Arabic is a Semitic language spoken by more than 400 million people. There are three main types of Arabic: Classical Arabic (CA), which is used in the Quran, modern standard Arabic (MSA), which is used in formal conversations and writing, and the Arabic dialect (AD), which is used in daily life communication and social media. Arabic is written from right to left. The number of Arabic alphabets, not counting the hamza, is 28. No capitalization exists in Arabic, but the letters change shapes according to their positions in words. To develop a model for Arabic, one must have insight into the structure and syntax of the Arabic language.

Motivated by the objective of boosting the research on Arabic NLP, this paper proposes three models, a human-engineered feature-based (HEF) model, a deep feature-based (DF) model, and a hybrid model (HEF+DF) for emotion recognition in Arabic text. For the HEF model, we selected features that represent different aspects of the text. The feature set includes stylistic, lexical, syntactic, and semantic features. For the DF model, we built the embedding

The associate editor coordinating the review of this manuscript and approving it for publication was Imran Sarwar Bajwa.

layer using four different pre-trained word embedding models. We overcame the out-of-vocabulary (OOV) word problem by calculating the characters' embeddings from these pre-trained word embedding models. The DF model consists of stacked deep neural networks in which the embedding layer is reinserted multiple times to slow down the learning process. The performance of the proposed models was tested on three datasets, the SemEval-2018 dataset, the Iraqi Arabic emotion dataset (IAEDS), and the Arabic emotions Twitter dataset (AETD). The results show that the HEF+DF model outperformed the HEF and DF models on all datasets. Moreover, the HEF+DF model outperformed other state-of-the-art models on the IAEDS dataset in terms of accuracy, macro-averaged precision (P^{macro}), macro-averaged recall (R^{macro}), and macro-averaged F-score (F^{macro}). It also outperformed the state-of-the-art models on the AETD dataset in terms of accuracy, weighted-average precision ($P^{weighted}$), weighted-average recall ($R^{weighted}$), and weighted-average F-score ($F^{weighted}$). Finally, it achieved the best F^{macro} and the second-best Jaccard accuracy and micro-averaged F-score (F^{micro}) on the SemEval-2018 dataset.

The remainder of this paper is organized as follows. Section II presents related works. Section III describes the proposed models for emotion recognition in Arabic text. Section IV presents the experiments, reports the performance results, and provides a discussion. Finally, we conclude this work in Section V and outline some future research directions.

II. RELATED WORK

The research work for emotion recognition in Arabic is not as advanced as is emotion recognition research work for English or Chinese. The limited resources in Arabic are the main contributors to this issue. Mohammad *et al.* [1] organized the SemEval-2018 Task 1: affect in Tweets, which included five subtasks. The fifth subtask was multi-label emotion recognition in tweets. They created labeled training, development, and testing datasets in three languages: Arabic, English, and Spanish. The annotations were performed by presenting one tweet at a time to the annotators and asking them which of eleven emotions best described the emotional state of the tweeter. More information on the dataset and the distribution of instances between the emotion labels is provided in Section IV-A Datasets. The number of participants in the SemEval-2018 competition for emotion recognition in Arabic compared to the number of English participants was low. Of the eleven participants, only five achieved results higher than the baseline, and of those five, only Badaro *et al.* [2], Mulki *et al.* [3], and Abdullah and Shaikh [4] submitted a paper describing their systems.

Badaro *et al.* [2] proposed a learning-based model for multi-label emotion recognition and tested several features, including n-grams, affect lexicons, sentiment lexicon, and word embeddings from AraVec [5] and FastText [6]. AraVec embeddings outperformed the other features. The authors also tested several learning models, including a support vector classifier (SVC) with both L1 and L2 penalties, ridge classification (RC), random forests (RF), and an ensemble of the three. Linear SVC with L1 outperformed the other learning models. Mulki *et al.* [3] formulated multi-label emotion recognition as a binary classification problem and tested different preprocessing steps. The preprocessing pipeline used in their best results replaced emoji with emotion tags and performed stemming and stop-word removal. They used term frequency-inverse document frequency (TF-IDF) to generate the features and performed classification using a one-vs-all support vector machine (SVM) classifier with a linear kernel. Abdullah and Shaikh [4] also formulated multi-label emotion recognition as a binary classification problem and used pre-trained AraVec word embeddings for word representation. The embeddings were fed into four dense neural networks (DNNs); the output of the fourth DNN was normalized to either one or zero based on a threshold of 0.5.

Samy *et al.* [7] proposed a context-aware gated recurrent unit (C-GRU). The preprocessing steps included removing links, hashtag symbols, user mentions, diacritics, and elongations. Then, they normalized characters such as the ''hamza'', ''alf'', ''haa'', and ''yaa''. The input to the C-GRU model was a set of sentences and their corresponding topic representations. For word representation, they used 300-dimensional pre-trained word embeddings from AraVec. A gated recurrent unit (GRU) model was pre-trained to detect topics on the SemEval-2017 [8] dataset. Utilizing a transfer learning approach for topic detection overcomes the challenges of learning from a small training dataset. The learned topics were fed into four stacked convolutional neural networks (CNNs); then, the output of the last CNN layer was input into a global max-pooling layer. The word embeddings were fed into a GRU layer. The outputs of the global max-pooling layer and the GRU layer were merged and fed into a DNN with a rectified linear unit (ReLU) activation function. The classification was performed by logistic regression. The performance of the C-GRU model was evaluated on the SemEval-2018 dataset. The results achieved by this model exceeded the results obtained by Badaro *et al.* [2], who ranked first on the leaderboard of the SemEval-2018 competition.

Abdul-Mageed *et al.* [9] created DINA, a multi-dialect dataset for Arabic emotion analysis, by crawling Twitter between July and October of 2015. The annotation process was conducted using two annotators who were native speakers of Arabic with postgraduate education. The annotators were provided with several examples and were advised to consult with each other, talk to their friends, and ask online on cases where a given dialect was not understandable. Their analysis shows the effectiveness of the phrase-based seed approach for automatically acquiring emotion data. Al-Khatib and El-Beltagy [10] also created a dataset for emotion recognition from tweets. More information on the AETD dataset is presented in section IV-A Datasets. The preprocessing steps included removing diacritics, links, mentions, and retweet indicators and normalization, where أ, إ, and آ were replaced by ا; ى was replaced by; ي; ة was replaced by ه; and

TABLE 1. Examples of sentences before and after preprocessing.

Dataset	Preprocessing status	Sentence
SemEval-2018	Before	وجع بطني مب وقته ابد :( )
	After	وجع بطني مب وقته ابد 😞
	Before	😤 والله كفايه @audinasser
	After	😤 والله كفايه
	Before	✨💔 أخاف قلبي يبطل يتحمل
	After	💔 أخاف قلبي يبطل يتحمل
IAEDS	Before	ادعـــولـــي باجـر اول امـتـحـــانـ دعـــووة الـــغـــريـــب مـســتـجـابـة
	After	ادعولي باجر اول امتحان دعووة الغريب مستجابة
	Before	!..انا لست# قبيح • انت فقط لا املك المال
	After	! لست قبيح فقط لا املك المال
	Before	ليس هناك مرض يقتل جسدك أكثر من التفكير الزائد
	After	ليس مرض يقتل جسدك التفكير الزائد
AETD	Before	هذا لي انا خايفه منه 💔😭 اخر جمله
	After	خايفه 💔😭 اخر جمله
	Before	💔 احس اني خايفه اروح اكثر من اني خايفه من الاختبار
	After	💔 احس خايفه اروح اكثر خايفه الاختبار
	Before	:D نيشيكوري بوظ عليا الاوليمبياد كلها النهارده
	After	😃 نيشيكوري بوظ عليا الاوليمبياد كلها النهارده

Arabic numerals replaced Hindi numerals. They used n-gram features and tested different classification algorithms, including naïve Bayes (NB), Complement NB [11], and sequential minimal optimization (SMO). The experiments showed that Complement NB outperformed the other models and achieved the highest results in terms of accuracy, $P^{weighted}$, $R^{weighted}$, and $F^{weighted}$.

Almahdawi and Teahan [12] created a dataset (IAEDS) for emotion recognition from Facebook posts. More information on the IAEDS dataset is presented in section IV-A Datasets. They performed two experiments. In the first experiment, WEKA[1] (Waikato Environment for Knowledge Analysis) was used to extract n-grams as features and tested with five classifiers, ZeroR, J48, NB, multinomial naïve Bayes (MNB) for text, and SVM with SMO. ZeroR and MNB resulted in the worst performances. In the second experiment, a compression-based classifier called prediction by partial matching (PPM) [13] was tested. The results showed that the PPM classifier significantly outperformed the other classifiers and achieved the highest results in terms of accuracy, precision, recall, and F-score.

III. PROPOSED MODELS

This section presents the proposed models for emotion recognition in Arabic text.

A. PREPROCESSING

The performances of the proposed models were tested on three datasets, All of which were created from social media platforms; for more details, see Section IV-A Datasets. The writing style used in social media is informal, contains grammatical and spelling mistakes, and includes hashtags, emoticons, and emojis. Table 1 shows some examples of sentences before and after preprocessing. The preprocessing pipeline includes the following:

- Using regular expressions to:
  - Normalize emoticons:
    - The positive emoticons were replaced by this (😃) emoji.
    - The negative emoticons were replaced by this (😞) emoji.
  - Normalize Arabic alphabets; for example, ک and گ were replaced by ك.
  - Replace the question mark emoji, and exclamation mark emoji with the characters ؟ and "!", respectively.
  - Remove numbers.
  - Remove symbols and special characters.
  - Remove English words.
  - Remove single-character words.
- Use Tashaphyne [14], which is an Arabic light stemmer, to remove diacritics (tashkeel) and tatweel.
- Use the natural language toolkit[2] (NLTK) TweetTokenizer[3] to tokenize the sentences.
- Remove stop words (except for negation words).
- Remove nonemotional emojis, such as flags, foods, tools, etc.

B. HUMAN-ENGINEERED FEATURE-BASED MODEL

This section presents the HEF model. Figure 1 shows a diagram of this model.

1) FEATURE SET

We selected features that represented different aspects of the text including stylistic, lexical, syntactic, and semantic

[1] https://www.cs.waikato.ac.nz/ ml/weka/

[2] https://www.nltk.org

[3] https://www.nltk.org/api/nltk.tokenize.html

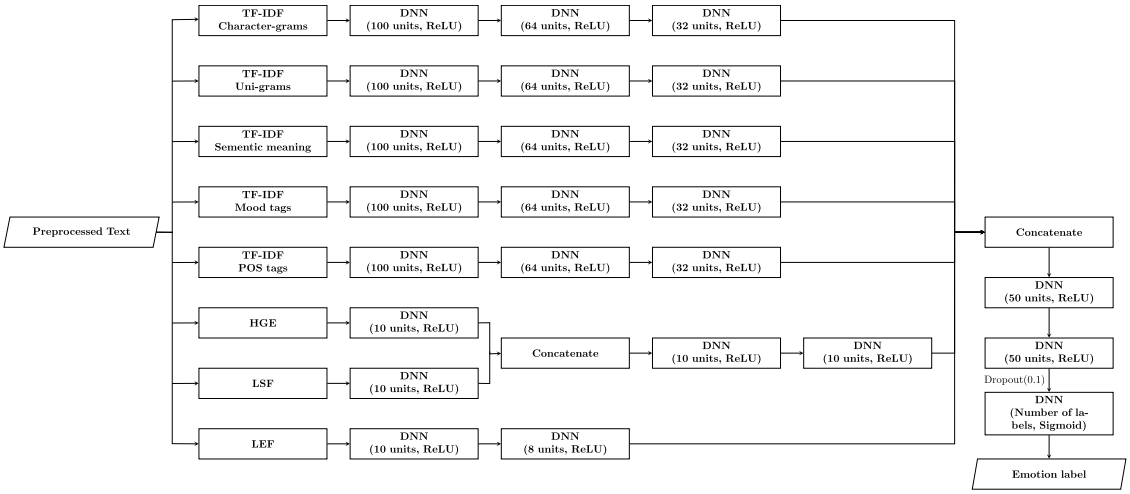

FIGURE 1. HEF model.

features. After text preprocessing, we extracted the following features:

- Domain-specific features: SenticNet [15] was used to retrieve the mood tag of each word in the dataset. Then, each word was replaced by its mood tag. Words without mood tags were deleted. Finally, the TF-IDF was calculated. Table 2 shows some examples of sentences and the mode tags assigned to their words.
- Linguistic features:
  - The TF-IDF of the character-grams: The number of *characters* ranges between one and ten.
  - The TF-IDF of the uni-grams.
- Lexical features:
  - Lexical sentiment features (LSF): The sentiment of sentences was calculated by summing the word sentiment score provided by of the following lexicons: Arabic Twitter sentiment lexicon [16], Arabic emoticon lexicon [17], [18], Arabic hashtag lexicon [17], [18], and Arabic hashtag lexicon dialectal [17], [18].
  - Lexical emotion features (LEF): The Arabic translation of NRC emotion lexicon[4] lists words. For each word, it provides a value of either zero or one for the emotions, anger, anticipation, disgust, fear, joy, negative, positive, sadness, surprise, and trust. We excluded the negative and positive emotion indicators from the SemEval-2018 dataset and the negative, positive, anticipation, and trust emotion indicators from the AETD, AIEDS datasets. The LEFs were calculated for each sentence by counting the number of words matching each emotion from this lexicon.
- Syntactic features: The TF-IDF of the POS tags.
- Semantic features:
  - The TF-IDF of the semantic meaning: SenticNet was used to retrieve the semantic meaning of each word in the dataset. Then, the word was replaced by its semantic meaning. Finally, the TF-IDF was calculated.
  - Hourglass of emotions (HGE) [19]: SenticNet was used to retrieve the sensitivity, attention, pleasantness, and aptitude scores of each word in a sentence. Then, the scores for each emotion dimension were added.

[4]http://saifmohammad.com/WebDocs/Arabic%20Lexicons/nrc_emotion_ar.txt

2) HEF MODEL

Three DNNs containing 100, 64 and 32 units, respectively, and a ReLU activation function were trained on each of the TF-IDF features. The HGE and LSF were trained with a DNN with ten units and a ReLU activation function. The outputs of both DNNs were concatenated and fed into two DNNs with ten units and a ReLU activation function. The LEFs were trained with two DNNs with ten and eight units and a ReLU activation function. To perform the classification, the outputs from all the previous DNNs were concatenated and passed into two DNNs with 50 units and a ReLU activation function. Finally, a dropout of value 0.1 was added to avoid overfitting, and a DNN whose units were equal to the number of emotion labels and a sigmoid activation function was added as an output layer.

C. DEEP FEATURE-BASED MODEL

This section presents the DF model. Figure 2 shows a diagram of this model.

1) PRE-TRAINED EMBEDDINGS

The available datasets for the Arabic language are small; however, deep learning requires large amounts of data for training. Thus, we used pre-trained word embeddings to serve

TABLE 2. Examples of sentences and their mood tags.

Arabic		English Translation	
Sentence	Mood tags	Sentence	Mood tags
اسوء يوم في حياتي لفقدانك يابي	غضب# قرف#	The worst day of my life is because your lost, dad	#disgust #anger
الحزن دائما أنيق لايختار الا القلوب الطيبة ليسكنها	مفاجأة# حزن# اعجاب#	Sadness is always elegant it only chooses the good hearts to settle down	#surprise #sadness #admiration
بعد فراق ٨ سنوات شفت بغداد وأهلي واحبابي	فرحة# مفاجأة#	I saw Baghdad and my family after 8 years separation	#surprise #joy

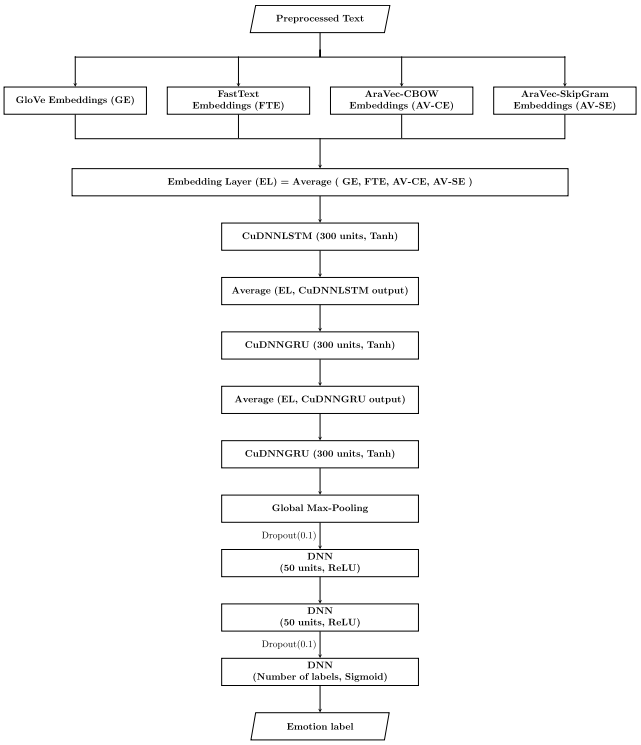

FIGURE 2. DF model.

as a means of transfer learning to train the deep learning models. However, not all words are represented in the pre-trained embedding models. This OOV word problem was solved by using character embeddings,[5] which were obtained by taking the average of the embeddings of all the words containing each character. The pre-trained embedding models used are as follows:

- Emoji2vec [20]: 300-dimensional emoji vectors learned from their description in the Unicode emoji standard.[6]
- GloVe [21]: 300-dimensional word vectors trained on tweets. To obtain the 300-dimensional word vectors, we concatenated the 200-dimensional and 100-dimensional word vectors.
- AraVec [5]: 300-dimensional word vectors trained on tweets. Two uni-gram models were used: continuous bag-of-words (CBOW) and SkipGram.
- FastText [22]: 300-dimensional word vectors trained on Common Crawl[7] using CBOW with position weights. FastText provides two models for Arabic, Arabic and Egyptian Arabic. We used the Arabic model.

For example, to build an embedding matrix from GloVe, we performed the following:

- Calculate the characters' embeddings[5].
- Use the emoji embeddings from the emoji2vec embeddings.
- Use the word embeddings if they are represented in GloVe.
- Use the word stem embeddings if the word is not represented in GloVe.
- If the word stem is not represented in GloVe, substitute the sum of the embeddings of the characters that comprise the word.

These steps were repeated three more times while varying only the source of the pre-trained embeddings. In the above example, we used GloVe; for the other matrices, we used AraVec-CBOW, AraVec-SkipGram, and FastText.

2) DF MODEL

We utilized different deep neural networks from the Keras[8] deep learning library. After text preprocessing, we built four embedding matrices and used them to create four embedding layers. Then, the average of the four embedding layers was fed into a CuDNNLSTM (long short-term memory built with the NVIDIA CUDA®deep neural network library) with 300 units and a tanh activation function. Then, the average of the CuDNNLSTM output and the averaged embedding layer was fed into a CuDNNGRU (gated recurrent unit built with the NVIDIA CUDA®deep neural network library) with 300 units and a tanh activation function. Next, the average of the CuDNNGRU output and the averaged embedding layer was fed into a CuDNNGRU with 300 units and a tanh activation function. Global max-pooling was conducted on the output of the last CuDNNGRU. A dropout value of 0.1 was added to help avoid overfitting. The same classification method used in the HEF model was used here.

D. HYBRID MODEL HEF+DF

This section presents the hybrid model HEF+DF. A diagram of this model is shown in Figure 3. The features from the

[5]https://github.com/minimaxir/char-embeddings
[6]https://unicode.org/emoji/charts/full-emoji-list.html
[7]https://commoncrawl.org/
[8]https://keras.io

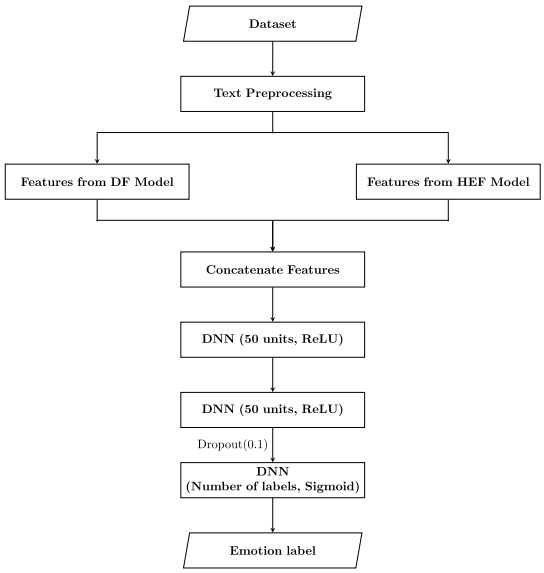

FIGURE 3. HEF+DF model.

HEF and DF models were concatenated, Algorithm 1 shows the pseudocode for the concatenation. As input, it takes the features from those two models (all of which should have the same shape except for the concatenation axis) and returns a single output, the concatenation of all inputs. The output of the concatenation was fed into two DNNs with 50 units and a ReLU activation function. A dropout of value 0.1 was added to avoid overfitting. Finally, a DNN with units equal to the number of emotion labels and a sigmoid activation function was added as an output layer for the classification of the emotions.

Algorithm 1 Concatenation

Input: Features F
Number of Features n
Output: The concatenation of the Features *ConcatF*

```
axis = −1, the dimension along which to concatenate
[D_1, D_2, ..., D_n] = F.shape
if (n ≥ 2) then
    if (D_1 == ... == D_n except for the concatenation axis) then
        ConcatF = F_1
        for (i: 2 to n) do
            Expand ConcatF dimension (D^i, axis)
            ConcatF = Concatenate(ConcatF, F^i)
        end for
    else
        print("Concatenation requires inputs with matching shapes, except for the concatenation axis")
    end if
else
    print("Concatenation must have a list of at least 2 inputs")
end if
```

IV. EXPERIMENTS

The proposed emotion recognition models were implemented in Python. We used the following libraries: NLTK[2], Tashaphyne, scikit-learn [23], and Keras[8] deep learning with a TensorFlow[9] backend and the Google Colaboratory[10] platform running on a 25-GB GPU.

A. DATASETS

In this section, we present the datasets used to evaluate the performance of the proposed emotion recognition models. Tables 3, 4, and 5 show the emotion labels, the number of instances in each label, and the distribution percentages of those instances in the AETD dataset, IAEDS dataset and the SemEval-2018 dataset, respectively.

- AETD [10]: This dataset consists of tweets mostly in the Egyptian dialect. The total number of instances is 10,065, and each instance is labeled as anger, fear, happiness, love, sadness, surprise, sympathy, or none. The distributions, as shown in Table 3, range from 10.38% to 15.40% for surprise and none, respectively.

[9]https://www.tensorflow.org
[10]https://colab.research.google.com/notebooks/welcome.ipynb

TABLE 3. Description of AETD dataset.

Emotion	Number of Instances	Distribution (%)
Anger	1444	14.35
Fear	1207	11.99
Happiness	1281	12.73
Love	1220	12.12
Sadness	1256	12.48
Surprise	1045	10.38
Sympathy	1062	10.55
None	1550	15.40

TABLE 4. Description of IAEDS dataset.

Emotion	Number of Instances	Distribution (%)
Anger	310	22.71
Disgust	185	13.55
Fear	148	10.84
Happiness	256	18.75
Sadness	238	17.44
Surprise	229	16.78

- IAEDS [12]: This dataset consists of Facebook posts in the Iraqi dialect. The dataset can be obtained by contacting the authors. It is divided into six files, and each file consists of instances belonging to anger, disgust, fear, happiness, sadness, or surprise. The total number of instances is 1,365. The distributions, as shown in Table 4, range from 10.84% to 22.71% for fear and anger, respectively.
- SemEval-2018 [1]: This dataset consists of tweets split into training, development, and testing datasets. Each instance is labeled as neutral or as one or more of eleven

TABLE 5. Description of SemEval-2018 dataset.

Emotion	Number of Instances			Distribution (%)		
	Train	Development	Test	Train	Development	Test
Anger	899	215	609	17.41	15.51	17.14
Anticipation	206	57	158	3.99	4.11	4.45
Disgust	433	106	316	8.38	7.65	8.89
Fear	391	94	295	7.57	6.78	8.30
Happiness	605	179	393	11.71	12.91	11.06
Love	562	175	367	10.88	12.63	10.33
Optimism	561	169	344	10.86	12.19	9.68
Pessimism	499	125	377	9.66	9.02	10.61
Sadness	842	217	579	16.30	15.66	16.30
Surprise	47	13	38	0.91	0.94	1.07
Trust	120	36	77	2.32	2.60	2.18

TABLE 6. Parameter values.

Parameter	AETD	IAEDS	SemEval-2018
Patch size	32	32	32
Epochs	7	10	10
Learning rate	0.001	0.001	0.001
Loss function	Categorical	Categorical	Binary
Optimizer	Adam	Adam	Adam

emotions: anger, anticipation, disgust, fear, happiness, love, optimism, pessimism, sadness, surprise, and trust. Therefore, the total number of instances may be less than the total associated with each emotion label in Table 5.

- Training dataset: The total number of instances is 2,278. The distributions range from 0.91% to 17.41% for surprise and anger, respectively.
- Development dataset: The total number of instances is 585. The distributions range from 0.94% to 15.66% for surprise and sadness, respectively.
- Test dataset: The total number of instances is 1,518. The distributions range from 1.07% to 17.14% for surprise and anger, respectively.

B. EVALUATION MEASURES

To evaluate the performance of the proposed models, we used the following metrics: multi-label accuracy (Jaccard accuracy) (Eq. 1), accuracy (Eq. 2), F^{micro} (Eq. 5), P^{macro} (Eq. 9), R^{macro} (Eq. 10), F^{macro} (Eq. 11), $P^{weighted}$ (Eq. 12), $R^{weighted}$ (Eq. 13), and $F^{weighted}$ (Eq. 14).

$$Jaccard\ \ accuracy = \frac{1}{|S|}\sum_{s\in S}\frac{|G_s \cap P_s|}{|G_s \cup P_s|} \tag{1}$$

where G_s is the set of gold labels for sentence s, P_s is the set of predicted labels for sentence s, and S is the set of sentences.

$$Accuracy = \frac{\sum_{e\in E} TP + \sum_{e\in E} TN}{\sum_{e\in E} TP + \sum_{e\in E} TN + \sum_{e\in E} FP + \sum_{e\in E} FN}, \tag{2}$$

where E is the set of emotion labels, TP is the number of true positives, TN is the number of true negatives, FP is the number of false positives, and FN is the number of false negatives. The TP, FP, FN, and TN values were calculated as follows:

- TP: For a given label, if that label occurs in both the set of gold labels and the set of predicted labels, then increment by one.
- FP: For a given label, if that label occurs in the set of predicted labels but not in the set of gold labels, then increment by one.
- FN: For a given label, if that label occurs in the set of gold labels but not in the set of predicted labels, then increment by one.
- TN: The total number of occurrences that are not a given label minus the FP of that label.

For the micro-averaged results, the TP, FP and FN for each emotion label e are summed and the average is taken. The micro-averaged precision (P^{micro}) and micro-averaged recall (R^{micro}) are calculated as follows:

$$P^{micro} = \frac{\sum_{e\in E} TP}{\sum_{e\in E} TP + \sum_{e\in E} FP} \tag{3}$$

$$R^{micro} = \frac{\sum_{e\in E} TP}{\sum_{e\in E} TP + \sum_{e\in E} FN}, \tag{4}$$

where F^{micro} is the harmonic mean of the above two equations:

$$F^{micro} = 2 \cdot \frac{P^{micro} \times R^{micro}}{(P^{micro} + R^{micro})}. \tag{5}$$

For the macro-averaged results, the precision and recall are calculated independently for each emotion label e, and then the average is taken. Hence, all the emotion label are treated equally.

$$precision_e = \frac{TP_e}{TP_e + FP_e} \tag{6}$$

$$reacll_e = \frac{TP_e}{TP_e + FN_e}. \tag{7}$$

The F-score$_e$ is the harmonic mean of the above two equations.

$$f\text{-}score_e = 2 \cdot \frac{precision_e \times reacll_e}{(precision_e + reacll_e)} \tag{8}$$

The P^{macro} and R^{macro} are calculated as follows:

$$P^{macro} = \frac{1}{|E|}\sum_{e\in E} precision_e \tag{9}$$

$$R^{macro} = \frac{1}{|E|}\sum_{e\in E} recall_e, \tag{10}$$

and the F^{macro} is the harmonic mean of the above two equations:

$$F^{macro} = 2 \cdot \frac{P^{macro} \times R^{macro}}{(P^{macro} + R^{macro})}. \tag{11}$$

The weighted average considers label imbalance and can result in an $F^{weighted}$ that is not between $P^{weighted}$ and

TABLE 7. Comparison results of the proposed models.

Dataset	Label	Precision			Recall			F-score		
		HEF	DF	HEF+DF	HEF	DF	HEF+DF	HEF	DF	HEF+DF
SemEval-2018	Anger	0.709	**0.780**	0.716	0.739	0.688	**0.780**	0.724	0.731	**0.747**
	Anticipation	**0.406**	0.097	0.177	0.082	0.038	**0.139**	0.137	0.055	**0.156**
	Disgust	0.476	**0.513**	0.496	0.190	0.430	**0.522**	0.272	0.468	**0.509**
	Fear	0.614	**0.695**	0.690	0.556	0.603	**0.678**	0.584	0.646	**0.684**
	Happiness	**0.798**	0.728	0.733	0.713	0.858	**0.860**	0.753	0.787	**0.792**
	Love	0.603	**0.632**	0.595	0.719	0.809	**0.817**	0.656	**0.710**	0.689
	Optimism	**0.682**	0.609	0.643	0.704	**0.765**	0.747	**0.692**	0.678	0.691
	Pessimism	**0.418**	0.372	0.389	0.236	**0.618**	0.456	0.302	**0.465**	0.420
	Sadness	0.640	0.582	**0.652**	0.572	**0.845**	0.720	0.604	**0.689**	0.684
	Surprise	0.000	**0.200**	0.000	0.000	**0.026**	0.000	0.000	**0.047**	0.000
	Trust	0.143	0.175	**0.209**	0.026	0.091	**0.117**	0.044	0.120	**0.150**
IAEDS	Anger	0.532	**0.545**	0.508	0.628	0.597	**0.770**	0.576	0.570	**0.612**
	Disgust	0.696	0.547	**0.722**	0.361	**0.391**	0.350	**0.476**	0.457	0.472
	Fear	0.768	0.706	**0.818**	0.540	**0.568**	0.556	0.634	0.629	**0.662**
	Happiness	0.805	**0.883**	0.842	**0.722**	0.663	0.675	**0.761**	0.757	0.750
	Sadness	0.777	0.737	**0.827**	0.642	0.676	**0.731**	0.703	0.705	**0.776**
	Surprise	0.424	**0.452**	0.4448	**0.558**	0.547	0.554	0.482	**0.495**	0.493
AETD	Anger	0.630	**0.717**	0.707	0.670	0.703	**0.755**	0.649	0.710	**0.730**
	Fear	**0.969**	0.948	0.939	0.854	0.901	**0.909**	0.908	**0.924**	**0.924**
	Happiness	0.566	0.607	**0.641**	0.506	**0.603**	0.584	0.534	0.605	**0.611**
	Love	0.722	**0.789**	0.782	0.705	0.733	**0.759**	0.713	0.760	**0.771**
	Sadness	0.421	0.553	**0.605**	0.497	**0.549**	0.502	0.456	**0.551**	0.549
	Surprise	**0.589**	0.576	0.576	0.486	0.482	**0.542**	0.533	0.525	**0.558**
	Sympathy	**0.894**	0.879	0.876	0.809	**0.878**	0.870	0.849	**0.878**	0.873
	None	0.649	0.670	**0.674**	0.723	0.794	**0.805**	0.684	0.727	**0.734**

$R^{weighted}$, $P^{weighted}$, $R^{weighted}$, and $F^{weighted}$ are calculated as follows:

$$P^{weighted} = \frac{\sum_{e \in E} precision_e \times size(e)}{size(dataset)} \quad (12)$$

$$R^{weighted} = \frac{\sum_{e \in E} recall_e \times size(e)}{size(dataset)} \quad (13)$$

$$F^{weighted} = \frac{\sum_{e \in E} f\text{-}score_e \times size(e)}{size(dataset)}. \quad (14)$$

C. PERFORMANCE RESULTS

In this section, we report the performance results and discuss the evaluation of the proposed models. For the SemEval-2018 dataset, we trained the models on the training dataset and report on the models' performance on the test dataset. The best performance results for the HEF+DF model were achieved when combining the TF-IDF of the uni-grams, TF-IDF of the POS tags, HGE, LSF, and LEF with the DF model. For the IAEDS dataset, stratified 10-fold cross-validation was performed to ensure that the percentages of each class in the dataset were equal within each fold. The best performance results for the HEF+DF model were achieved when combining the TF-IDF of the character-grams, TF-IDF of the POS tags, HGE, LSF, and LEF with the DF model. For the AETD dataset, stratified 10-fold cross-validation was performed. The best performance results for the HEF+DF model were achieved when combining the HGE, LSF, and LEF with the DF model. The best performance results for the HEF model on all three datasets were achieved when all the human-engineered features were used. Table 6 shows the hyperparameter values, and Table 7 shows comparison results of the proposed models. Tables 8, 9, and 10 show the comparison results of the proposed models with state-of-the-art models on the SemEval-2018, IAEDS, and AETD datasets, respectively.

TABLE 8. Comparison results of the proposed models with state-of-the-art models on the SemEval-2018 dataset.

Model	Measures		
	Jaccard accuracy	F^{micro}	F^{macro}
DF	0.505	0.627	0.490
HEF	0.448	0.583	0.433
HEF+DF	0.512	0.631	**0.502**
Abdullah and Shaikh [4]	0.446	0.572	0.447
Badaro et al. [2]	0.489	0.618	0.461
Mulki et al. [3]	0.465	0.597	0.446
Samy et al. [7]	**0.532**	**0.648**	0.495

TABLE 9. Comparison results of the proposed models with the state-of-the-art models on the IAEDS dataset.

Model	Measures			
	Accuracy	P^{macro}	R^{macro}	F^{macro}
DF	86.1	0.64	0.57	0.61
HEF	86.4	0.67	0.58	0.62
HEF+DF	**87.2**	**0.69**	**0.60**	**0.64**
Almahdawi and Teahan [12]	87.1	0.63	0.59	0.61

1) COMPARISON OF THE PROPOSED MODELS

On the SemEval-2018 dataset, the HEF+DF model performed better than the DF and HEF models, and the DF model performed better than the HEF model. The HEF+DF model outperformed the DF model, achieving improvements

TABLE 10. Comparison results of the proposed models with the state-of-the-art models on the AETD dataset.

Model	Measures			
	Accuracy	weighted average		
		P	R	F
DF	0.708	0.714	0.708	0.710
HEF	0.658	0.674	0.658	0.664
HEF+DF	**0.718**	**0.722**	**0.718**	**0.718**
Al-Khatib and El-Beltagy [10]	68.12	0.688	0.681	0.658

of 0.7%, 0.4%, and 1.2% in Jaccard accuracy, F^{micro}, and F^{macro}, respectively. The DF model outperformed the HEF model, achieving improvements of 5.7%, 4.4%, and 5.7% in Jaccard accuracy, F^{micro}, and F^{macro}, respectively.

On the IAEDS dataset, the HEF+DF model performed better than the HEF and DF models, and the HEF model performed better than the DF model. The HEF+DF model outperformed the HEF model, achieving improvements of 0.8%, 2%, 3%, and 2% in accuracy, P^{macro}, R^{macro}, and F^{macro}, respectively. The HEF model outperformed the DF model, achieving improvements of 0.3%, 3%, 1%, and 1% in accuracy, P^{macro}, R^{macro}, and F^{macro}, respectively.

On the AETD dataset, the HEF+DF model performed better than the DF and HEF models, and the DF model performed better than the HEF model. The HEF+DF model outperformed the DF model, achieving improvements of 1%, 0.8%, 1%, and 0.8% in accuracy, $P^{weighted}$, $R^{weighted}$, and $F^{weighted}$, respectively. The DF model outperformed the HEF model, achieving improvements of 5%, 4%, 5%, and 4.6% in accuracy, $P^{weighted}$, $R^{weighted}$, and $F^{weighted}$, respectively.

2) PERFORMANCE COMPARISON BASED ON EMOTION LABELS

Precision The performance results on the SemEval-2018 dataset showed that the HEF model achieved the highest performance results for anticipation, happiness, optimism, and pessimism, while the DF model achieved the highest performance results for anger, disgust, fear, love, and surprise. The HEF+DF model achieved the highest performance results for sadness and trust. For the emotion labels on which either the HEF model or the DF model achieved the highest result, the HEF+DF model achieved the second-best results with one exception: the only time that the HEF+DF model performance result came in last was for the emotion label love; however, the difference between it and the HEF model was insignificant. The best performance results on the IAEDS dataset were achieved by either the DF model or the HEF+DF model. The HEF model came in a close second to the DF model for anger (a 1.37% difference). Moreover, the HEF model was second to the HEF+DF model for disgust, fear, and sadness (differences of 2.57%, 5.06%, and 5%, respectively). The performance results on the AETD dataset show that the HEF model achieved the highest performance results for fear, surprise, and sympathy, while the DF model achieved the highest performance results for anger and love. The HEF+DF model achieved the highest performance results for happiness, sadness, and none. For the emotion labels in which the DF model achieved the highest result, the HEF+DF model achieved the second-best results. However, for the emotion labels in which the HEF model achieved the highest result, the DF model achieved the second-best results.

Recall The best performance results on the SemEval-2018 dataset were consistently achieved by either the DF model or the HEF+DF model. The HEF model was second to the HEF+DF model for anger and anticipation with 4.11% and 5.69% differences, respectively. The performance results on the IAEDS dataset show that the HEF model achieved the highest performance results for happiness and surprise; the DF model achieved the highest performance results for disgust and fear; and the HEF+DF model achieved the highest performance results for anger and sadness. For the emotion labels in which either the HEF model or the DF model achieved the highest result, the HEF+DF model achieved the second-best results except for the emotion label disgust, where it came in last, but with only an insignificant difference between it and the HEF model. The best performance results on the AETD dataset were achieved by either the DF model or the HEF+DF model. The HEF+DF model came in a close second to the DF model for happiness and sympathy (1.9% and 0.8% differences, respectively).

F-score The performance results on the SemEval-2018 dataset showed that the HEF+DF model achieved the highest performance results for anger, anticipation, disgust, fear, happiness, and trust, while the DF model achieved the highest performance results for love, pessimism, sadness, and surprise, and the HEF model achieved the highest performance results for optimism. For the emotion labels in which either the HEF model or the DF model achieved the highest result, the HEF+DF model achieved the second-best results. Moreover, the difference was insignificant between the HEF+DF model and the first-place model for optimism and sadness. The performance results on the IAEDS dataset show that the HEF+DF model achieved the highest performance results for anger, fear, and sadness, and the HEF model achieved the highest performance results for disgust and happiness. Although the HEF model and the DF model achieved the highest performance results for disgust and surprise, respectively, the differences between their results and the results achieved by the HEF+DF model were insignificant. The best performance results on the AETD dataset were mostly achieved by the HEF+DF model, and came in a close second to the DF model for sadness and sympathy with 0.2% and 0.5% differences, respectively.

D. DISCUSSION

In this section, we discuss the performances of the HEF, DF, and HEF+DF models in light of the result presented in Section IV-C Performance Results.

1) COMPARISON WITH STATE-OF-THE-ART

On the SemEval-2018 dataset, the HEF+DF model outperformed the Samy *et al.* [7] model, achieving a 0.7%

improvement in F^{macro}. However, Samy *et al.* [7] outperformed the HEF+DF model, achieving 2% and 1.7% differences in Jaccard accuracy and F^{micro}, respectively. Moreover, the HEF+DF model outperformed Badaro *et al.* [2] by 2.3%, 1.3%, and 4.1% on Jaccard accuracy, F^{micro}, and F^{macro}, respectively. It also outperformed Mulki *et al.* [3] by 4.7%, 3.4%, and 5.6% on Jaccard accuracy, F^{micro}, and F^{macro}, respectively. Finally, it outperformed the Abdullah and Shaikh [4] model by 6.6%, 5.9%, and 5.5% on Jaccard accuracy, F^{micro}, and F^{macro}, respectively. The DF model outperformed Badaro *et al.* [2] by 1.6%, 0.9%, and 2.9% in terms of Jaccard accuracy, F^{micro}, and F^{macro}, respectively. It also outperformed Mulki *et al.* [3] by 4%, 3%, and 4.3% on Jaccard accuracy, F^{micro}, and F^{macro}, respectively. Last, it outperformed Abdullah and Shaikh [4] by 5.9%, 5.5%, and 4.4% on Jaccard accuracy, F^{micro}, and F^{macro}, respectively. However, Samy *et al.* [7] outperformed the DF model by 2.7%, 2.1%, and 0.5% on Jaccard accuracy, F^{micro}, and F^{macro}, respectively. The HEF model did not perform as well as the DF and HEF+DF models.

On the IAEDS dataset, the HEF+DF model outperformed the Almahdawi and Teahan [12] model by 0.1%, 6%, 1%, and 3% on accuracy, P^{macro}, R^{macro}, and F^{macro}, respectively. Moreover, the HEF model and the DF model outperformed the Almahdawi and Teahan [12] model on P^{macro} by 4% and 1%, respectively. The DF model achieved the same F^{macro} as the Almahdawi and Teahan [12] model, but the HEF model outperformed them by 1% improvement.

On the AETD dataset, the HEF+DF model outperformed the Al-Khatib and El-Beltagy [10] model by 3.7%, 3.4%, 3.7%, and 6% in accuracy, $P^{weighted}$, $R^{weighted}$, and $F^{weighted}$, respectively. Moreover, the DF model outperformed the Al-Khatib and El-Beltagy [10] model by 2.7%, 2.6%, 2.7%, and 5.2% in accuracy, $P^{weighted}$, $R^{weighted}$, and $F^{weighted}$, respectively. The HEF model outperformed the Al-Khatib and El-Beltagy [10] model by 0.6% in $F^{weighted}$, but the Khatib and El-Beltagy [10] model outperformed the HEF model by 2.3%, 1.4%, and 2.3% in accuracy, $P^{weighted}$, and $R^{weighted}$, respectively.

2) THE IMPACT OF HYBRIDIZING THE HEF AND DF MODELS

The AETD dataset size is almost ten times the size of the IAEDS dataset; however, the hybrid model HEF+DF outperformed the other two models on both these datasets. On the AETD dataset, which has 10,065 instances, the DF model performed better than did HEF model in terms of accuracy, $P^{weighted}$, $R^{weighted}$, and $F^{weighted}$, while on the IAEDS dataset, which only has 1,365 instances, the HEF model performed better than the DF model in terms of accuracy, P^{macro}, R^{macro}, and F^{macro}. Moreover, in terms of precision, the HEF model outperformed the DF model on both datasets for the emotion labels with the smallest number of instances (surprise, sympathy, and fear in the AETD dataset, and fear and disgust in the IAEDS dataset). These results show that the performance of the DF model is affected by the dataset size and the instance distribution of the emotion labels. They also show that hybridizing the two models improved the results by combining the strength of both models.

3) THE IMPACT OF IMBALANCED DATASETS

All three datasets are imbalanced, but the imbalance is greater in the SemEval-2018 dataset than in the IAEDS and AETD datasets. The emotion label with the largest number of instances in the SemEval-2018 dataset was anger, comprising 17.41% and 17.14% of the instances in the training and testing datasets, respectively. On the other hand, the surprise, trust, and anticipation emotion labels had the smallest number of instances—only 0.91%, 2.32%, 3.99% in the training dataset and 1.07%, 2.18%, and 4.45% in the testing dataset, respectively. All three models had difficulty recognizing the trust and surprise emotions; in fact, the HEF model and the HEF+DF model failed to recognize the surprise emotion.

4) EASY-TO-GRASP CHARACTERISTICS

Some emotions are easier to recognize than others. In the Semeval-2018 dataset, although the number of instances for the emotion label happiness was less than the number of instances for the emotion label anger, the F-score for recognizing the emotion happiness was higher than that for recognizing anger. Moreover, all the models recognized fear better than they did disgust or pessimism. In the IAEDS dataset, although the number of instances of the emotion label anger was the largest, the models were able to recognize sadness, happiness, and fear better than anger. Furthermore, while the emotion label fear had the smallest number of instances, the F-scores for recognizing the emotions disgust and surprise were lower than that for fear. In the AETD dataset, the models were able to recognize fear, sympathy, and love better than anger even though the emotion label anger has more instances. Hence, emotions, happiness, love, sadness, fear, and anger have characteristics and indicators that are easier to grasp.

V. CONCLUSION

In this paper, we proposed three models, the HEF model, the DF model, and the hybrid model HEF+DF, for emotion recognition in Arabic text. The DF model performed better than the HEF model on the SemEval-2018 dataset; however, the SemEval-2018 dataset was more imbalanced than the IAEDS dataset. Utilizing different pre-trained embedding models provided the DF model with a good starting point. Reinserting the embedding layer allowed the DF model time to learn by delaying the convergence caused by stacking deep neural networks and training on a small dataset. Moreover, it improved the prediction of emotion labels with only small numbers of instances, such as surprise. Combining the HEF model with the DF model achieved the highest performance in terms of F^{macro}. Although the HEF+DF model improved the predictions on the majority of the emotion labels, the limitations of the HEF model affected its prediction of some emotion labels. Nevertheless, the HEF model performed better

than the DF model when tested on the IAEDS dataset, which was smaller than the AETD and SemEval-2018 datasets. The performance of the DF model was affected dataset size. Combining the HEF model with the DF model achieved the highest performance results on the IAEDS dataset in terms of accuracy, P^{macro}, R^{macro}, and F^{macro}; however, the DF model performed better than the HEF model when tested on the AETD dataset, which is larger than the SemEval-2018 and IAEDS datasets. Combining the HEF model with the DF model achieved the highest performance result on the AETD dataset in terms of accuracy, $P^{weighted}$, $R^{weighted}$, and $F^{weighted}$.

People tend to use strong words and more emojis when expressing happiness, love, sadness, fear, and anger, which makes it easier to recognize those emotions. We used the NRC emotion lexicon to help improve the recognition of anticipation, disgust, surprise, and trust. Nevertheless, the NRC emotion lexicon is a translated lexicon. Creating emotion lexicons specifically for Arabic would help improve the recognition of emotions that lack distinct characteristics and indicators.

In the future, we plan to investigate how to represent words that share the same spelling but have different meanings. We noticed this problem when we dealt with ADs. For example, consider the word كذاب (English translation: liar). In regions nearest to the Arabian Gulf, the letter ك is pronounced (cha) instead of (ka), and when writing ك, some people replace it with ج. A word such as كذاب could be written as جذاب (English translation: attractive). Hence, a sentence such as (this girl is lying) written in an Iraqi dialect can become هاي البنت جذابه (English translation: this girl is attractive). FastText provided a pre-trained word embedding model in Egyptian Arabic. Providing pre-trained word embedding models for other ADs would help solve such problems.

Deep learning requires large datasets for training. Using pre-trained word embeddings helps minimize the effect of the absence of a large training dataset; however, we still needed to address the OOV word problem. We overcame that by calculating characters' embeddings[5] from the available pre-trained word embedding models, but a robust solution to solve this problem is still needed. Finally, more research should be conducted to improve emotion recognition in Arabic and boost Arabic NLP.

ACKNOWLEDGMENT

The authors are grateful for the support provided by the Research Center of the College of Computer and Information Sciences at King Saud University. We also thank Dr. William Teahan and Dr. Amer Almahdawi for providing the IAEDS dataset to us and allowing its use in our research.

REFERENCES

[1] S. Mohammad, F. Bravo-Marquez, M. Salameh, and S. Kiritchenko, "SemEval-2018 task 1: Affect in tweets," in *Proc. 12th Int. Workshop Semantic Eval.*, New Orleans, LA, USA, 2018, pp. 1–17.

[2] G. Badaro, O. El Jundi, A. Khaddaj, A. Maarouf, R. Kain, H. Hajj, and W. El-Hajj, "EMA at SemEval-2018 task 1: Emotion mining for arabic," in *Proc. 12th Int. Workshop Semantic Eval.*, New Orleans, LA, USA, 2018, pp. 236–244.

[3] H. Mulki, C. Bechikh Ali, H. Haddad, and I. Babaoglu, "Tw-StAR at SemEval-2018 task 1: Preprocessing impact on multi-label emotion classification," in *Proc. 12th Int. Workshop Semantic Eval.*, 2018, pp. 167–171.

[4] M. Abdullah and S. Shaikh, "TeamUNCC at SemEval-2018 task 1: Emotion detection in english and arabic tweets using deep learning," in *Proc. 12th Int. Workshop Semantic Eval.*, New Orleans, LA, USA, 2018, pp. 350–357.

[5] A. B. Soliman, K. Eissa, and S. R. El-Beltagy, "AraVec: A set of arabic word embedding models for use in arabic NLP," *Procedia Comput. Sci.*, vol. 117, pp. 256–265, 2017.

[6] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching word vectors with subword information," *Trans. Assoc. for Comput. Linguistics*, vol. 5, pp. 135–146, Dec. 2017.

[7] A. E. Samy, S. R. El-Beltagy, and E. Hassanien, "A context integrated model for multi-label emotion detection," *Procedia Comput. Sci.*, vol. 142, pp. 61–71, 2018.

[8] S. Rosenthal, N. Farra, and P. Nakov, "SemEval-2017 task 4: Sentiment analysis in Twitter," in *Proc. 11th Int. Workshop Semantic Eval. (SemEval)*, Vancouver, BC, Canada, 2017, pp. 502–518.

[9] M. Abdul-Mageed, H. AlHuzli, and M. Duaa'Abu Elhija, "DINA: A multi-dialect dataset for Arabic emotion analysis," in *Proc. 2nd Workshop Arabic Corpora Process. Tools Theme, Social Media*, 2016, pp. 29–37.

[10] A. Al-Khatib and S. R. El-Beltagy, "Emotional tone detection in arabic tweets," in *Computational Linguistics and Intelligent Text Processing* (Lecture Notes in Computer Science), vol. 10762, A. Gelbukh, Ed. Cham, Switzerland: Springer, 2018, pp. 105–114.

[11] J. D. M. Rennie, L. Shih, J. Teevan, and D. R. Karger, "Tackling the poor assumptions of Naïve Bayes text classifiers," in *Proc. 20th Int. Conf. Int. Conf. Mach. Learn. (ICML)*, 2003, pp. 616–623.

[12] A. J. Almahdawi and W. J. Teahan, "A new arabic dataset for emotion recognition," in *Intelligent Computing*, vol. 998, K. Arai, R. Bhatia, and S. Kapoor, Eds. Cham, Switzerland: Springer, 2019, pp. 200–216.

[13] W. J. Teahan and D. J. Harper, "Using compression-based language models for text categorization," in *Language Modeling for Information Retrieval*, W. B. Croft and J. Lafferty, Eds. Dordrecht, Netherlands: Springer, 2003, pp. 141–165.

[14] T. Zerrouki. (2012). *Tashaphyne, Arabic Light Stemmer*. [Online]. Available: https://pypi.python.org/pypi/Tashaphyne/0.2

[15] E. Cambria, S. Poria, D. Hazarika, and K. Kwok, "SenticNet 5: Discovering conceptual primitives for sentiment analysis by means of context embeddings," in *Proc. AAAI Conf. Artif. Intell.*, New Orleans, LA, USA, 2018, pp. 1795–1802.

[16] S. Kiritchenko, S. Mohammad, and M. Salameh, "SemEval-2016 task 7: Determining sentiment intensity of english and arabic phrases," in *Proc. 10th Int. Workshop Semantic Eval. (SemEval-)*, San Diego, CA, USA, 2016, pp. 42–51.

[17] M. Salameh, S. Mohammad, and S. Kiritchenko, "Sentiment after translation: A case-study on arabic social media posts," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol.*, Denver, CO, USA, 2015, pp. 767–777.

[18] S. M. Mohammad, M. Salameh, and S. Kiritchenko, "How translation alters sentiment," *J. Artif. Intell. Res.*, vol. 55, no. 1, pp. 95–130, Jan. 2016.

[19] E. Cambria, A. Livingstone, and A. Hussain, "The hourglass of emotions," in *Cognit. Behavioural Syst.*, A. Esposito, A. M. Esposito, A. Vinciarelli, R. Hoffmann, and V. C. Müller, Eds. Berlin, Germany: Springer, 2012, pp. 144–157.

[20] B. Eisner, T. Rocktäschel, I. Augenstein, M. Bošnjak, and S. Riedel, "emoji2vec: Learning emoji representations from their description," in *Proc. 4th Int. Workshop Natural Lang. Process. Social Media*. Austin, TX, USA, 2016, pp. 48–54.

[21] J. Pennington, R. Socher, and C. Manning, "Glove: Global vectors for word representation," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*. Doha, Qatar, 2014, pp. 1532–1543.

[22] E. Grave, P. Bojanowski, P. Gupta, A. Joulin, and T. Mikolov, "Learning word vectors for 157 languages," in *Proc. 11th Int. Conf. Lang. Resour. Eval. (LREC)*, Paris, France, 2018, pp. 3483–3487.

[23] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, Oct. 2011.

NOURAH ALSWAIDAN received the master's degree in computer science from King Saud University, Saudi Arabia, in 2014, where she is currently pursuing the Ph.D. degree with the Department of Computer Science. Her main research interests include meta-heuristics, natural language processing, and machine learning.

MOHAMED EL BACHIR MENAI received the Ph.D. degree in computer science from the Mentouri University of Constantine, Algeria, and the University of Paris VIII, France, in 2005. He also received the postdoctoral degree "Habilitation Universitaire" in computer science from the Mentouri University of Constantine, in 2007 (it is the highest academic qualification in Algeria, France and Germany). He is currently a Professor with the Department of Computer Science, King Saud University. His main interests include satisfiability problems, evolutionary computing, natural language processing, machine learning, and AI in medicine.