Journal of Theoretical and Applied Information Technology
15th May 2023. Vol.101. No 9

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

EMOTION RECOGNITION IN ARABIC: A BERT-BASED TRANSFER LEARNING APPROACH LEVERAGING SEMANTIC INFORMATION OF ONLINE COMMENTS

MAHA JARALLAH ALTHOBAITI
Department of Computer Science, College of Computers and Information Technology, Taif University
P.O.BOX 11099, Taif 21944, Saudi Arabia
E-mail: maha.j@tu.edu.sa

ABSTRACT

With a wide range of practical applications, such as the diagnosis of mental health disorders and the detection of suspicious online behavior, the recognition of emotions in textual data is a crucial task. However, Arabic emotion recognition remains under-addressed in comparison to other languages, primarily due to the scarcity of labeled data. Pre-trained language models and transfer learning offer promising techniques to overcome the scarcity of labeled data for downstream tasks, such as emotion recognition. In this paper, we comprehensively explore the utilization of pre-trained Bidirectional Encoder Representations from Transformers (BERT) models for Arabic fine-grained emotion recognition. We further propose a straightforward yet effective method for enhancing transfer learning in emotion recognition through the integration of semantic information, specifically, information extracted based on sentiment analysis and named entity recognition. To evaluate our proposed method, we conduct experiments on an existing benchmark dataset using pre-trained BERT models. The results indicate that our proposed approach of integrating the named entity information with the BERT model yielded a high weighted-average F1 score of 81.60%. Compared to the existing studies on Arabic emotion recognition in the literature, our proposed method outperforms the state-of-the-art approach, yielding of 9.80%, 9.72%, and 9.60% in weighted-average F1 score, precision, and recall respectively.

Keywords: *Natural Language Processing, Emotion Recognition, BERT, Pre-trained Language Models, Semantic Information.*

1. INTRODUCTION

Arabic emotion recognition is an emerging research area, with most of the available studies for emotion recognition in Arabic being focused on spoken rather than written texts [1], [2]. As a result, there are a handful of emotion-annotated corpora collected from social media [3], [4]. The lack of available resources is the main contributor to the lag in development of Arabic emotion recognition in comparison with other languages. Moreover, the emotion taxonomy varies among studies, and the number of emotions to be covered can be extensive, reaching up to 27 categories as in the study [5]. Moreover, these emotions are overlapping and not mutually exclusive (i.e., the occurrence of one emotion in a given text does not imply the non-occurrence of the others). Consequently, a significant amount of labeled data is required for each emotion in order to build a well-performing model.

Pre-trained Language Models (PLMs) and transfer learning can be exploited to overcome the shortage of labeled data for downstream tasks. Indeed, transfer learning has witnessed a major revolution in the last few years, leading to state-of-the-art results in various Natural Language Processing (NLP) tasks. Nevertheless, their effectiveness is constrained by differences between the datasets utilized for pre-training the model and those used for fine-tuning it for the target task [6]. These differences may include the domain, Arabic variety [7], and writing style (i.e., formal or informal) [8]–[10].

Our study seeks to explore the utilization of PLMs and transfer learning for Arabic fine-grained emotion recognition. The paper comprehensively explores the use of various PLMs trained on datasets with different domains (e.g., Wikipedia articles versus social media posts) and different Arabic varieties, including Modern Standard Arabic (MSA)

and different spoken Arabic dialects commonly employed on the Internet [11].

We also propose a new and straightforward method to efficiently improve the performance of the transfer learning-based model for Arabic emotion recognition using semantic information, namely sentiments and named entities extracted from text. We use a simple strategy wherein the semantic information concatenates with the input text before passing it to the model. To this end, we attempt two methods. The first method is to embed the semantic information within the text while the second method utilizes a [SEP] token to differentiate the input text from the added semantic information when feeding the model.

Our proposed method is extensively evaluated using an existing emotion benchmark dataset in the literature [8]. The incorporation of semantic information in text before fine-tuning a model enhances its overall performance, surpassing the performance of transfer learning-based model that does not use semantic information. Not all semantic information contributes equally to the advances in the model's performance. The inclusion of Named Entity (NE) information results in higher model performance compared to the use of only sentiment information. Our proposed method also outperforms the current state-of-the-art method for Arabic emotion recognition, achieving an improvement of 9.80% in weighted-average F1 score. To our knowledge, this study is the first to examine the utilization of NEs to enhance emotion recognition model performance. We can summarize our contributions in this paper as follows:

- We investigate the use of transfer learning for Arabic emotion recognition using different PLMs architectures, data domain, and Arabic varieties.
- We propose a simple method to improve cross-domain cross-dialect transfer learning for Arabic emotion recognition using semantic information.
- We extensively evaluate our proposed method by conducting transfer learning experiments on an existing emotion benchmark to showcase how well our method enhances the transfer learning model regardless of the differences between data domains and Arabic varieties utilized in the pre-training and fine-tuning.

The remainder of this paper is organized as follows. Section 2 presents a background on the research, in terms of the role of sentiment analysis and named entity recognition in Natural Language Processing (NLP). Section 3 reviews existing Arabic emotion recognition approaches. Section 4 presents the proposed methodology in detail and discusses the benchmark dataset. Section 5 outlines the experimental setup. Section 6 reports on the performance results and includes a discussion. Finally, Section 7 concludes the work and suggests future research directions.

2. BACKGROUND

2.1 Sentiment Analysis

Sentiment Analysis (SA) is a well-established field of research in Natural Language Processing (NLP) that aims to automatically identify the sentiment conveyed in a given text, whether it is positive, negative, or neutral.

SA has numerous applications in various sectors, including but not limited to marketing, education, healthcare, and business [12]–[14]. Moreover, SA has been used in other research fields of NLP. For example, SA is employed to detect fake news, as demonstrated by [15], which revealed that fake news headlines tend to have a more negative sentiment than real news headlines. Consequently, the sentiments expressed in news headlines can be a crucial differentiator between real and fake news. Additionally, the combination of topic detection and SA can enable the identification of topics that are highly correlated with positive or negative opinions, thus helping business analysts understand the drivers behind sentiments [16]. SA has also been employed to predict stock market movements, as evidenced by [17]. The study showed that analyzing financial microblogs to identify sentiments and combining them with historical data of the Shanghai Composite Index can be effective in predicting stock market movements.

2.2 Named Entity Recognition

Named Entity Recognition (NER) is an information extraction task that involves identifying and recognizing named entities in text. A named entity is simply a real-world object that can be referred to using a proper name. For example, in the sentence "Apple Inc. was founded on April 1, 1976, by Steve Jobs, Steve Wozniak, and Ronald Wayne as a partnership", a generic named-entity recognizer might identify the organization's name "Apple" and the persons' names "Steve Jobs," "Steve Wozniak," and "Ronald Wayne" [18], [19].

NER plays a crucial role in several NLP applications, including Question Answering systems that take questions as input and return accurate answers. Many questions revolve around NEs such

as "who" (person), "where" (location), "what" (product, organization, etc.). Therefore, if a question asks about an NE, NE recognizer can identify the NEs within the question, facilitating the retrieval of the exact answer from a relevant passage [20]. Additionally, NER is useful for information retrieval, which aims to identify and retrieve relevant documents from a dataset based on a query. Utilizing a high-performing NER model would enhance the quality of retrieving documents for queries containing NEs [21].

3. RELATED WORK

In the initial stages of research on Arabic emotion recognition, lexicons were utilized as a primary means of recognition [22]–[24]. These studies investigated the use of lexicons to detect and classify emotional expressions across various linguistic units, such as individual words, sentences, and entire documents.

Machine Learning (ML) algorithms have played a vital role in NLP tasks including emotion recognition. In particular, [25] utilized Multinomial Naive Bayes (MNB) and Support Vector Machine (SVM) algorithms to implement emotion classifier for the four emotions of anger, sadness, joy, and disgust. The researcher conducted their experiments on a Twitter dataset. Likewise, [8], employed a complementary Naive Bayes algorithm for Arabic emotion recognition, achieving superior performance compared to simple Naive Bayes and Sequential Minimal Optimization classifiers. The researchers conducted their experiments on a Twitter dataset, which they annotated and represented to the ML algorithms as an n-gram Bag of Words (BOW) of stemmed input tweets. The tweets were stemmed using the Arabic Light Stemmer, and the n-grams utilized ranged from one to three grams.

Another study [4], introduced SemEval-2018 Task 1: Affect in Tweets, which included a subtask for multi-label emotion detection in Arabic. One of the participants in the competition, as detailed in [26], achieved the highest ranking by utilizing the SVM algorithm. Indeed, the study examined three different learning algorithms, including SVM, ridge classification, random forests, and an ensemble of the three. The researchers also investigated a range of features, such as sentiment lexicon, n-grams, AraVec [27], and FastText [28]. It was found that the SVM classifier yielded superior results compared to the other models, and the use of AraVec word embeddings produced the highest performance among the examined features. Additionally, Mulki et al. [29] employed Term Frequency-Inverse Document Frequency (TF-IDF) to represent texts in their dataset. In contrast to conducting multi-class emotion recognition, they performed binary classification, utilizing a one-vs-all SVM classifier.

Neural Networks (NN) and deep learning have been employed to identify emotions in Arabic texts. The study conducted by Abdullah and Shaikh [30], participated in the SemEval-2018 competition on multi-label emotion detection and approached the task as a binary classification problem. The study utilized four Dense Neural Networks (DNNs) and normalized the output of the last DNN to either one or zero. Meanwhile, the study conducted by Abdullah et al. [31] utilized Convolutional Neural Networks (CNN), Long Short-Term Memory (LSTM), and fully connected neural network architectures to detect emotions in Twitter contents. This study exploited emojis as a feature in tweets and a set of lexicons containing words for Arabic emotions and Arabic hashtags (MSA and dialects), focusing on four emotions: anger, joy, fear, and sadness. Additionally, the study conducted by [32] proposed three models for identifying emotions in Arabic text: a human-engineered feature-based model, a deep feature-based model, and a hybrid model. The human-engineered feature-based model utilized a feature set that included stylistic, lexical, syntactic, and semantic features, such as POS tags and the use of emotion and sentiment lexicons. The text was represented using TF-IDF, and DNN with a ReLU activation function was trained on TF-IDF features. On the other hand, the deep feature-based model employed four different pre-trained word embeddings, namely emoji2vec [33], AraVec [27], Glove [34], and FastText [28], [35]. They examined the utilization of various LSTM and Gated Recurrent Unit (GRU) deep models. Results showed that the hybrid model, which combined both human-engineered feature-based model and deep feature-based model, exceeded the individual performance of each model.

In recent years, pre-trained language models have emerged as a prominent trend in the field of NLP. These models provide effective transfer learning techniques that can be fine-tuned on limited labeled data to achieve good results for various downstream tasks [6], [36]–[40].

As far as we are aware, only two studies have utilized PLMs and transfer learning for emotion recognition in Arabic text [6], [40]. The first study [6] introduced two PLMs, namely ARBERT and MARBERT, which were developed using massive Arabic datasets. While ARBERT was mainly pre-trained on Modern Standard Arabic (MSA) datasets,

MARBERT was pre-trained on social media data consisting of one billion Arabic tweets. Both PLMs were evaluated on a social meaning task that included emotion recognition, and it was found that fine-tuning MARBERT led to the best results. The second study [40] pre-trained five BERT models, collectively referred to as QARIB, on a comprehensive collection of both MSA and informal texts (Arabic dialects). The preprocessing steps involved splitting off the prefixes and suffixes from words using Farasa tool for segmentation [41], and removing diacritical marks and word elongation. The QARIB PLMs were evaluated on several downstream tasks, including emotion recognition. The study found that combining tweets and formal Arabic when pre-training the models improved the results compared to using tweets alone. They also reported that word segmentation improved named entity recognition but did not lead to substantial improvements in emotion recognition.

4. MATERIAL AND PROPOSED METHOD

In this section, we provide an overview of the benchmark dataset used in the study. Additionally, we introduce our proposed approach for Arabic emotion recognition, which involves integrating semantic information during the fine-tuning of PLMs for the task.

4.1 Dataset

We utilized Arabic Emotions Twitter Dataset (AETD) presented to the research community by [8]. The AETD contains 10,065 tweets, mostly written in the Egyptian dialect, each labeled with one of seven emotions: anger, fear, joy, love, sadness, surprise, sympathy, or none, in case of the absence of emotion in the tweet. Table 1 shows the distribution of tweets in the dataset across the emotion classes. Since the exact training and test partitions of the AETD dataset are not available, a stratified 10-fold cross-validation approach has been used in the experiments. This approach is to ensure that each fold contains approximately the same percentage of samples from each emotion class.

We opted to utilize the AETD dataset, as it is one of the earliest benchmark datasets created and labeled for Arabic emotion recognition. Furthermore, the AETD dataset has been employed in the literature to assess various methods for Arabic emotion recognition, including the current state-of-the-art model. As a result, we can compare our proposed method to those described in literature for the task of Arabic emotion recognition.

Table 1: Distribution of AETD Dataset Across Emotion Classes.

Class	# Tweets
anger	1,444
fear	1,207
joy	1,281
love	1,220
sadness	1,256
surprise	1,045
sympathy	1,062
none	1,550
Total	10,065

4.2 Proposed Method

The methodology involves applying transfer learning techniques to fine-tune Arabic pre-trained language models on the emotion recognition dataset. We will investigate various pre-trained language models that differ not only in the datasets on which they were trained but also in the domains from which the datasets were drawn, the Arabic varieties in which the datasets were written, the model architectures, and whether word segmentation was applied before pre-training or not.

In addition to exploring different pre-trained language models, we propose a simple and efficient method to improve cross-domain cross-dialect transfer learning for emotion recognition. Two types of semantic information are utilized: sentiments and named entities. Specifically, semantic information is extracted from the texts, and then added to the model to enhance its performance in identifying emotions. There are various techniques available to incorporate external information into pre-trained language models [42], [43]. We use a simple strategy where the semantic information is concatenated with the input text before passing it to the model. Two methods are attempted: embedding the semantic information within text and using a [SEP] token to differentiate the input text from the added semantic information. Figure 1 and Figure 2 present the two methods of concatenating semantic information. An ablation study is conducted for both methods to analyze the effect of each semantic knowledge (i.e., NE and sentiments) on the overall performance. Thus, three attempts are made for each method: (a) incorporating only NE, (b) incorporating only sentiment, and (c) incorporating both NEs and sentiments, as illustrated in the figures.

Journal of Theoretical and Applied Information Technology
15th May 2023. Vol.101. No 9

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

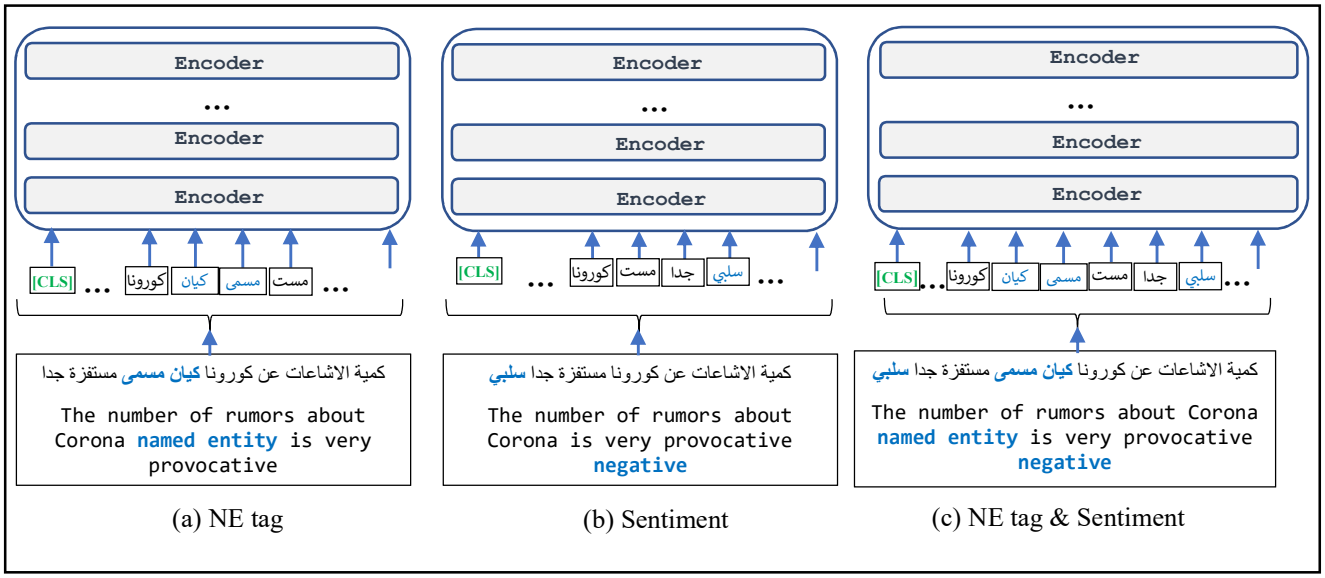

Figure 2: Embedded Semantic Information within Text.

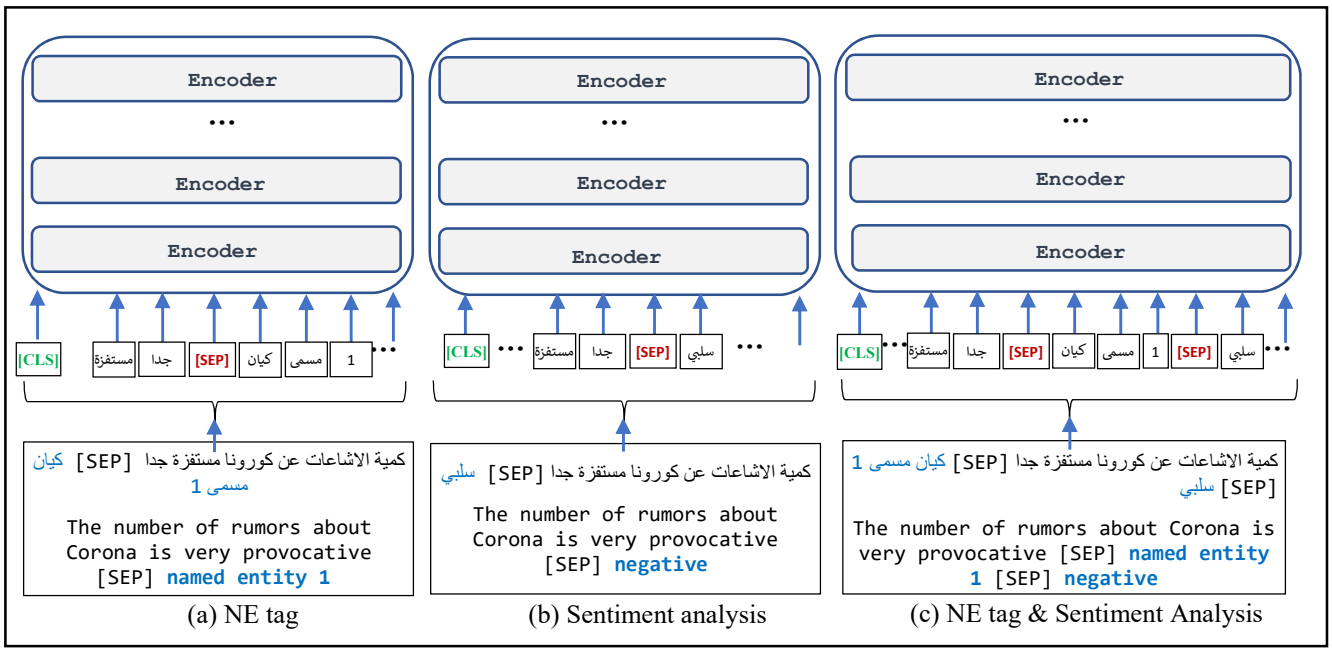

Figure 1: Semantic Information Concatenated as Separate Sentences

When using the [SEP] token to differentiate between input text and added information about named entities in text, the algorithm first extracts NEs from the input texts and then counts their frequencies regardless of their types. Then, the frequencies along with the phrase "named entity" will be added after the [SEP] token as shown in (c) of Figure 2.

5. EXPERIMENTAL SETUP

5.1 Dataset Preprocessing

In order to build our proposed models and make the evaluation comparisons with other techniques, the first step should involve preprocessing the utilized dataset to remove unnecessary characters from the raw text and normalize letters that are usually written interchangeably, which can lead to data sparsity. Preprocessing steps we applied include:

- removing URLs, mentions, retweet and hashtag symbols,
- replacing underscores in hashtag texts with spaces,
- removing all diacritical marks and punctuation,
- removing repeated letters in the words,
- removing English words,
- removing letter elongation in Arabic,
- normalizing different forms of Arabic letters.

We used regular expressions to remove repeated letters, URLs, mentions, retweets, and hashtags. Regarding the normalization of the different forms of letters and the removal of diacritical marks and punctuation, we use the normalizer provided by the AraNLP [44].

5.2 Pre-trained Models

We used AraBERT [36] as a pre-trained language model, which was fine-tuned for Arabic emotion recognition and evaluated with our suggested modification to increase the performance of the model for the target task. The AraBERT is an Arabic pre-trained language model based on Google's BERT architecture [45]. The first released AraBERT v0.1/v1 (original) models were trained on about 23GB of MSA text extracted from Arabic Wikipedia and News articles. The difference between the two versions is that AraBERT v1 pre-segments text using the Farasa segmenter [41] while AraBERT v0.1 does not use any text segmentation. Many versions released later trained on more data with various Arabic varieties and for longer.

We summarized below the versions of AraBERT we used in our experiments.

- **AraBERTv0.2-Twitter-large**
  - Arabic variety of training dataset: Mixture of MSA and Arabic dialects.
  - size of training dataset: 200M MSA sentences + 60M Multi-Dialect Tweets.
  - pre-segmentation: No.
  - Encoder layers: 24.
- **AraBERTv0.2-Twitter-base**
  - Arabic variety of training dataset: Mixture of MSA and Arabic dialects.
  - size of training dataset: 200M MSA sentences + 60M Multi-Dialect Tweets.
  - pre-segmentation: No.
  - Encoder layers: 12.
- **AraBERTv0.2-large**
  - Arabic variety of training dataset: MSA.
  - size of training dataset: 200M MSA sentences.
  - pre-segmentation: No.
  - Encoder layers: 24.
- **AraBERTv2-large**
  - Arabic variety of training dataset: MSA.
  - size of training dataset: 200M MSA sentences.
  - pre-segmentation: Yes.
  - Encoder layers: 24.
- **AraBERTv0.2-base**
  - Arabic variety of training dataset: MSA.
  - size of training dataset: 200M MSA sentences.
  - pre-segmentation: No.
  - Encoder layers: 12.

5.3 Semantic Information Extraction Tools

In order to extract named entities from the text in order to use it in our proposed method, we used a NER model that was built by fine-tuning the CAMeLBERT dialectal Arabic model using the ANERcorp dataset [46], [47]. For each token in an input text, the NER model produces a label for each token that indicates one of the following named entities: 'B-LOC', 'I-LOC', 'B-ORG', 'I-ORG', 'B-PERS', 'I-PERS', 'B-MISC','I-MISC', or 'O'. The LOC is used for location, ORG for organization, PERS for person, and MISC for miscellaneous. The 'B' and 'I' differentiates the beginning (B) and the inside (I) of entities. The O is used for non-entity tokens.

Regarding extracting sentiments from the input text, we used a sentiment analysis model, which was built by fine-tuning the CAMeLBERT dialectal Arabic model [46], [47] to identify the sentiment of each sentence in our experiments. The SA model was fine-tuned using a combination of multiple Arabic datasets for sentiment analysis. It outputs one of three sentiment labels for each sentence: 'positive', 'negative' or 'neutral'.

5.4 Evaluation Metrics

The most common evaluation metrics used for the performances of the text classification models, including those built for emotion recognition, are: *Precision* (*P*), *Recall* (*R*), *F1 score* (*F1*), and *Accuracy*. Given a set of emotion categories *C*, and the built model, we calculated the evaluation metrics of the model's performance on each emotion category *c* as follows:

$$Accuracy_c = \frac{TP_c+TN_c}{TP_c+TN_c+FP_c+FN_c} \quad (1)$$

$$Precision_c = P_c = \frac{TP_c}{TP_c+FP_c} \quad (2)$$

$$Recall_c = R_c = \frac{TP_c}{TP_c+FN_c} \quad (3)$$

Where TP_c is the number of dataset instances that are correctly recognized by the model as c and TN_c is the number of dataset instances that are correctly recognised by the model as *not c*. On the other hand, FP_c is the number of instances that are incorrectly recognized by the model as c and FN_c is the number of instances that are incorrectly recognized by the model as *not c*.

The F1 score is the harmonic mean of precision and recall. Therefore, it symmetrically represents both precision and recall in one metric. The equation of F1 score is as follows:

$$F1\ score = F1_c = 2 * \frac{P_c*R_c}{P_c+R_c} \quad (4)$$

In order to measure the overall performance of the built models in our experiments, we used *Accuracy*, and the weighted average of the three metrics, namely, *Precision*, *Recall*, and *F1 score* as follows:

$$Accuracy = \frac{\sum_{c=1}^{C} TP_c+\sum_{c=1}^{C} TN_c}{\sum_{c=1}^{C} TP_c+\sum_{c=1}^{C} TN_c+\sum_{c=1}^{C} FP_c+\sum_{c=1}^{C} FN_c} \quad (5)$$

The weighted average, on the other hand, takes the proportion for each emotion category (weight) into account when computing the average of the metrics. They are calculated using the following formulas:

$$P_{weighted} = \frac{\sum_{c=1}^{C} P_c*|c|}{\sum_{c=1}^{C} |c|} \quad (6)$$

$$R_{weighted} = \frac{\sum_{c=1}^{C} R_c*|c|}{\sum_{c=1}^{C} |c|} \quad (7)$$

$$F1_{weighted} = \frac{\sum_{c=1}^{C} F1_c*|c|}{\sum_{c=1}^{C} |c|} \quad (8)$$

where $|c|$ indicates the total number of instances in the dataset of emotion category c and $\sum_{c=1}^{C} |c|$ is the sum of the total number of each emotion category's instances in the dataset.

5.4 Hyperparameters Tuning

We run all our experiments using a Tesla T4 GPU. For fine-tuning AraBERT models on emotion recognition task, we used the standard hyper-parameters recommended in AraBERT where the learning rate is equal to 2e-5, the batch size is 16, and the maximum input sequence length is 128. We set the number of epochs to 20. Each round of 20 epochs took around 40 to 50 minutes. As previously mentioned, we used 10-fold cross-validation when conducting our experiments on the AETD dataset. For each fold we have 20 epochs of training and evaluation. The final result is the average of all the evaluation scores of the 10 folds. It took approximately 9 and a half hours.

6. RESULTS AND DISCUSSION

This section extensively investigates the cross-domain cross-dialect transfer learning for emotion recognition. Then, it explores the effects of our proposed approach on the aforementioned issue. Finally, we evaluate the efficacy of our proposed method for Arabic emotion recognition by comparing its performance with existing techniques in the literature, including the state-of-the-art approach, using the benchmark dataset (AETD). We also report the performance results and discuss our insights.

6.1 Cross-domain Cross-dialectal Transfer Learning for Arabic Emotion Recognition

We used five different Arabic pre-trained language models: AraBERTv0.2-Twitter-large, AraBERTv0.2-Twitter-base, AraBERTv0.2-large, AraBERTv2-large, and AraBERTv0.2-base. Two of the aforementioned PLMs were pre-trained on multi-dialect tweets and MSA corpora, while the remaining were pre-trained using only MSA corpora written in formal styles such as Wikipedia and news articles. These PLMs also differ in terms of the complexity of the BERT model used in the pre-training, as some of them are (*large*) models with 24 encoder layers and others are (*base*) models with 12 encoders. In our experiments, we also examined the use of segmentation during the pre-training phase and investigated its effects on the overall performance of the model when fine-tuned for Arabic emotion recognition. To this end, we used the AraBERTv2-large model, which was trained after segmenting the text using Farasa tool [41] As explained in detail in Section 5.2. It is worth mentioning that the Farasa tool was developed using MSA text. Table 2 presents the results of the aforementioned five pre-trained models on the dataset.

The PLMs which pre-trained on a mixture of dialectal Arabic and MSA performed better than those pre-trained only on MSA data when fine-tuned for Arabic emotion recognition. The use of AraBEERTv0.2-Twitter-large model (pre-trained on multi-dialect tweets and MSA data) outperformed the AraBERTv0.2-large (pre-trained only on MSA data), achieving an improvement of 2.17% in the weighted-average F1 score.

Table 2: Emotion Recognition Results of Different PLMs when Fine-tuning for Arabic Emotion Recognition on AETD Dataset.

Model	Accuracy	Weighted-average		
		F1	P	R
AraBERTv0.2-Twitter-large	78.93	**78.93**	78.97	78.97
AraBERTv0.2-Twitter-base	77.99	78.06	78.23	77.99
AraBERTv0.2-large	76.90	**76.76**	76.80	76.80
AraBERTv0.2-base	75.35	75.40	75.80	75.57
AraBERTv2-large	55.39	51.12	50.37	52.50

Notably, the results also suggest that using segmentation during pre-training does not improve the performance of a model when fine-tuning it for Arabic emotion recognition on dialectal Arabic. Despite both AraBERTv2-large and AraBERTv0.2-large models being pre-trained on MSA data from the same domain and having a similar formal writing style, the former, which uses segmentation, showed a decline in performance compared to the latter, which does not use segmentation. The weighted-average F1 score, precision, and recall of AraBERTv2-large dropped by 25.64%, 26.43%, and 24.30%, respectively. The negative impact of segmentation on AraBERTv2-large when fine-tuned for dialectal Arabic emotion recognition can be partly attributed to the Farasa segmenter's development based on MSA data, while the fine-tuning data used dialectal Arabic collected from online content. Applying the Farasa segmenter to a dialectal dataset before fine-tuning may result in incorrect outputs, given the differences between dialectal Arabic and MSA in syntax, methodology, and vocabulary.

We observed that larger models with 24 encoder layers outperformed base models with 12 encoder layers, regardless of the data's domain or Arabic variety used in pre-training and fine-tuning phases. For example, AraBERTv0.2-large (pre-trained on MSA data with 24 encoders) performed better than AraBERTv0.2-base (pre-trained on MSA with 12 encoders), achieving a weighted-average F1 score of 76.76%, with an improvement of 1.36%. This improvement can be attributed to the fact that increasing the number of encoder layers also increases the number of parameters (weights) and attention heads, resulting in a more complex model that better represents the complexity and richness of Arabic language morphology. The same conclusion can be drawn from the results of AraBERTv0.2-Twitter-large (pre-trained on dialectal Arabic and MSA data with 24 encoders) and AraBERTv0.2-Twitter-base (pre-trained on dialectal Arabic and MSA data with 12 encoders), where AraBERTv0.2-Twitter-large outperformed AraBERTv0.2-Twitter-base, achieving a higher weighted-average F1 score by 0.81%.

6.2 Influence of Proposed Method

In this section, we present the results of our investigation into the extent to which our proposed method enhances cross-domain and cross-dialect transfer learning for Arabic emotion recognition. We evaluated the impact of our method in two scenarios. Firstly, we considered when the pre-training domain and Arabic variety were relatively similar to those used in fine-tuning. For this, we chose AraBERTv0.2-Twitter-large, which produced one of the best results for Arabic emotion recognition when fine-tuned on dialectal Arabic data. Secondly, we investigated our proposed method when the pre-training domain and Arabic variety differed from those used in fine-tuning. For this, we selected AraBERTv0.2-large, which yielded the best results when fine-tuned on dialectal Arabic for emotion recognition despite being pre-trained on MSA data (as shown in Table 2).

We examined the performance of the models when applying transfer learning and incorporating the semantic information. Both methods of incorporating semantic information were investigated: (a) embedding the semantic information within the input text and (b) using the [SEP] to differentiate between the input text and the added information. Table 3 presents the comparison results between the various models.

When we applied transfer learning to our model, we referred to the version without our semantic information approach as the "baseline" model. Our experiments showed that adding named entity tags to the text before fine-tuning the model yielded the best performance for Arabic emotion recognition on the AETD dataset. Specifically, the inclusion of NE tags resulted in a 1.34% improvement in the weighted-average F1 score for the baseline model using the AraBERTv0.2-large model, and a 2.67% improvement in the weighted-average F1 score for the baseline using the AraBERTv0.2-Twitter-large model.

Additionally, incorporating the sentiment analysis within the text prompts an improvement in the baseline's model performance, albeit a lesser increase than the impact of adding the NE information. The use of AraBERTv0.2-large produced a baseline with a weighted-average F1 score of 76.76%. Incorporating only sentiment

analysis information within the text input increased the baseline's weighted-average F1 score to 77.12%, while incorporating only NE information increases the F1 score to 78.10%. Also, the sentiment information resulted in an improvement of around 0.44% in the AraBERTv0.2-Twitter-large model's performance (weighted-average F1 score).

Table 3: Emotion Recognition Results of Our Proposed Methods on AETD set. "NE" indicates named entity. "SA" indicates sentiment analysis. "embedded" indicates within text. "[SEP]" indicates using [SEP] token. "ACC" indicates Accuracy.

Model		ACC	Weighted-average		
			F1	P	R
AraBERT v0.2-Twitter-large	Baseline	78.93	78.93	78.97	78.97
	+NE (embedded)	**81.73**	**81.60**	**81.92**	**81.40**
	+SA (embedded)	79.48	79.37	79.70	79.13
	+NE&SA (embedded)	80.43	80.41	80.49	80.49
	+NE [SEP]	80.68	80.52	81.01	80.18
	+SA [SEP]	79.38	79.26	79.54	79.00
	+NE&SA [SEP]	80.28	80.14	80.55	79.87
AraBERT v0.2-large	Baseline	76.90	76.76	76.80	76.80
	+NE (embedded)	**78.19**	**78.10**	**78.13**	**78.21**
	+SA (embedded)	77.24	77.12	77.20	77.21
	+NE&SA (embedded)	77.29	77.16	77.25	77.30
	+NE [SEP]	78.05	78.03	76.28	76.15
	+SA [SEP]	77.24	77.18	77.21	77.38
	+NE&SA [SEP]	77.30	77.17	77.29	77.34

For all pre-trained models, the use of both NEs and sentiments as appending features along with the textual content of the input yielded a better F1 score than the baseline's F1 score, but at the same time lesser than the F1 score that resulted from adding only NEs information to the input text. For example, when employing the AraBERTv0.2-large model on the AETD dataset in the fine-tuning phase, the use of both sentiments and NEs information produced a weighted-average F1 score of 77.16%. On the other hand, using only NE information as additional features within text input results in a better weighted-average F1 score of 78.10%, while the baseline's weighted-average F1 score is 76.76%. According to the aforementioned experiments and results, the semantic information, such as the sentiment of a given text (i.e., positive, negative, or neutral) and the existing named entities help to increase the overall performance of the emotion recognition's model.

In order to understand the positive impacts of adding sentiments and NEs information as features within text input on the overall performance of the transfer learning-based model for emotion recognition, a preliminary analysis was conducted on the AETD dataset where we extracted named entities from the samples. To this end, we used a NER model and a sentiment analyzer based on CAMeLBERT dialectal Arabic model [47]. Then, we extracted the named entities from the AETD dataset and analysed the distribution of the named entity frequencies across emotion categories. We found that the samples of emotions like *anger, joy, surprise, sympathy,* and *sadness* contain more named entities than samples of other emotions, such as *love* and *fear*. These findings confirm that some emotion categories are explicitly expressed towards something while other emotion categories are more of an intrinsic feeling [5]. The set of emotion categories that are directed explicitly towards a subject/object and those considered intrinsic feelings may slightly differ from one dataset to another, depending on the domain of the dataset and the topics discussed in the dataset. Our findings indicate that including the NE information as features within input text may help the emotion recognition model to distinguish between various emotion categories. Figure 3 presents the distribution of the number of named entities across various emotion categories for the AETD dataset.

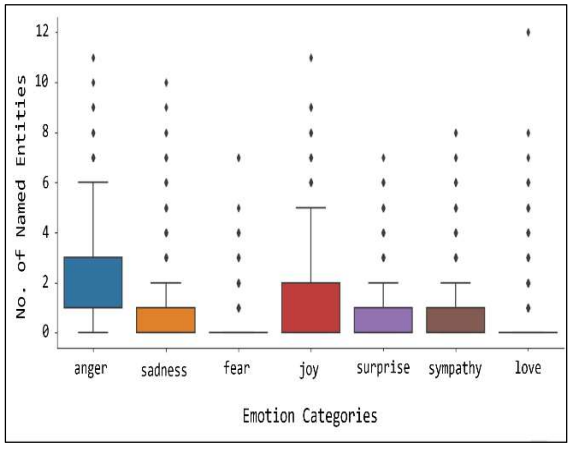

Figure 3: Distribution of Named Entity Frequencies Across Emotion Categories in AETD Dataset.

We expected that incorporating sentiments with transfer learning would result in a great positive impact on emotion recognition. The results, however, only showed a slight improvement in the model's performance. Analyzing the distribution of sentiments across emotion categories for AETD dataset revealed that even for positive emotions, like *joy*, some sentences were tagged by the sentiment analyzer as *negative* or *neutral*. Of the 1,281 sentence samples of the *joy* emotion in the AETD dataset, 12.02% were considered *negative* while 10.30% of the samples were considered *neutral*. The *fear* emotion is an example of negative sentiment. Surprisingly, 10.59% of the sentence samples annotated as *fear* emotion in the AETD dataset were considered *positive* by the sentiment analyzer. This may explain the slight improvement that resulted from incorporating sentiment information when fine-tuning the model. Figure 4 shows the distribution of the sentiments across various emotion categories for the AETD dataset.

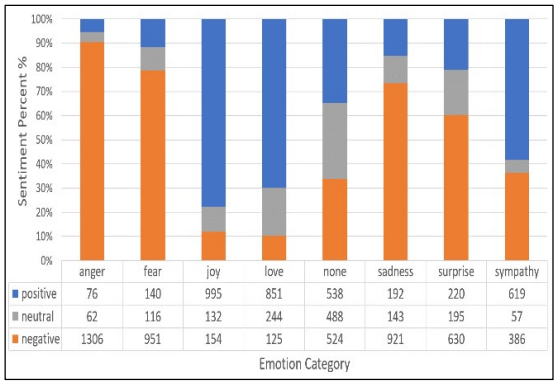

Figure 4: Distribution of Sentiments Across Emotion Categories in AETD Dataset.

The use of [SEP] token in order to separate between text input and external knowledge of semantic information had a positive impact on the transfer learning-based model for emotion recognition but led to less improvement than incorporating the semantic information within text. We observed that the AraBERTv0.2-Twitter-large model's weighted-average F1 score decreased from 81.60% (when embedded named entity information within text) to 80.52% (when adding named entity information using [SEP] token). This can be partially attributed to the fact that the utilized AraBERT models had not been trained using the [SEP] token in the pre-training phase. The [SEP] token usually has to be inserted at the end of a single input when the given task requires two inputs, such as Natural Language Inference (NLI).

6.3 Performance Comparison

As our best performing proposed method is the one that incorporates NE information within input text by way of transfer learning using the AraBERTv0.2-Twitter-large model, we compared its performance with the methods in literature for emotion recognition, including the state-of-the-art study [32]. The state-of-the-art method for emotion recognition, in literature, employed a hybrid model of a human-engineered feature-based (HF) model and deep feature-based (DF) model. In the human-engineered feature-based model, the feature set included stylistic, lexical, syntactic, and semantic features as well as the use of emotion and sentiment lexicons. The text was represented using TF-IDF. In the deep feature-based model, the word embedding was utilized (see Section 3 for more details). Table TABLE 4 juxtaposes the comparison results of our proposed model with various emotion recognition models [8] including the state-of-the-art one [32] when using the AETD dataset to test them.

Table 4: Comparison Results of Our Proposed Method with Emotion Recognition Models including the State-of-the-art model when evaluated on AETD Dataset.

Model	ACC	Weighted-average		
		F1	P	R
Our Proposed Method	81.73	81.60	81.92	81.40
HF+DF [32]	71.80	71.80	72.20	71.80
Complement Naïve Bayes [8]	68.12	65.80	68.80	68.10
Sequential Minimal Optimization [8]	63.43	63.70	64.30	63.40

Our proposed method for emotion recognition performed better than the state-of-the-art (DF+HF) model, achieving an improvement of 9.80%, 9.72%, and 9.60% in weighted-average F1 score, precision, and recall respectively. These results prove the positive effect of employing semantic information, such as named entities and sentiments, in the emotion recognition task. The results also shed light on the considerable influence of large language models in emotion recognition. Our proposed method showed that combining PLMs with semantic information helped to surpass the state-of-the-art model in the literature, which relied on deep learning and human-engineered features.

7. CONCLUSION

Arabic emotion recognition lacks the annotated datasets which are not only necessary for supervised learning, but also for transfer learning when fine-

tuning pre-trained language models for downstream tasks like emotion recognition.

We conducted transfer learning experiments by exploring the use of pre-trained language models for Arabic emotion recognition using various PLM architectures (i.e., various number of encoder layers), data domains, and Arabic varieties. We found out that the model with a large number of encoder layers impacted the overall performance of the transfer learning-based models in Arabic. More encoder layers resulted in better performance. Also, the PLMs trained on a mixture of dialectal Arabic and MSA performed better than other PLMs trained only on MSA when fine-tuning them on Arabic emotion recognition datasets.

The paper also presented a novel approach to enhance cross-domain cross-dialect transfer learning for Arabic emotion recognition by leveraging semantic information, including sentiments and named entities present in the text. The proposed method was thoroughly evaluated on the existing benchmark (AETD) dataset, and the results showed that incorporating semantic information in the text before fine-tuning the model improved the performance of the transfer learning-based model, regardless of the differences or similarities between the pre-training and fine-tuning data. However, not all semantic information had the same contribution to the model's performance improvement. In particular, incorporating only named entities information yielded better results than incorporating only sentiment information. The proposed method outperformed the state-of-the-art method in literature, achieving a 9.80% improvement in the weighted-average F1 score for Arabic emotion recognition.

For future work, we plan to investigate techniques that address the shortage of labeled data, such as zero-shot and few-shot learning. We also plan to explore various techniques for word embeddings and analyze their relationship with the overall accuracy of the model, particularly for low-resource annotated data in tasks such as Arabic emotion recognition.

REFERENCES

[1] A. Mefiah, Y. A. Alotaibi, and S.-A. Selouani, "Arabic speaker emotion classification using rhythm metrics and neural networks," in *2015 23rd European Signal Processing Conference (EUSIPCO)*, 2015, pp. 1426–1430.

[2] S. Klaylat, Z. Osman, L. Hamandi, and R. Zantout, "Emotion recognition in Arabic speech," *Analog Integr Circuits Signal Process*, vol. 96, no. 2, 2018, pp. 337–351.

[3] M. Abdul-Mageed, H. AlHuzli, and M. D. DuaaAbu Elhija, "Dina: A multi-dialect dataset for arabic emotion analysis," in *The 2nd workshop on Arabic corpora and processing tools*, 2016, p. 29.

[4] S. Mohammad, F. Bravo-Marquez, M. Salameh, and S. Kiritchenko, "Semeval-2018 task 1: Affect in tweets," in *Proceedings of the 12th international workshop on semantic evaluation*, 2018, pp. 1–17.

[5] D. Demszky, D. Movshovitz-Attias, J. Ko, A. Cowen, G. Nemade, and S. Ravi, "GoEmotions: A Dataset of Fine-Grained Emotions," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020, pp. 4040–4054.

[6] M. Abdul-Mageed, A. Elmadany, and others, "ARBERT & MARBERT: Deep Bidirectional Transformers for Arabic," in *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 2021, pp. 7088–7105.

[7] M. J. Althobaiti, "Automatic Arabic dialect identification systems for written texts: a survey," *arXiv preprint arXiv:2009.12622*, 2020.

[8] A. Al-Khatib and S. R. El-Beltagy, "Emotional tone detection in arabic tweets," in *International Conference on Computational Linguistics and Intelligent Text Processing*, 2017, pp. 105–114.

[9] H. Alhuzali, M. Abdul-Mageed, and L. Ungar, "Enabling deep learning of emotion with first-person seed expressions," in *Proceedings of the Second Workshop on Computational Modeling of People's Opinions, Personality, and Emotions in Social Media*, 2018, pp. 25–35.

[10] A. J. Almahdawi and W. J. Teahan, "A new arabic dataset for emotion recognition," in *Intelligent Computing-Proceedings of the Computing Conference*, 2019, pp. 200–216.

[11] M. J. Althobaiti, "Creation of annotated country-level dialectal Arabic resources: An unsupervised approach," *Nat Lang Eng*, vol. 28, no. 5, 2022, pp. 607–648.

[12] R. Feldman, "Techniques and applications for sentiment analysis," *Commun ACM*, vol. 56, no. 4, 2013, pp. 82–89.

[13] P. Adinolfi, E. D'Avanzo, M. D. Lytras, I. Novo-Corti, and J. Picatoste, "Sentiment analysis to evaluate teaching performance," *International Journal of Knowledge Society Research (IJKSR)*, vol. 7, no. 4, 2016, pp. 86–107.

[14] J. Qiu, C. Liu, Y. Li, and Z. Lin, "Leveraging sentiment analysis at the aspects level to predict ratings of reviews," *Inf Sci (N Y)*, vol. 451, 2018, pp. 295–309.

[15] J. Paschen, "Investigating the emotional appeal of fake news using artificial intelligence and human contributions," *Journal of Product & Brand Management*, vol. 29, no. 2, 2020, pp. 223–233.

[16] K. Cai, S. Spangler, Y. Chen, and L. Zhang, "Leveraging sentiment analysis for topic detection," *Web Intelligence and Agent Systems: An International Journal*, vol. 8, no. 3, 2010, pp. 291–302.

[17] B. Zhao, Y. He, C. Yuan, and Y. Huang, "Stock market prediction exploiting microblog sentiment analysis," in *2016 International Joint Conference on Neural Networks (IJCNN)*, 2016, pp. 4482–4488.

[18] D. Jurafsky and J. H. Martin, *Speech & language processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*. Pearson Education International, 2009.

[19] M. Althobaiti, U. Kruschwitz, and M. Poesio, "Combining minimally-supervised methods for Arabic named entity recognition," *Trans Assoc Comput Linguist*, vol. 3, 2015, pp. 243–255.

[20] R. SRIHARI and W. Li, "Information extraction supported question answering," in *Proceedings of the 8th Text Retrieval Conference (TREC-8)*, 1999, pp. 185–196.

[21] J. Guo, G. Xu, X. Cheng, and H. Li, "Named entity recognition in query," in *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2009, pp. 267–274.

[22] A. M. Abd Al-Aziz, M. Gheith, and A. S. Eldin, "Lexicon based and multi-criteria decision making (MCDM) approach for detecting emotions from Arabic microblog text," in *2015 First International Conference on Arabic Computational Linguistics (ACLing)*, 2015, pp. 100–105.

[23] M. Al-A'abed and M. Al-Ayyoub, "A lexicon-based approach for emotion analysis of arabic social media content," in *The International Computer Sciences and Informatics Conference (ICSIC)*, 2016, pp. 343–351.

[24] A. F. El Gohary, T. I. Sultan, M. A. Hana, and M. M. El Dosoky, "A computational approach for analyzing and detecting emotions in Arabic text," *International Journal of Engineering Research and Applications (IJERA)*, vol. 3, no. 3, 2013, pp. 100–107.

[25] W. A. Hussien, Y. M. Tashtoush, M. Al-Ayyoub, and M. N. Al-Kabi, "Are emoticons good enough to train emotion classifiers of arabic tweets?," in *2016 7th International Conference on Computer Science and Information Technology (CSIT)*, 2016, pp. 1–6.

[26] 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 *Proceedings of The 12th International Workshop on Semantic Evaluation*, 2018, pp. 236–244.

[27] 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, 2017, pp. 256–265.

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

[29] H. Mulki, C. B. Ali, H. Haddad, and I. Babaoğlu, "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, pp. 167–171.

[30] M. Abdullah and S. Shaikh, "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, pp. 350–357.

[31] M. Abdullah, M. Hadzikadicy, and S. Shaikhz, "SEDAT: sentiment and emotion detection in Arabic text using CNN-LSTM deep learning," in *2018 17th IEEE international conference on machine learning and applications (ICMLA)*, 2018, pp. 835–840.

[32] N. Alswaidan and M. E. B. Menai, "Hybrid feature model for emotion recognition in Arabic text," *IEEE Access*, vol. 8, 2020, pp. 37843–37854.

[33] B. Eisner, T. Rocktäschel, I. Augenstein, M. Bošnjak, and S. Riedel, "emoji2vec: Learning emoji representations from their description," *arXiv preprint arXiv:1609.08359*, 2016.

[34] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 2014, pp. 1532–1543.

[35] É. Grave, P. Bojanowski, P. Gupta, A. Joulin, and T. Mikolov, "Learning Word Vectors for 157 Languages," in *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, 2018.

[36] W. Antoun, F. Baly, and H. Hajj, "AraBERT: Transformer-based Model for Arabic Language Understanding," in *Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, with a Shared Task on Offensive Language Detection*, 2020, pp. 9–15.

[37] A. Safaya, M. Abdullatif, and D. Yuret, "Kuisail at semeval-2020 task 12: Bert-cnn for offensive speech identification in social media," in *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, 2020, pp. 2054–2059.

[38] W. Antoun, F. Baly, and H. Hajj, "AraELECTRA: Pre-Training Text Discriminators for Arabic Language Understanding," in *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, 2021, pp. 191–195.

[39] W. Antoun, F. Baly, and H. Hajj, "AraGPT2: Pre-Trained Transformer for Arabic Language Generation," in *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, 2021, pp. 196–207.

[40] A. Abdelali, S. Hassan, H. Mubarak, K. Darwish, and Y. Samih, "Pre-training bert on arabic tweets: Practical considerations," *arXiv preprint arXiv:2102.10684*, 2021.

[41] A. Abdelali, K. Darwish, N. Durrani, and H. Mubarak, "Farasa: A fast and furious segmenter for arabic," in *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: Demonstrations*, 2016, pp. 11–16.

[42] R. Wang, D. Tang, N. Duan, Z. Wei, X.-J. Huang, J. Ji, G. Cao, D. Jiang, and M. Zhou, "K-Adapter: Infusing Knowledge into Pre-Trained Models with Adapters," in *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, 2021, pp. 1405–1418.

[43] Q. Liu, D. Yogatama, and P. Blunsom, "Relational Memory-Augmented Language Models," *Trans Assoc Comput Linguist*, vol. 10, 2022, pp. 555–572.

[44] M. Althobaiti, U. Kruschwitz, and M. Poesio, "AraNLP: a Java-based Library for the Processing of Arabic Text," in *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, 2014, pp. 4134–4138.

[45] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in *Proceedings of NAACL-HLT*, 2019, pp. 4171–4186.

[46] G. Inoue, B. Alhafni, N. Baimukan, H. Bouamor, and N. Habash, "The Interplay of Variant, Size, and Task Type in Arabic Pre-trained Language Models," in *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, 2021.

[47] O. Obeid, N. Zalmout, S. Khalifa, D. Taji, M. Oudah, B. Alhafni, G. Inoue, F. Eryani, A. Erdmann, and N. Habash, "CAMeL tools: An open source python toolkit for Arabic natural language processing," in *Proceedings of the 12th language resources and evaluation conference*, 2020, pp. 7022–7032.