IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received March 3, 2020, accepted March 23, 2020, date of publication April 2, 2020, date of current version April 21, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2985228

Identifying Emotion Labels From Psychiatric Social Texts Using a Bi-Directional LSTM-CNN Model

JHENG-LONG WU[1], YUANYE HE[2,3], LIANG-CHIH YU[4], AND K. ROBERT LAI[2]

[1]School of Big Data Management, Soochow University, Taipei City 11102, Taiwan
[2]Department of Computer Science and Engineering, Yuan Ze University, Taoyuan City 32003, Taiwan
[3]Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100864, China
[4]Department of Information Management, Yuan Ze University, Taoyuan City 32003, Taiwan

Corresponding author: Liang-Chih Yu (lcyu@saturn.yzu.edu.tw)

This work was supported by the Ministry of Science and Technology (MOST), Taiwan, under Grant MOST 107-2628-E-155-002-MY3 and Grant MOST 107-2218-E-031 -002 -MY2.

ABSTRACT Discussion features in online communities can be effectively used to diagnose depression and allow other users or experts to provide self-help resources to those in need. Automatic emotion identification models can quickly and effectively highlight indicators of emotional stress in the text of such discussions. Such communities also provide patients with important knowledge to help better understand their condition. This study proposes a deep learning framework combining word embeddings, bi-directional long short-term memory (Bi-LSTM), and convolutional neural networks (CNN) to identify emotion labels from psychiatric social texts. The Bi-LSTM is a powerful mechanism for extracting features from sequential data in which a sentence consists of multiple words in a particular sequence. CNN is another powerful feature extractor which can convolute many blocks to capture important features. Our proposed deep learning framework also applies word representation techniques to represent semantic relationships between words. The paper thus combines two powerful feature extraction methods with word embedding to automatically identify indicators of emotional stress. Experimental results show that our proposed framework outperformed other models using traditional feature extraction such as bag-of-words (BOW), latent semantic analysis (LSA), independent component analysis (ICA), and LSA+ICA.

INDEX TERMS Multiple emotion labeling, deep learning, bi-directional recurrent neural network, long short-term memory neural network, convolutional neural network.

I. INTRODUCTION

Rather than seek professional help, people suffering from mental illness or emotional strain often turn to online communities in search of advice or a sense of human intimacy and understanding. In recent years, many online services have been developed to provide such people with a means for identifying and understanding the issues they face, and for finding helpful resources. Sufferers interact with these services through written texts and comments about their feelings, and qualified therapists who monitor these services then provide replies and appropriate suggestions.

However, such services suffer from an imbalance between ''clients'' and ''providers''. Combined with the asynchronous nature of such communication, clients may wait for considerable lengths of time between replies, which not only reduces the potential benefit of engaging with the service but can also increase client anxiety. Excessive response delay increases the potential of self-harm or other negative behavior on the part of the client. A system that automatically parses client comments to identify particular emotions and their respective severity would allow providers to quickly identify clients in crisis, allowing them to prioritize responses, and thus, potentially avert undesirable outcomes. Such a system could also help provide a macro view of the relative prevalence of various emotional and psychological issues among service clients [1].

Healthcare-oriented web-based services draw many text-based queries related to depression. Psychiatrists and therapists reviewing and replying to these queries label them appropriately to represent the particular type of depression indicated. This paper seeks to automatically label such texts with appropriate emotion labels [2], thus reducing therapist workload and response latency. Table 1 shows an example

The associate editor coordinating the review of this manuscript and approving it for publication was Jenny Mahoney.

TABLE 1. Example of multiple emotion labels in a psychiatric social text query.

Query	Label
I broke up with my dear but cruel boyfriend recently. Since then, I have often felt like crying out of nowhere, and I feel pain every day. Also, it takes me a long time to fall asleep at night. So, I think that continuing to live like this is meaningless.	<Depression>; <insomnia>; <suicide>

text annotated with three emotion labels [3], depression, insomnia, and suicide. This example shows the difficulty of detecting and labeling multiple emotions in the field of sentiment analysis.

This paper uses deep learning models as neural networks (NNs) to resolve the issue of multiple label classification. Word embeddings, convolutional neural networks (CNN), and long short-term memory (LSTM) NNs are used to build a powerful classifier to identify emotion labels. These models have been successfully applied in a wide range of categorization tasks and are used here to develop a powerful classifying mechanism for multiple emotion labels within psychiatric social texts based on the classification performance of two factors: word embeddings and neural network architectures.

This paper has three main contributions. First, this study is the first work to use NN-based methods to address multiple emotion classification for psychiatric social texts, which can reduce the workload and response latency of psychiatrists and therapists. Second, the proposed framework outperforms other models using traditional feature extraction such as bag-of-words (BOW), latent semantic analysis (LSA), independent component analysis (ICA), and LSA+ICA. Third, different NN models are evaluated, providing a baseline result to facilitate future research on emotion classification for psychiatric social texts.

The remainder of this paper is organized as follows. Section II reviews the relevant literature. Section III describes the proposed model for multiple emotion classification. Section IV explains the generation of the psychiatric social text dataset and summarizes the experimental results. Conclusions and directions for future work are presented in Section V.

II. LITERATURE REVIEW

A. EMOTION LABELING

Natural language processing and text mining techniques have been widely applied to analyzing the emotional content of text. Yu *et al.* [4] built Chinese affective resources for two dimensions of emotion labeling, valence and arousal in their dataset. Therefore, they not only consider the valence of positive/negative but also obtain arousal of high/low. Wu *et al.* [5] proposed a categorization approach to filter significant association language features, capturing important discrimination information and, incrementally improving feature precision by removing noise. Multiple labeling can be divided into two methods, problem transformation and algorithm adaptation [6], [7]. In the former, the problem is transformed into multiple classification problems (e.g., binary problems) which are then solved using many classifiers [8]–[10]. The latter modifies the existing algorithm to a new classifier that better fits the current dataset. However, unlike problem transformation, algorithm adaptation can avoid information loss when transforming multiple classification problems into multiple binary classifier problems [11]–[14].

Cui *et al.* [15] addressed problem-insensitivity to the order of n-grams. Their model used distributed semantic features of part-of-speech (POS) sequences to improve the quality of sentiment analysis. Fattach [16] compared different term weighting schemes with a combination of multiple classifiers for sentiment analysis, with results that outperformed state-of-the-art term weighting schemes. Xiong *et al.* [17] used the word-pair sentiment-topic model for the review of short texts. They assumed that all words in a sentence are related to the same topic. Zhou *et al.* [18] used the stack bi-directional long short-term memory (Bi-LSTM) model for sentiment analysis of Chinese microblogs, which also use a word2vec model to capture the semantic features of words. Their experimental result showed that the stack models outperform single-layer LSTM. Xu *et al.* [19] proposed that the extended sentiment dictionary with a naive Bayesian field classifier improved accuracy in review sentiment classification on two datasets on online retail and travel websites. Xia *et al.* [20] used the conditional random field algorithm to extract features from review texts, and a support vector machine was used to classify the sentiment classification of the review. Almeida *et al.* [21] used an ensemble of methods to identify multiple sentiment classifications to explore the wide range of multi-label solutions, outperforming other traditional algorithms on two real datasets. However, the problem of multiple emotion labeling is another major issue in the fields of NLP and text mining.

B. WORD EMBEDDINGS

Many studies have explored the use of word presentation approaches to capture textual semantics and syntax. The word representation approach is based on knowledge repositories such as WordNet, a lexical database annotated by linguists [22]. However, this approach requires laborious manual annotation of word relations, and results are subject to annotator subjectivity, complicating the automatic computation of word similarity [23]. One traditional word representation is the corpus-based one-hot representation, also called the bag-of-words approach, in which each word is represented by a vector [24]. The word co-occurrence matrix is another representation approach that measures the semantic relation of the context between a range of words in a sentence with a given window size. Rohde *et al.* [25] used the singular value decomposition (SVD) approach to reduce the dimension size for a co-occurrence matrix, but new words cannot be added into SVD.

In past years, a distributed word representation learning method, known as word embeddings [26]–[31] has been developed to represent words as low-dimensional dense vectors of real numbers using neural network architectures. Word embeddings can efficiently capture semantic and syntactic contextual information from very large datasets. The neural network language model (NNLM) is a pioneering work that learns word embeddings based on word contexts [26]. Word2vec [27], [28] is a popular method that uses a simple single-layer neural network architecture to learn word embeddings which aim to predict target words based on contexts. GloVe [29] is another word embedding learning method that GloVe constructs word embeddings using overall statistics to probe the underlying co-occurrence statistics of the corpus. In addition to the above general-purpose word embeddings, some researchers have suggested *retrofitting* the pre-trained word vectors using additional knowledge resources to enhance specific downstream applications [30], [31].

Recently, the attention mechanism has used in language modeling approaches such as BERT [32] and GPT-2 [33] for word representation. BERT and GPT-2 are transformer-based models [34] in which BERT only uses the encoder of transformer, and GPT-2 uses the decoder of transformer. In the transformer, both encoder and decoder use the self-attention layer to learn the attention weights for all hidden status. The attention mechanism is a powerful learning approach in neural network models for many NLP tasks.

C. DEEP LEARNING

Deep learning (DL) frameworks provide powerful learning mechanisms for many types of research and applications. Deep NN is multiple feedforward NNs and generally use the approximation theorem to approximate any complex continuous function given enough neurons to build a NN model [35]–[41]. Neural networks have been used to solve many NLP tasks [42]. Recurrent neural networks (RNN) have been used with good effect on text-based data. A simple RNN is a short-term memory using backpropagation through time for model optimization through backward computation, and cannot be applied to long data sequences. RNN can use two advanced cells to capture long-term memory: long short-term memory (LSTM) and gated recurrent units (GRU). LSTM is an extension of RNN and has achieved excellent performance in various tasks, especially long sequential problems. It uses three gates to extract hidden features through time, including a forget gate, an input gate, and an output gate. If the current word is different from the previous word, the information will be forgotten by the forget gate. The input gate is used to determine the information of the current word to be output. The output gate is used to determine the information of the current word to be output. Therefore, these three gates allow data to enter, exit or delete through a forward loop process and avoid vanishing gradients, especially in long sequential texts. Rao *et al.* [43] proposed an LSTM model with sentence representations to build a document sentiment classification model. Their proposed model outperformed other state-of-the-art models on three publicly available document-level review datasets. Wang *et al.* [44] proposed a stacked residual LSTM model to predict sentiment intensity for a given text. According to their experimental results, LSTM with additional stack layers can successfully obtain high classification performance.

Many studies have used bi-directional LSTM to extract effective features. Xie *et al.* [45] used Bi-LSTM as a classifier to extract variable feature length, finding that Bi-LSTM significantly outperformed the INTERSPEECH 2010 features on the CASIA database. He *et al.* [49] proposed two implementations of LSTM for review data of Arabic Hotels: a character-level bi-directional LSTM along with a conditional random field classifier (Bi-LSTM-CRF), and an aspect-based LSTM considered as attention expressions. GRU has also been widely applied. Li *et al.* [47] proposed a bi-directional gated recurrent unit NN model (BiGRULA) for sentiment analysis for tourism review sentiment classification. They used a topic model (lda2vec) and an attention mechanism in their BiGRULA model, where lda2vec is used to discover all the main topics in a review corpus. Tian *et al.* [48] proposed an attention aware bi-GRU-based framework for sentiment analysis, using bi-GRU to account for complicated interaction and obtain the weight of keywords for sentiment apprehension.

Using a more advanced model, many researchers use recurrent neural networks with convolutional neural networks to solve many tasks [49]–[51]. The Bi-LSTM-CNN model has been used for named entity recognition in Indonesian, which features four classes including person, organization, location, and event [52]. Song *et al.* [53] proposed two models: Bi-LSTM-CNN and CNN-LSTM, which proceed in reverse order to each other. The two models outperformed many baseline models. Their experimental results show the CNN-LSTM model outperforms CNN-LSTM and other models.

However, many deep learning models have been applied to many tasks and obtained significant performance. In recurrent years, the hybrid deep learning frameworks such as CNN-LSTM and LSTM-CNN are very useful for text feature extraction. Therefore, we use the hybrid framework to extract the text feature and predict the multiple emotion labels.

III. BI-LSTM-CNN MODEL FOR EMOTION LABEL IDENTIFICATION

We propose a deep learning framework for multiple emotion label identification using two powerful feature extractors with a word presentation approach. The two extractors are recurrent neural networks and convolutional neural networks used for feature extractions, and word embeddings are used to capture the relationship between each word pair for representation. The goal of our proposed deep learning framework is to obtain the optimal model parameters θ^*, which is defined as:

$$\theta^* = \arg\min_{\theta} L(\theta) \tag{1}$$

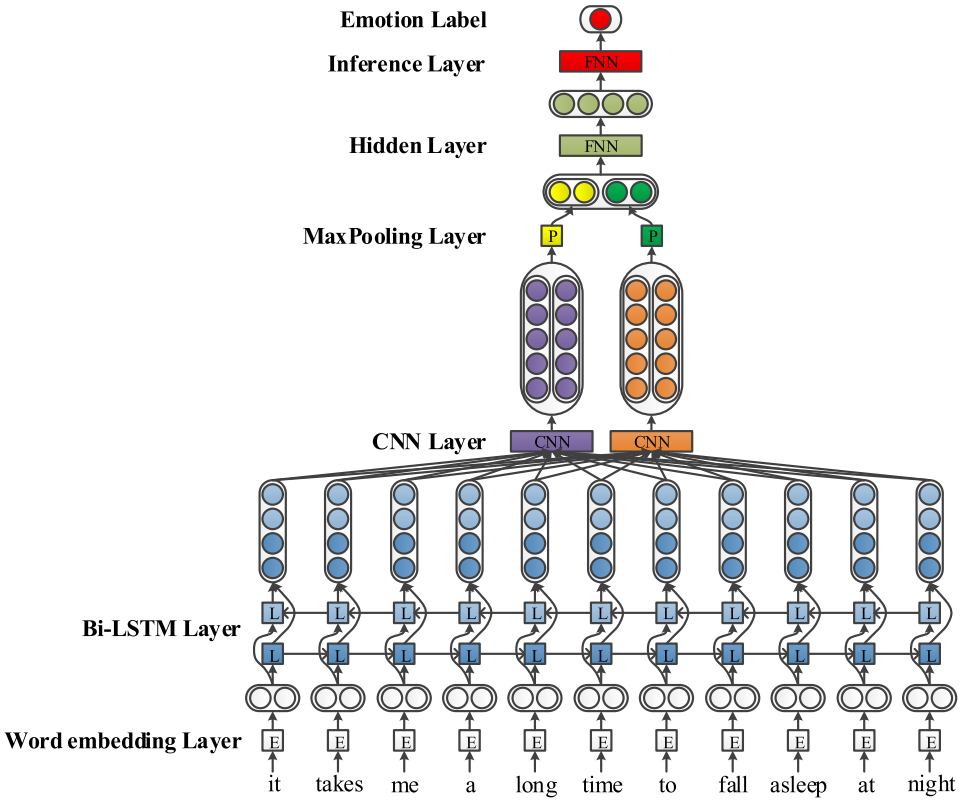


FIGURE 1. Proposed Bi-LSTM-CNN classifier.

where θ^* is the optimal model parameters to be weighted by gradient-based optimization.

Figure 1 shows the overall framework for emotion label identification. We propose a deep neural network model comprising six layers of neural networks including a word embedding layer, a Bi-LSTM layer, a CNN layer, a max-pooling layer, a hidden layer, and an inference layer. (1) The word embedding layer (E) is used to learn the word vector. (2) The Bi-LSTM layer serves as a feature extractor (L) to extract the features of each word vector; it can read a sentence through both forward and backward passes. (3) The CNN layer is a feature extractor used to convolute the hidden features of Bi-LSTM; it can capture local important features in a sentence using a fixed window size. (4) The max-pooling operation (P) selects the maximum value of each convoluted feature to obtain the most important feature. (5) The hidden layer is an FNN used to map the maximum hidden feature to a new hidden feature vector. (6) Finally, the inference layer is used to predict the probabilities of different emotion labels.

Below we describe the three major models used to build our proposed Bi-LSTM-CNN classifier:

A. WORD EMBEDDINGS TRAINING

The GloVe approach is used to train word embeddings from a larger Chinese corpus and domain corpus. The GloVe algorithm optimizes the co-occurrence probability of words i and j and is performed on aggregated global word-word co-occurrence statistics from a larger corpus. In this paper, we use the GloVe tool to create the word embeddings for the next emotion label detection.

B. EXTRACTOR OF BI-LSTM NEURAL NETWORK

This section seeks to capture the meaningful hidden features from each word in the query sentence using word embeddings pre-trained from the GloVe model. The bi-directional long short-term memory (Bi-LSTM) is a powerful extraction model for sequence data [54]. Therefore, we propose a Bi-LSTM model to extract hidden features by capturing raw text information due to the long length of sentences in psychiatric social texts. Psychiatric social texts include multiple emotion labels, so the Bi-LSTM is used to capture multiple features by forward and backward mapping. The Bi-LSTM is used to map word embedding of the sequence $S = [w_1, w_2, \ldots, w_T]$ into a hidden feature $\mathbf{H}^{\mathrm{L}} = [\mathbf{h}_1^{\mathrm{L}}, \mathbf{h}_2^{\mathrm{L}}, \ldots, \mathbf{h}_T^{\mathrm{L}}]$. Since each word is represented using word vectors from word embeddings, to compute a hidden representation of each word, the $\mathbf{h}_t^{\mathrm{L}} = [\vec{\mathbf{h}}_t^{\mathrm{L}}, \overleftarrow{\mathbf{h}}_t^{\mathrm{L}}]$ is the concatenated output of the Bi-LSTM, where $\vec{\mathbf{h}}_t^{\mathrm{L}}$ is a forward LSTM over a sequence $S = [\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_T]$ and $\overleftarrow{\mathbf{h}}_t^{\mathrm{L}}$ is a backward LSTM over a sequence $S = [w_T, w_{T-1}, \ldots, w_1]$.

The monodirectional LSTM is computed as follows:

$$\mathbf{f}_t^{\mathrm{L}} = \sigma(\mathbf{w}_f[\mathbf{h}_{t-1}^{\mathrm{L}}, w_t] + \mathbf{b}_f) \tag{2}$$

$$\mathbf{i}_t^{\mathrm{L}} = \sigma(\mathbf{w}_i[\mathbf{h}_{t-1}^{\mathrm{L}}, w_t] + \mathbf{b}_i) \quad (3)$$
$$\mathbf{o}_t^{\mathrm{L}} = \sigma(\mathbf{w}_o[\mathbf{h}_{t-1}^{\mathrm{L}}, w_t] + \mathbf{b}_o) \quad (4)$$
$$\tilde{\mathbf{c}}_t^{\mathrm{L}} = \tanh(\mathbf{w}_c[\mathbf{h}_{t-1}^{\mathrm{L}}, w_t] + \mathbf{b}_c) \quad (5)$$
$$\mathbf{c}_t^{\mathrm{L}} = \mathbf{f}_t^{\mathrm{L}} \otimes \mathbf{c}_{t-1}^{\mathrm{L}} + (1 - \mathbf{i}_t^{\mathrm{L}}) \otimes \tilde{\mathbf{c}}_t^{\mathrm{L}} \quad (6)$$
$$\mathbf{h}_t^{\mathrm{L}} = \mathbf{o}_t^{\mathrm{L}} \otimes \tanh(\mathbf{c}_t^{\mathrm{L}}) \quad (7)$$

where $[\mathbf{h}_{t-1}^{\mathrm{L}}, w_t] \in R^{ws+hs}$ is a concatenation vector of the previously hidden state h_{t-1}^{L} and the current word embedding as input $\mathbf{e}_t$. $\mathbf{w}_f$, $\mathbf{w}_i$, $\mathbf{w}_o$, $\mathbf{w}_s \in R^{hs\times(hs+ws)}$, and $\mathbf{b}_f$, $\mathbf{b}_i$, $\mathbf{b}_o$, $\mathbf{b}_s \in R^{hs}$ are learnable parameters. σ and $\otimes$ are respectively a logistic sigmoid function and an elementwise multiplication. The *tanh* is a tanh function as activation.

C. CNN EXTRACTOR WITH MAX-POOLING

All words are encoded by a Bi-LSTM feature extractor. A convolution operation involves a filter $\mathbf{w}_{cnn} \in R^{k\times hs}$, which is applied to a window of k hidden features to produce a new hidden feature [55]. This paper designs multiple filters for the CNN layer because different filter sizes can capture different meaningful features. For example, a convoluted hidden feature c_i^{c} is generated from a window of hidden features $\mathbf{h}_{i:i+k-1}^{\mathrm{L}}$:

$$c_i^{\mathrm{c}} = \sigma(\mathbf{w}_{cnn} \cdot \mathbf{h}_{i:i+k-1}^{\mathrm{L}} + \mathbf{b}_{cnn}) \quad (8)$$

where $c_i^{\mathrm{c}} \in R$ is a convoluted feature. $\mathbf{b}_{cnn} \in R$ is the bias term. $\mathbf{w}_{cnn}$ and $\mathbf{b}_{cnn} \in R$ are learnable parameters. σ is a rectified linear unit (ReLU) function. The CNN is applied to each hidden feature in the Bi-LSTM hidden feature $\{\mathbf{h}_{1:k}^{\mathrm{L}}, \mathbf{h}_{2:k+1}^{\mathrm{L}}, \ldots \mathbf{h}_{i:i-k+1}^{\mathrm{L}}\}$to produce a feature map

$$\mathbf{c}^{\mathrm{c}} = [c_1^{\mathrm{c}}, c_1^{\mathrm{c}}, \ldots, c_{T-k+1}^{\mathrm{c}}] \quad (9)$$

with $\mathbf{c}^{\mathrm{c}} \in R^{T-k+1}$. In this case, we apply a max-overtime pooling operation over the hidden features and capture the most important feature $\hat{c} = \max\{\mathbf{c}^{\mathrm{c}}\}$ as the feature corresponding to each filter. The framework uses multiple filters (with varying windows sizes) to obtain multiple CNN-Max-Pooling filters' features as follows:

$$\mathbf{h}^{\mathrm{c}} = [\hat{c}_1, \hat{c}_2, \ldots, \hat{c}_m] \quad (10)$$

where $\mathbf{h}^{\mathrm{c}}$ denotes the concatenation hidden features which concatenate all features from each filter.

D. FNN HIDDEN FEATURE EXTRACTION AND OUTPUT PREDICTION

To predict the emotion labels, we use a fully connected neural network to extract the hidden feature from the hidden feature of CNN with max-pooling. The extracted hidden feature by FNN is defined as:

$$\mathbf{h}^{f_1} = \mathbf{w}_{f_1}\mathbf{h}^{\mathrm{c}} + \mathbf{b}_{f_1} \quad (11)$$

where, $\mathbf{h}^{f_1} \in R^{hs}$ is a new hidden feature mapping, and $\mathbf{w}_{f_1} \in R^{hs\times m}$ and $\mathbf{b}_{cnn} \in R^{hs}$ are trainable parameters. The last FNN layer of our proposed framework predicts the probabilities of various emotion labels and is defined as:

$$\hat{\mathbf{y}} = \sigma(\mathbf{w}_{out}\mathbf{h}^{f_1} + \mathbf{b}_{out}) \quad (12)$$

where $\hat{\mathbf{y}} \in R^l$ is the predictive probabilities of the emotion label, $\mathbf{w}_{out} \in R^{hs\times 1}$ and $\mathbf{b}_{cnn} \in R$ are trainable parameters, and σ is a sigmoid function.

E. MODEL TRAINING

The previous section describes how to build the Bi-LSTM-CNN model and there are many parameters of Bi-LSTM-CNN. In this section, the model training uses the gradient-based optimization approach to optimize our proposed Bi-LSTM-CNN model and to obtain a better predicting model. Therefore, we use the log-likelihood method to measure the model performance of training in multiple emotion label classifications. In this case, each label of the multiple emotion labels uses a Bi-LSTM-CNN model to train a specific binary classifier. The negative log-likelihood function L for the model parameters of the proposed neural network is defined as:

$$L(\theta) = -\sum_{i=1}^{n} \mathbf{y}^{(i)} \log \hat{\mathbf{y}}^{(i)} + (1 - \mathbf{y}^{(i)}) \log[1 - \hat{\mathbf{y}}^{(i)}] \quad (13)$$

where θ is all parameters in our proposed deep neural network model. We use the Nadam optimizer to optimize the model parameters according to negative log-likelihood loss.

IV. EXPERIMENTAL RESULTS

This section describes the experimental setup and results including the query dataset, comparative classifiers, evaluation metrics and classification performance comparisons.

A. DATASET

A corpus of 1,711 psychiatric social texts was collected from the PsychPark website (http://www.psychpark.org); a virtual psychiatric clinic maintained by a group of volunteer professionals, including psychiatrists, psychologists, social workers, etc., [56]. Each text is originally an e-mail submitted by a web visitor about his/her psychiatric problems. The professionals then assign one or multiple emotion labels to each e-mail and de-identify them to periodically update the online dataset. Table 2 shows the proportions of the emotion

TABLE 2. Distribution and count of emotion labels in the experimental data.

No.	Emotion Label	Count	Proportion
1	Depression	743	35.26%
2	Mood	633	30.04%
3	Drug	282	13.38%
4	Insomnia	122	5.79%
5	Social anxiety	119	5.65%
6	Schizophrenia	113	5.36%
7	OCD (Obsessive compulsive disorder)	95	4.51%

labels in the corpus, counted for each text. In the evaluation, 274 samples of the texts were randomly selected as a validation set for hyper-parameter tuning of all classifiers, and 343 samples were randomly selected as a test set. The remaining 1,094 samples were used for model training. In addition, an extra resource of Wikipedia Chinese text corpus is used to train word embeddings to obtain a more general semantic relationship for each word. However, the out of vocabulary issue is a considerable problem when using only the Wikipedia corpus because of the general nature of these words. Therefore, we combine the Wikipedia dataset with the psychiatric social texts to train word embeddings. Following training, each word has a vector which reflects the meaningful relationship between itself and each other word. These word embeddings capture the general semantic relationship from the Wikipedia corpus and specific domain relationship from the psychiatric social texts.

B. CLASSIFIERS

To evaluate our proposed Bi-LSTM-CNN model in the multiple emotion labeling problem, we built and compared four classifiers CNN, LSTM, LSTM-CNN, and Bi-LSTM-CNN. Each CNN and LSTM was implemented using a single layer structure.

CNN: This simple classifier only uses the CNN model with a max-pooling operation to extract hidden features and predict emotion labels.

LSTM: This simple classifier only uses the single directional LSTM model to extract hidden features, and we select the last hidden feature of LSTM to predict emotion labels.

LSTM-CNN: This combined classifier uses the single directional LSTM and CNN to extract hidden features. In addition, this LSTM model only selects the last encoded hidden feature to predict emotion labels.

Bi-LSTM-CNN: This relatively complex classifier is our proposed deep learning framework using CNN and Bi-LSTM feature extraction.

C. EVALUATION METRICS

To identify multiple emotion labels contained in the testing examples, each emotion label presented in Table 2 was used to train seven classifiers in the training phase. For the classifiers presented above, we built a multiclass classifier to output the probabilities of the emotion labels and used a threshold of 0.5 to determine positive labels. That is, each text may have more than one emotion label depending on whether or not the probability output by their corresponding classifiers exceeds the threshold of 0.5. The metrics used for performance evaluation included recall, precision, and F-measure defined as:

$$recall_i = \frac{\text{number of labels correctly classified}}{\text{number of label } i \text{ in gold standard}} \quad (14)$$

$$precision_i = \frac{\text{number of label } i \text{ correctly classified}}{\text{number of label } i \text{ by the classifier}} \quad (15)$$

$$f1_i = \frac{2(precision_i \times recall_i)}{precision_i + recall_i} \quad (16)$$

The above metrics are used to evaluate each emotion binary classification. To evaluate multiple emotion labels, we use the macro averaging method to compute overall metrics for the seven emotion labels. The three macro metrics are defined as:

$$macro\_recall = \frac{1}{7}\sum_{i=1}^{7} recall_i \quad (17)$$

$$macro\_precision = \frac{1}{7}\sum_{i=1}^{7} precision_i \quad (18)$$

$$macro\_f1 = \frac{1}{7}\sum_{i=1}^{7} f1_i \quad (19)$$

D. MODEL PARAMETERS SETUP

As summarized in Table 3, our proposed deep learning framework features six hyper-parameters: word embedding size, maximum sentence length, batch size, hidden size of CNN, kernel size of CNN and hidden size of FNN. To deal with varying sentence length, a zero-padding method is used to fix the sentence length at 500. That is, sentences with fewer than 500 words are padded with a zero value, whereas those exceeding a maximum value of 500 words are ignored.

TABLE 3. Best six hyper-parameter values in our proposed deep learning framework.

Hyper-parameter	Best Value
Word embedding size	300
Maximum sentence length	500
Batch size	10
Hidden size of CNN	128
Filter size of CNN	3 and 5
Hidden size of LSTM and FNN	64

E. CLASSIFICATION PERFORMANCE COMPARISONS ON WORD EMBEDDINGS

Chinese texts from Wikipedia were used to train word embeddings. All experiments used 300 dimensions in each classifier for word embedding training. First, we compared the performances of each classifier on the Chinese Wikipedia corpus and domain text corpus. Table 4 shows the emotion label classification performance in the validation set, both with and without the extra corpus. The training corpus combining Wikipedia and the domain improves the performance of all four classifiers by an average of 0.025. Among the four classifiers, our proposed Bi-LSTM-CNN performs the best. Thus, the additional corpus provides a slight improvement to classification performance. However, the extra words from the domain corpus training data provide more information, and thus, increase the *macro_f*1 metric.

The word2vec model is a very popular word representation approach, and we used it here for additional word embedding training. The experiments on the validation set used the four classifiers and the same training dataset (combined Chinese Wikipedia and psychiatric social texts), and the results are shown in Table 5. Here, we design 5 different dimension

TABLE 4. *macro_f*1 **Comparisons using different corpora on the validation dataset.**

Classifier	Corpus	
	Wikipedia + Domain	Wikipedia
CNN	**0.66**	0.63
LSTM	**0.47**	0.43
LSTM-CNN	**0.67**	0.65
Bi-LSTM-CNN	**0.69**	0.68

TABLE 5. *macro_f*1 **Comparisons between the GloVe and Word2vec models.**

Classifier	Embeddings model	Word Dimension Size					
		100	150	200	250	300	Avg.
CNN	GloVe	**0.644**	**0.653**	**0.656**	0.657	**0.662**	**0.654**
CNN	Word2Vec	0.638	0.650	0.650	**0.658**	0.655	0.650
LSTM	GloVe	**0.390**	**0.438**	**0.438**	**0.432**	**0.471**	**0.434**
LSTM	Word2Vec	0.354	0.400	0.419	0.401	0.427	0.400
LSTM-CNN	GloVe	**0.647**	**0.661**	**0.667**	**0.671**	**0.672**	**0.664**
LSTM-CNN	Word2Vec	0.623	0.620	0.643	0.659	0.657	0.640
Bi-LSTM-CNN	GloVe	**0.664**	**0.667**	**0.683**	**0.683**	**0.688**	**0.677**
Bi-LSTM-CNN	Word2Vec	0.634	0.657	0.655	0.669	0.665	0.656

sizes to evaluate classification performance for two models of word embeddings, with dimension sizes for all experiments set to 100, 150, 200, 250 and 300. Table 4 shows that the GloVe word embeddings training model performs best in 19 of the 20 experiments. Our proposed Bi-LSTM-CNN classifier provides the best classification performance, resulting in 0.664, 0.667, 0.683, 0.683 and 0.688 *macro_f*1 for respective dimension sizes of 100, 150, 200, 250 and 300, where the 300 is the best dimension size. The average *macro_f*1 of the four classifiers using GloVe are 0.654, 0.432, 0.664 and 0.677. The experimental results indicate combining the Chinese Wikipedia corpus and the domain Chinese texts improves classification performance over single corpora because word embeddings trained on the domain texts alone only obtain a limited relationship among the domain texts and do not provide sufficient information. The supplemental corpus can be used to define common relationships between these words, and the domain data can be tuned to the word relationship to capture the domain information. However, in this paper, word2vec does not outperform GloVe because we need to detect emotion labels from a longer sentence, but word2vec performs well in capturing local relationships in short sentences. This paper seeks to assign emotion labels to words in a sentence. Word embedding using the GloVe approach is performed on aggregated global word-word cooccurrence statistics from a corpus. Therefore, the GloVe approach can capture more relationship information within each sentence. Therefore, the GloVe approach outperforms word2vec.

F. CLASSIFICATION PERFORMANCE COMPARISONS FOR DIFFERENT NUMBER OF FILTERS

This experiment compares performance with different number of filters. Figure 2 shows the classification performance of Bi-LSTM-CNN with GloVe, with results showing that using two filters sized 3 and 5 obtain the best performance.

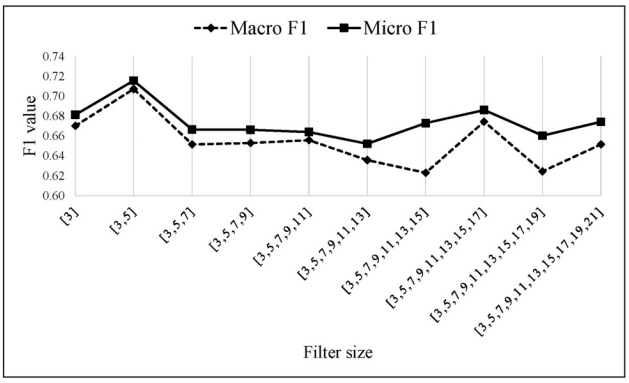


FIGURE 2. **Classification performance on different number of filters using classifier of Bi-LSTM-CNN with GloVe.**

TABLE 6. *macro_f*1 **Comparisons with other classifiers.**

Emotion Label	Classifier				
	Pure SVM	LSA-SVM	ICA-SVM	LSA+ICA-SVM	Bi-LSTM-CNN
Depression	0.59	0.71	0.73	**0.75**	0.72
Mood	0.54	0.58	0.62	0.64	**0.69**
Drug	0.47	0.63	0.59	0.62	**0.80**
Insomnia	0.59	0.60	0.70	**0.70**	0.69
Social anxiety	0.37	0.37	0.60	0.64	**0.76**
Schizophrenia	0.34	0.60	0.51	0.57	**0.73**
OCD	0.47	0.53	0.53	**0.57**	0.56
*macro_f*1	0.48	0.57	0.61	0.64	**0.71**
*micro_f*1	0.53	0.62	0.65	0.67	**0.72**

G. CLASSIFICATION PERFORMANCE COMPARISONS AGAINST OTHER CLASSIFIERS

Table 6 shows the classification performance results for the seven emotion labels in the test set. Our proposed Bi-LSTM-CNN classifier obtains a 0.71 average *macro_f*1, which is a respective 0.23, 0.14, 0.1 and 0.07 improvement over pure SVM, LSA-SVM, ICA-SVM and LSA+ICA-SVM. Two emotion labels (depression and insomnia) show approximate results using the proposed classifier and the LSA+ICA-SVM classifier. However, our proposed Bi-LSTM-CNN provides the maximum improvement (0.18) for the drug label. Overall, Bi-LSTM-CNN obtains significant improvements over these powerful neural network models. In addition, the micro f1 (*macro_f*1) of our proposed classifier is 0.72, and also outperforms the other classifiers. Therefore, the deep learning framework was successfully used for emotion label classification. We also performed classification using bi-directional LSTM-CNN and single directional LSTM-CNN. The Bi-LSTM-CNN classifier outperformed the LSTM-CNN classifier in emotion label classification since the learning direction of LSTM is the key learning component.

H. DISCUSSION

The single directional LSTM model learns from left to right, so the last hidden feature produces a higher gradient value on the last word of a query. However, psychiatric social text queries have multiple emotion labels; thus, the last hidden feature of the single directional LSTM gives more weight

TABLE 7. Example of two emotion labels in a query.

No.	Sentences in a query	Label
1	Good morning!	None
2	I've been experiencing a long-term **sleep disorder** and **have trouble sleeping** every day.	<Insomnia>
3	In addition to this, I have become **indifferent** because I have **no friends**, I'm **feeling down**, and have lost my goals.	<Depression>
4	Can you offer any solution to improve my life.	None

Note: the bold denotes the related emotion word

to the last emotion label. For example, Table 7 shows two emotion labels in a psychiatric social query: depression and insomnia. Sentence 2 describes the sleeping state, indicating a problem related to insomnia. Sentence 3 describes issues related to feeling, thinking, and handling daily activities, suggesting depression. However, the single directional LSTM classifies the text as belonging to depression only due to the left to right inference processing. Using the bi-directional LSTM, this query is found to belong to both insomnia and depression. Bi-directional LSTM helps avoid additional weighting on the words related to the last emotion label because it can obtain words related to both the first and last emotion label.

V. CONCLUSION AND FUTURE WORK

We proposed a deep learning model to assign emotion labels to psychiatric social texts. The proposed Bi-directional LSTM-CNN combines word embedding, long short-term memory networks, and convolutional neural networks to extract the hidden features. The major decision hidden features are obtained from the previous hidden features by CNN, and the word hidden features are obtained from the word embeddings. The abstractive hidden features are then successfully used to identify emotion labels. Therefore, the pipeline feature extraction processes, such as word embedding layer, bi-directional LSTM layer, and CNN layer extract many important features and provide high detection performance. Experimental results show that the proposed deep learning model significantly improved performance over other conventional models. Our proposed model using pretrained word embeddings through the GloVe model outperformed random initial word embeddings. Future work will focus on other deep learning approaches such as the attention-based model and tree-LSTM to improve performance and will include additional sentiment corpora to allow the system to capture more sentiment information.

REFERENCES

[1] L.-C. Yu, C.-H. Wu, and F.-L. Jang, "Psychiatric document retrieval using a discourse-aware model," *Artif. Intell.*, vol. 173, nos. 7–8, pp. 817–829, May 2009.

[2] M.-L. Zhang and Z.-H. Zhou, "A review on multi-label learning algorithms," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 8, pp. 1819–1837, Aug. 2014.

[3] L.-C. Yu and C.-Y. LC, "Identifying emotion labels from psychiatric social texts using independent component analysis," in *Proc. 25th Int. Conf. Comput. Linguistics (COLING)*, 2014, pp. 837–847.

[4] L.-C. Yu, L.-H. Lee, S. Hao, J. Wang, Y. He, J. Hu, K. R. Lai, and X. Zhang, "Building Chinese affective resources in valence-arousal dimensions," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol. (NAACL-HLT)*, 2016, pp. 540–545.

[5] J.-L. Wu, L.-C. Yu, and P.-C. Chang, "Emotion classification by removal of the overlap from incremental association language features," *J. Chin. Inst. Engineers*, vol. 34, no. 7, pp. 947–955, Oct. 2011.

[6] J. Ma, Z. Tian, H. Zhang, and T. W. S. Chow, "Multi-label low-dimensional embedding with missing labels," *Knowl.-Based Syst.*, vol. 137, pp. 65–82, Dec. 2017.

[7] Y. Wang, Y. Rao, X. Zhan, H. Chen, M. Luo, and J. Yin, "Sentiment and emotion classification over noisy labels," *Knowl.-Based Syst.*, vol. 111, pp. 207–216, Nov. 2016.

[8] Y. Huang, W. Wang, L. Wang, and T. Tan, "Multi-task deep neural network for multi-label learning," in *Proc. IEEE Int. Conf. Image Process.*, Sep. 2013, pp. 2897–2900.

[9] J. Fürnkranz, E. Hüllermeier, E. L. Mencía, and K. Brinker, "Multilabel classification via calibrated label ranking," *Mach. Learn.*, vol. 73, no. 2, pp. 133–153, Nov. 2008.

[10] G. Tsoumakas and I. Vlahavas, "Random K-labelsets: An ensemble method for multilabel classification," in *Proc. Eur. Conf. Mach. Learn. (ECML)*, 2007, pp. 406–417.

[11] M.-L. Zhang and Z.-H. Zhou, "ML-KNN: A lazy learning approach to multi-label learning," *Pattern Recognit.*, vol. 40, no. 7, pp. 2038–2048, Jul. 2007.

[12] A. Clare and R. D. King, "Knowledge discovery in multi-label phenotype data," in *Proc. 5th Eur. Conf. Princ. Data Mining Knowl. Discovery*, 2002, pp. 42–53.

[13] A. Elisseeff and J. Westom, "A kernel method for multi-labelled classification," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 14, 2001, pp. 681–687.

[14] J. Nam, J. Kim, E. L. Mencía, I. Gurevych, and J. Fürnkranz, "Large-scale multi-label text classification—Revisiting neural networks," in *Machine Learning and Knowledge Discovery in Databases* (Lecture Notes in Computer Science), vol. 8725. Berlin, Germany: Springer-Verlag, 2013, pp. 437–452.

[15] Z. Cui, X. Shi, and Y. Chen, "Sentiment analysis via integrating distributed representations of variable-length word sequence," *Neurocomputing*, vol. 187, pp. 126–132, Apr. 2016.

[16] M. Abdel Fattah, "New term weighting schemes with combination of multiple classifiers for sentiment analysis," *Neurocomputing*, vol. 167, pp. 434–442, Nov. 2015.

[17] S. Xiong, K. Wang, D. Ji, and B. Wang, "A short text sentiment-topic model for product reviews," *Neurocomputing*, vol. 297, pp. 94–102, Jul. 2018.

[18] J. Zhou, Y. Lu, H.-N. Dai, H. Wang, and H. Xiao, "Sentiment analysis of Chinese microblog based on stacked bidirectional LSTM," *IEEE Access*, vol. 7, pp. 38856–38866, 2019.

[19] G. Xu, Z. Yu, H. Yao, F. Li, Y. Meng, and X. Wu, "Chinese text sentiment analysis based on extended sentiment dictionary," *IEEE Access*, vol. 7, pp. 43749–43762, 2019.

[20] H. Xia, Y. Yang, X. Pan, Z. Zhang, and W. An, "Sentiment analysis for online reviews using conditional random fields and support vector machines," *Electron. Commerce Res.*, pp. 1–18, May 2019, doi: 10.1007/s10660-019-09354-7.

[21] A. M. G. Almeida, R. Cerri, E. C. Paraiso, R. G. Mantovani, and S. B. Junior, "Applying multi-label techniques in emotion identification of short texts," *Neurocomputing*, vol. 320, pp. 35–46, Dec. 2018.

[22] E. Barbu, "Property type distribution in Wordnet, corpora and Wikipedia," *Expert Syst. Appl.*, vol. 42, no. 7, pp. 3501–3507, May 2015.

[23] R. Socher. (2016). *Deep Learning for Natural Language Processing*. Accessed: Sep. 1, 2018. [Online]. Available: http://cs224d.stanford.edu/lectures/CS224d-Lecture2.pdf

[24] J. Turian, L. Ratinov, and Y. Bengio, "Word representations: A simple and general method for semi-supervised learning," in *Proc. 48th Annu. Meeting Assoc. Comput. Linguistics (ACL)*, 2010, pp. 384–394.

[25] D. L. T. Rohde, L. M. Gonnerman, and D. C. Plaut, "An improved model of semantic similarity based on lexical co-occurrence," *Commun. ACM*, vol. 8, no. 116, pp. 627–633, Nov. 2006.

[26] Y. Bengio, R. Ducharme, P. Vincent, and C. Janvin, "A neural probabilistic language model," *J. Mach. Learn. Res.*, vol. 3, pp. 1137–1155, Feb. 2003.

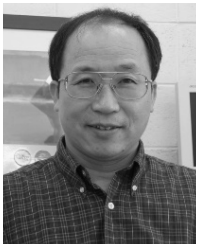
[27] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Proc. Adv. Neural Inf. Process. Syst.*, 2013, pp. 3111–3119.

[28] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *CoRR*, vol. abs/1301.3781, pp. 1–12, Jan. 2013.
[29] J. Pennington, R. Socher, and C. Manning, "Glove: Global vectors for word representation," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2014, pp. 1532–1543.
[30] M. Faruqui, J. Dodge, S. K. Jauhar, C. Dyer, E. Hovy, and N. A. Smith, "Retrofitting word vectors to semantic lexicons," in *Proc. NAACL*, 2015, pp. 1606–1615.
[31] L.-C. Yu, J. Wang, K. R. Lai, and X. Zhang, "Refining word embeddings using intensity scores for sentiment analysis," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 26, no. 3, pp. 671–681, Mar. 2018.
[32] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," 2018, *arXiv:1810.04805*. [Online]. Available: http://arxiv.org/abs/1810.04805
[33] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, "Language models are unsupervised multitask learners," *OpenAI Blog*, vol. 1, no. 8, pp. 1–9, 2019.
[34] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. 31st Annu. Conf. Neural Inf. Process. Syst.*, 2017, pp. 6000–6010.
[35] M. Chen, Y. Hao, K. Hwang, L. Wang, and L. Wang, "Disease prediction by machine learning over big data from healthcare communities," *IEEE Access*, vol. 5, pp. 8869–8879, 2017.
[36] F. Karim, S. Majumdar, H. Darabi, and S. Chen, "LSTM fully convolutional networks for time series classification," *IEEE Access*, vol. 6, pp. 1662–1669, 2018.
[37] G. Lee, J. Jeong, S. Seo, C. Kim, and P. Kang, "Sentiment classification with word localization based on weakly supervised learning with a convolutional neural network," *Knowl.-Based Syst.*, vol. 152, pp. 70–82, Jul. 2018.
[38] L. Li, T. T. Goh, and D. Jin, "How textual quality of online reviews affect classification performance: A case of deep learning sentiment analysis," *Neural Comput. Appl.*, Jul. 2018, doi: 10.1007/s00521-018-3865-7.
[39] J. Wang, L.-C. Yu, K. R. Lai, and X. Zhang, "Dimensional sentiment analysis using a regional CNN-LSTM model," in *Proc. 54th Annu. Meeting Assoc. Comput. Linguistics*, vol. 2, 2016, pp. 225–230.
[40] Y. Xin, L. Kong, Z. Liu, Y. Chen, Y. Li, H. Zhu, M. Gao, H. Hou, and C. Wang, "Machine learning and deep learning methods for cybersecurity," *IEEE Access*, vol. 6, pp. 35365–35381, 2018.
[41] L.-C. Yu, J. Wang, X. Zhang, and K. R. Lai, "Pipelined neural networks for phrase-level sentiment intensity prediction," *IEEE Trans. Affect. Comput.*, early access, Feb. 20, 2018, doi: 10.1109/TAFFC.2018.2807819.
[42] F. A. Gers and E. Schmidhuber, "LSTM recurrent networks learn simple context-free and context-sensitive languages," *IEEE Trans. Neural Netw.*, vol. 12, no. 6, pp. 1333–1340, 2001.
[43] G. Rao, W. Huang, Z. Feng, and Q. Cong, "LSTM with sentence representations for document-level sentiment classification," *Neurocomputing*, vol. 308, pp. 49–57, Sep. 2018.
[44] J. Wang, B. Peng, and X. Zhang, "Using a stacked residual LSTM model for sentiment intensity prediction," *Neurocomputing*, vol. 322, pp. 93–101, Dec. 2018.
[45] Y. Xie, F. Zhu, J. Wang, R. Liang, L. Zhao, and G. Tang, "Long-short term memory for emotional recognition with variable length speech," in *Proc. 1st Asian Conf. Affect. Comput. Intell. Interact. (ACII Asia)*, May 2018, pp. 1–4.
[46] M. Al-Smadi, B. Talafha, M. Al-Ayyoub, and Y. Jararweh, "Using long short-term memory deep neural networks for aspect-based sentiment analysis of arabic reviews," *Int. J. Mach. Learn. Cybern.*, vol. 10, no. 8, pp. 2163–2175, Aug. 2019.
[47] Q. Li, S. Li, J. Hu, S. Zhang, and J. Hu, "Tourism review sentiment classification using a bidirectional recurrent neural network with an attention mechanism and topic-enriched word vectors," *Sustainability*, vol. 10, no. 9, p. 3313, 2018.
[48] Z. Tian, W. Rong, L. Shi, J. Liu, and Z. Xiong, "Attention aware bidirectional gated recurrent unit based framework for sentiment analysis," in *Proc. Int. Conf. Knowl. Sci., Eng. Manage. (KSEM)*, 2018, pp. 67–78.
[49] Y. He, L.-C. Yu, K. R. Lai, and W. Liu, "YZU-NLP at EmoInt-2017: Determining emotion intensity using a bi-directional LSTM-CNN model," in *Proc. 8th Workshop Comput. Approaches Subjectivity, Sentiment Social Media Anal.*, 2017, pp. 238–242.
[50] M. Pei, X. Wu, Y. Guo, and H. Fujita, "Small bowel motility assessment based on fully convolutional networks and long short-term memory," *Knowl.-Based Syst.*, vol. 121, pp. 163–172, Apr. 2017.
[51] H. Wang, Z. Yang, Q. Yu, T. Hong, and X. Lin, "Online reliability time series prediction via convolutional neural network and long short term memory for service-oriented systems," *Knowl.-Based Syst.*, vol. 159, pp. 132–147, Nov. 2018.
[52] S. L. Oh, E. Y. K. Ng, R. S. Tan, and U. R. Acharya, "Automated diagnosis of arrhythmia using combination of CNN and LSTM techniques with variable length heart beats," *Comput. Biol. Med.*, vol. 102, pp. 278–287, Nov. 2018.
[53] M. Song, X. Zhao, Y. Liu, and Z. Zhao, "Text sentiment analysis based on convolutional neural network and bidirectional LSTM model," in *Proc. Int. Conf. Pioneering Comput. Scientists, Eng. Educators*, 2018, pp. 55–68.
[54] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Trans. Signal Process.*, vol. 45, no. 11, pp. 2673–2681, Nov. 1997.
[55] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Commun. ACM*, vol. 60, no. 6, pp. 84–90, May 2017.
[56] Y. M. Bai, C. C. Lin, J. Y. Chen, and W. C. Liu, "Virtual Psychiatric Clinics," *Amer. J. Psychiat.*, vol. 158, no. 7, pp. 1160–1161, 2001.

JHENG-LONG WU received the Ph.D. degree in information management from Yuan Ze University, Taiwan. He is currently an Assistant Professor with the School of Big Data Management, Soochow University, Taiwan. His research interests include natural language processing, sentiment analysis, deep learning, and text mining.

YUANYE HE received the M.S. degree from the Department of Computer Science and Engineering, Yuan Ze University, Taiwan. He is currently working with the Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China. His research interests include natural language processing, text mining, and machine learning.

LIANG-CHIH YU received the Ph.D. degree in computer science and information engineering from National Cheng Kung University, Taiwan. He was a Visiting Scholar with the Natural Language Group, Information Sciences Institute, University of Southern California (USC/ISI), from 2007 to 2008, and with DOCOMO Innovations, for three months in 2018. He is currently a Professor with the Department of Information Management, Yuan Ze University, Taiwan. His research interests include natural language processing, sentiment analysis, and computer-assisted language learning. He is also a Board Member and a Convener of SIGCALL of the Association for Computational Linguistics and Chinese Language Processing (ACLCLP). He also serves as an Editorial Board Member of the *International Journal of Computational Linguistics* and *Chinese Language Processing*. His team has developed systems that ranked first in IJCNLP 2017 Task 4: Customer Feedback Analysis, and second in the recent SemEval and BEA shared task competitions.



K. ROBERT LAI received the Ph.D. degree in computer science from North Carolina State University, in 1992. He is currently a Professor with the Department of Computer Science and Engineering and the Director of the Innovation Center for Big Data and Digital Convergence, Yuan Ze University, Taiwan. His research interests include big data analytics, agent technologies, and mobile computing.