IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received August 27, 2019, accepted September 3, 2019, date of publication September 9, 2019, date of current version October 8, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2940051

Deep Learning Based Weighted Feature Fusion Approach for Sentiment Analysis

MOHD USAMA[1], WENJING XIAO[1], BELAL AHMAD[1], JIAFU WAN[2], MOHAMMAD MEHEDI HASSAN[3], AND ABDULHAMEED ALELAIWI[3]

[1]Embedded and Pervasive Computing (EPIC) Laboratory, School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan 430074, China
[2]Mechanical and Automotive Engineering, South China University of Technology, Guangzhou 510006, China
[3]College of Computer and Information Sciences, King Saud University, Riyadh 11543, Saudi Arabia

Corresponding author: Jiafu Wan (jiafu_wan@ieee.org)

The authors are grateful to King Saud University, Riyadh, Saudi Arabia for funding this work through Researchers Supporting Project number RSP-2019/18.

ABSTRACT Deep learning algorithms have achieved remarkable results in the natural language processing(NLP) and computer vision. Hence, a trend still going on to use these algorithms, such as convolution and recurrent neural networks, for text analytic task to extract useful information. Features extraction is one of the important reasons behind the success of these networks. Moreover passing features from one layer to another layer within the network and one network to another network have done. However multilevel and multitype features fusion remains unexplored in sentiment analysis. So, in this paper, we use three datasets to display the advantages of extracting and fusing multilevel as well as multitype features from different neural networks. Multilevel features are from different layers of the same network, and multitype features are from different network architectures. Experiment results demonstrate that the proposed model based on multilevel and multitype weighted features fusion outperforms than many exiting works with an accuracy of 80.2%, 48.2%, and 87.0% on MR, SST1, and SST2 datasets respectively.

INDEX TERMS Convolution neural network, recurrent neural network, feature fusion, feature learning, and sentiment analysis.

I. INTRODUCTION

The enormous and rapid growth of the social network, smart gadgets, and the internet brings together billions of users to generate short texts on the internet such as public opinion on services, products, movies, and blogs. These reviews as short texts are usually found to be semantic and subjective oriented. Identifying and classifying proper semantic orientation of text reviews written by authors on the internet is essential to research which solves customers as well as company's various practical value problems, such as what product a customer like or dislike? Moreover, whether the product is doing good in the market or not? Sentiment analysis of short text remains a challenging one because short text usually contains limited semantic and contextual information, which limits the accuracy of the analysis. The existing work in sentiment analysis has done by various methods such as data mining technique [1], cognitive computing [2] and machine learning algorithms [3].

With rapid growth of the deep learning, classical deep learning based algorithms such as the CNN (convolution neural network) and RNN (recurrent neural network) have attained remarkable success in many areas such as NLP [5], computer vision [4], speech recognition [6], disease diagnosis [7], [8], Smart healthcare [9]–[11], Robotics [12], and 5G [13]. Typical CNN [14] and RNN [15], [16] have already been used for sentiment analysis and achieved remarkable results. Kim [17] used CNN to perform sentence-level classification with pre-trained word vectors. The recursive deep learning studied for sentiment classification by Socher *et al.* [18]–[20]. Tai *et al.* [21] proposed a generalization of long short-term memory(LSTM) network for prediction of semantic relatedness and sentiment classification of text, named as Tree-LSTM. Wang and Manning [22] utilized machine learning methods for sentiment analysis and shows that NB performs better on the short text while SVM on the long text by

The associate editor coordinating the review of this manuscript and approving it for publication was Victor Hugo Albuquerque.

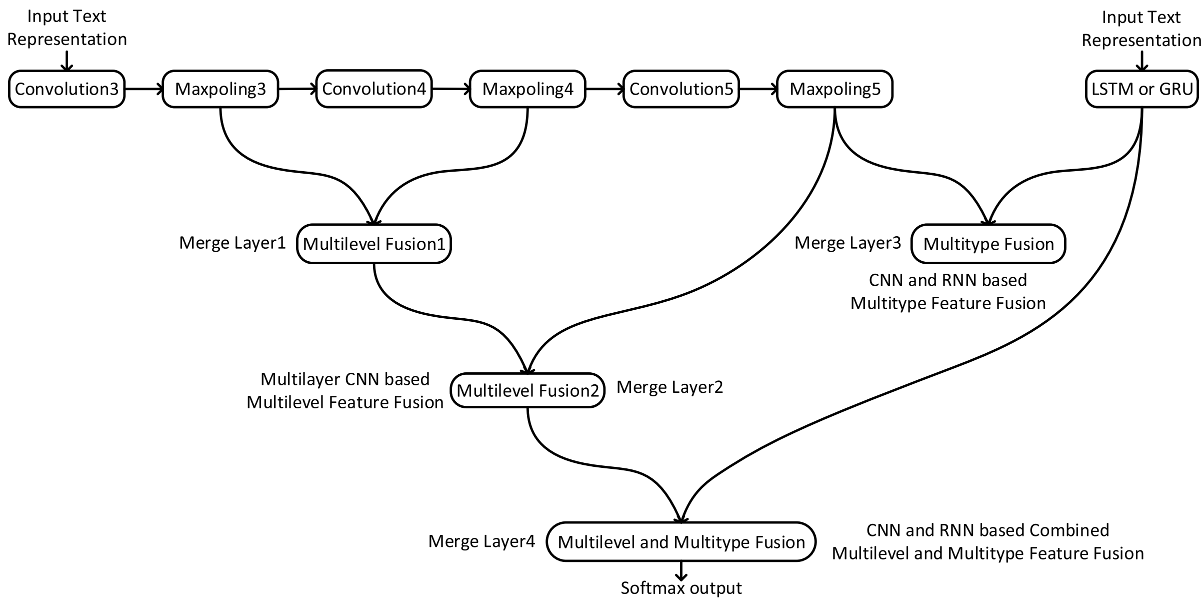

FIGURE 1. Structured diagram of the proposed model to represent the features fusion approach. Input text representation is given to CNN as well as RNN to learn the multilevel and multitype features. Multilevel features fusion at merge layer2, multitype feature fusion at merge layer3, and combined multilevel and multitype feature fusion at merge layer4 are used in the experiments.

including bigram features. Wang and Manning [23] proposed an approach of getting the benefit of the dropout training without sampling.

After that, an exclusive idea of research by combining two networks begun. Very few researchers used two networks together in one model for sentiment classification. Kim *et al.* [24] presented the neural language model that receives input at the character-level and predict the output at word-level. Their model derived by combining CNN, highway network, and LSTM. In his model output from the highway network and CNN over characters is used as input for LSTM RNN-LM (recurrent neural network language model). Chen and Hao [25] and Vu *et al.* [26] both utilized two distinct neural networks together and combine CNN and RNN for relation classification. All these model achieved better results compared to existing models. Furthermore, Hassan and Mahmood [27] put forward an end-to-end bottom-up architecture by combining CNN and RNN for sentiment analysis. The present sequence architecture in their paper and used word embedding as an input to the convolutional layer, which learned features maps by using the window of different sizes and various weights. Then obtained output of convolutional layer is passed as an input to the LSTM layer, where LSTM learned long-term dependencies from these feature maps to generate sentence level representation.

In this paper, the proposed model performs multilevel and multi-type features extraction and features fusion by combining convolutional and recurrent neural network differently. In the proposed scheme, both CNN and RNN received sentiment text as an input and learns different features according to network architecture. The CNN itself constitutes of three convolution layers with different filter sizes. First, we give word embedding as an input to CNN and learn multilevel contextual features from every layer of CNN and perform multilevel features fusion as shown in Figure 1. Similarly, we give word embedding again as an input to RNN and learn temporal features in sentiment text. Then, we merge these multiple type features at merge layer to get combined multilevel and multitype features fusion. Finally, the softmax classifier is use to accomplish the sentiment classification. In contrast to the existing models, contributions of the proposed model are as follow:

- We separately learn CNN and RNN type features by using word embedding as an input for both CNN and RNN and merge both types of features to get multitype features fusion, Instead of passing CNN features to RNN in sequence way as done in exiting works, then perform sentiment analysis over the union of features map.
- Within CNN we used three convolution layers with different filter sizes. We give word embedding to convolution layer as input and records multilevel features after max-pooling layer as shown in Figure 1 to get the final feature map from CNN.
- After getting multilevel feature fusion from CNN, combined multilevel and multitype features fusion is performed at the merged layer by using multilayer CNN and RNN as shown in Figure 1.

We perform experimentation on three benchmark datasets for sentiment classification and achieved better results compared to existing work. The remaining article is arrange as follows. In section 2 we discuss the related works. In section 3 we describe the proposed model with implementation details. Section 4 explains the experiment setup, datasets, and model variations. Section 5 discuss the experiment results and finally, section 6 presents the conclusion of the paper.

II. RELATED WORK

Nowadays sentiment analysis is a rapidly growing research topic in NLP [33]. It has been done in two distinct levels of granularity i.e. document level and sentence level. In general sentiment classification can be viewed as the text classification problem too which has been solved through statistical learning methods [2], [1]. Later with growing deep learning algorithms, neural network-based models show remarkable results in sentiments analysis task [33]. Various classical models based on neural network frameworks such as CNN and RNN have been established to better represent the sentence and document for classification. For sentence-level classification, Kim [17] performs a series of experiments with pre-trained word vectors and reported better results on three out of seven datasets. Socher *et al.* [18]–[20] proposed recursive models for sentiment analysis. Tai *et al.* [21] proposed Tree-LSTM model, a generalization of LSTM for sentiment analysis from the text. Others researchers proposed the joint architecture model by combining more than one network. Hassan and Mahmood [27] proposed a joint model by combining CNN and RNN for sentiment classification. Kim *et al.* [24] proposed a new language model by combining CNN, LSTM, and highway network. Similarly for document-level classification, Moraes *et al.* [28] done empirical comparative study for documents-level sentiment classification and there are some others works too have been done in sentiment analysis at the document-level [29], [30]. The joint architecture model for document classification to have been proposed. Tang *et al.* [31] proposed a gating mechanism-based RNN for sentiment classification of document level. In text processing, RNN better models the long term dependency than CNN. However, classical RNN is unable to focus on salient parts in the documents, having essential meaning in sentiment analysis. Yang *et al.* [32] proposed attention mechanism based hierarchical neural model for document-level classification. The proposed models have a hierarchical architecture which maps the hierarchical format documents. The model presets two level attention which applied to the sentence as well as word-level, and allow the model to attend less or more important text during documents representation.

III. PROPOSED MODEL

The proposed model's architecture is shown in Figure 2. The model consists of the following parts: word embedding with the glove, CNN, RNN with its variants, e.g. LSTM and GRU (gated recurrent unit), and soft-max classifier. Word embedding is used to converts input text into numerical word vectors to fed into CNN and RNN. Multilayer CNN is used to get the multilevel features from different filter sizes. RNN layer is used to get the temporal features at input level and learns long-term dependencies. At the merged layer, we perform multilevel, multitype, and combined multilevel & multitype features fusion by using CNN and RNN as shown in Figure 1. the softmax classifier is use to accomplish the sentiment classification task.

A. WORD EMBEDDING AND INPUT REPRESENTATION

We know that text cannot be feed directly to neural networks. So first we have to convert our text words into numerical form. Thus to generate the word embedding, we use unsupervised pre-trained embedding vector from GloVe[2] to the obtained representation of word vectors.

Let d be the vocabulary size of dataset and X is an input sentence consisting of n words with each word embedding dimension m then embedding matrix A will have dimension space R^{m*d}. Hence, the input representation of the sentence is as follows:

$$X(t_1, t_2, t_3, \ldots, t_n), \quad X \epsilon R^{m*n} \tag{1}$$

where R^m is the dimension space of each word in the vocabulary.

B. CNN LAYERS

Convolution layer performs the convolution operation over word vector receive from the embedding layer in a successive sequence of row representation form. Let h words be selected at a time with weight matrix w of dimension $w \epsilon R^{h*m}$ to perform convolution operation as follows:

$$C_i = f(X_{i+h-1} * w + b_i) \tag{2}$$

Here, $C_i \epsilon R^{n-h+1}$ is the feature map generated with h words every time repeatedly, f is non-linear Relu function, and b_i is bias term. Then we perform the max-pooling operation over generated features map from convolution, which converts the features map into its half by selecting the features with its maximum activation value, as follow:

$$p_i = \max C_i \tag{3}$$

Here, $p_i \epsilon R^{n-h+1/2}$ is the new feature map. To get different level features from the convolutional layer, we use three the convolutional layers with filter size 3, 4, and 5. After that, we merge features from the different level of convolution layers after performing the max-pooling operation to get CNN final multilevel feature fusion output as shown in Figure 2.

C. RNN LAYERS

Since, RNN [35] process data in a sequential way and learn long term dependencies, thus instead of using CNN generated features we feed original word embedding as an input to RNN layer to learns temporal features from it. After then, we perform the multitype fusion of CNN and RNN generated features to conduct sentiment classification.

In the proposed model, we used two different variations for RNN layers, e.g. LSTM and GRU.

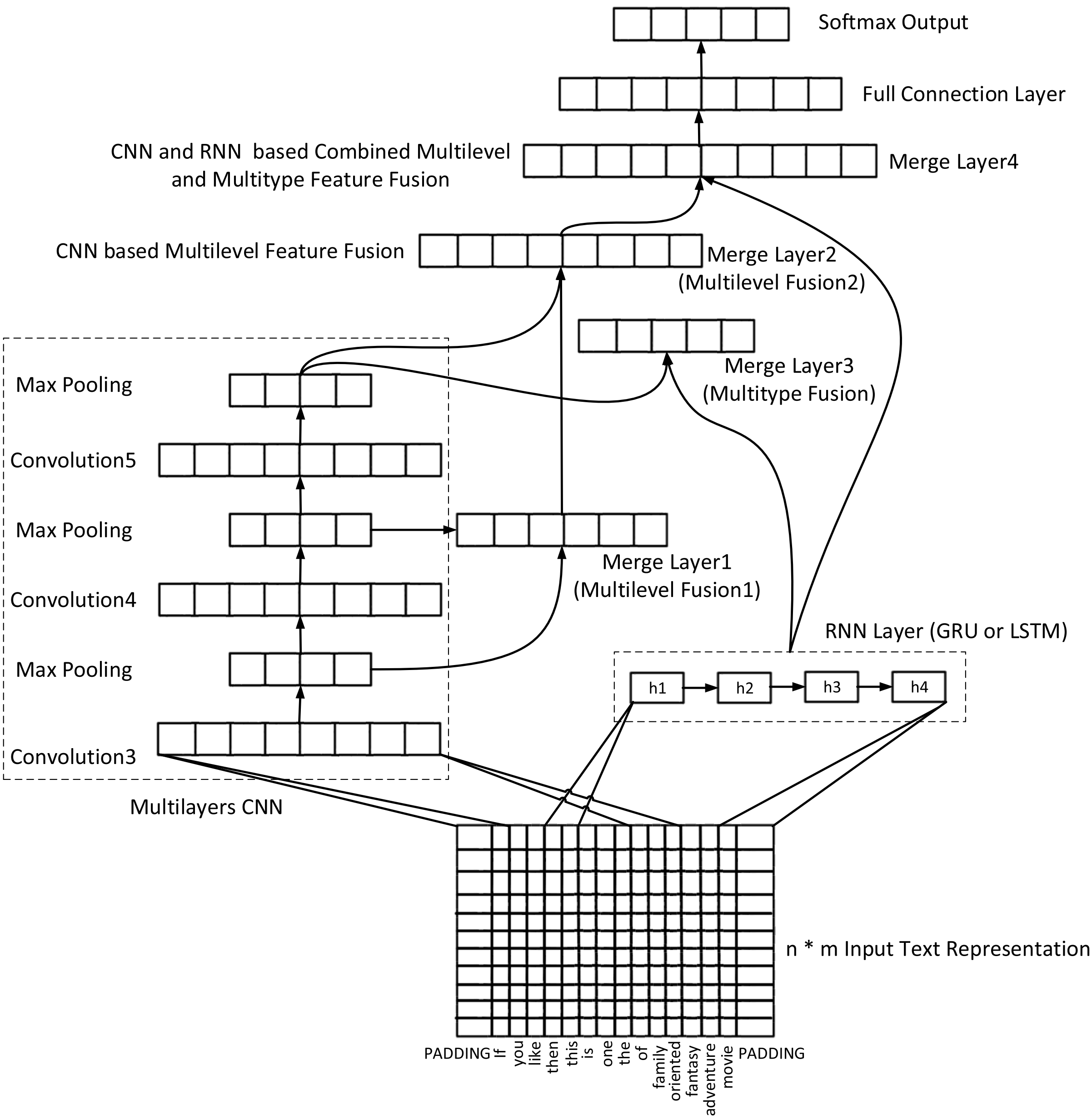

FIGURE 2. Proposed architecture for a sentence example. Input representation $n * m$ passes through RNN layer and multiple convolution layers with filter size 3, 4, and 5 to get multilevel and multitype features. Sub-sampling of the features map is performed for all convolution layers by using max-pool operation. Multilevel feature fusion is performed at merge layer2 based on multiple convolution layers, and multitype feature fusion is performed at merge layer3 based on CNN and RNN layer. After getting the multilevel features from CNN, we again fuse these multilevel features with features from RNN layer to get the combined multilevel and multitype features at merge layer4. Then after the full connection layer we use softmax classifier to get the sentiment classification results.

1) LSTM

LSTM is different from traditional RNN. Initially proposed by Hochreiter and Schmidhuber (1997) [36] to learn long-term dependencies.

Mathematical formulation of LSTM is define as follows:

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \tag{4}$$

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f) \tag{5}$$

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o) \tag{6}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot tanh(W_c x_t + U_c h_{t-1} + b_c) \tag{7}$$

$$h_t = o_t \odot tanh(c_t) \tag{8}$$

where x_t and h_t are the input and hidden state vector of LSTM unit at t time. o_t, i_t, and f_t are the activation vector of output, input, and forget gate respectively. U and W are the weight parameters. b is the bias vector. σ and c_t are the sigmoid function and memory cell state vector.

2) GRU

Initially, Cho *et al.* (2014) [37] proposed the GRU to capture the dependencies of every recurrent unit at different time scales. GRU is a gating mechanism in RNN and is like LSTM with forget gate.

Mathematical formulation of GRU is define as follows:

$$r_t = \sigma(W_r x_t + U_r h_{t?1} + b_r) \tag{9}$$
$$z_t = \sigma(W_z x_t + U_z h_{t?1} + b_z) \tag{10}$$
$$h_t = (1 - z_t) \odot h_{t?1} + z_t \odot tanh(W x_t + U(r_t \odot h_{t?1}) + b_h) \tag{11}$$

where h_t and x_t are the output and input vector at t time. r_t and z_t are the reset and update gate vector. $\odot$ and σ are the element-wise multiplication operator and sigmoid function. U and W are the weight parameters. b is the bias vector.

D. SOFTMAX CLASSIFIER

Finally, the feature fusion map from the full connection layer is passed to soft-max classifier, as shown in Figure 2, to perform sentiment classification which result is a probability distribution over all categories as follows:

$$Y_i = \frac{exp(h_i)}{\sum_{j=1}^{c} exp(h_j)} \tag{12}$$

where results Y_i is the probability distribution of i^{th} class overall classes.

To measure the disparity between real sentiment and predicted sentiment of the sentence in corpus text, we used cross-entropy as a loss function as follows:

$$loss = \sum_{s=T} \sum_{i=1}^{s'} Y_i^t(C) log(Y_i(C)) \tag{13}$$

where, $Y^t(C)$ is the element corresponding to the real sentiment of the sentence, $Y(C)$ is the element corresponding to predicted sentiment of the sentence, s' is the sentiment classes, and T is the training corpus text. We initiate training processes with pre-trained word vectors from GloVe then used stochastic gradient descendent method to train and update the parameters of the model.

IV. EXPERIMENT SETUP AND DATASETS

A. DATASETS

We experimented with three benchmark datasets to evaluate the proposed model. Statistical details of the all three datasets are given in Table 1.

- **MR[1](Movie Reviews):** This dataset contains a total of 10662 positive and negative reviews of movie with one sentence corresponds to one review. Challenge is to predict positive and negative sentiment associated with sentences [38].
- **SST1[2](Stanford Sentiment Treebank):** This dataset is the extension of the MR dataset with training, test, and deviation set splits provided with labeled into five classes as very positive, very negative, positive, negative, and neutral [18].
- **SST2[3]:** Similar to SST1 but with neutral reviews removed and binary labeled.

B. MODEL VARIATIONS

We experimented with several variations of the model. Generally, GRU performs better than LSTM and combined multilevel and multitype fusion performs better than multilevel and multitype fusion alone. Thus we choose combined multilevel and multitype fusion with GRU for random initialization.

- **CNN-GRU-Multilevel & multitype fusion:** A model with pre-trained word-vectors from GloVe, to perform combined multilevel and multitype feature fusion based on CNN and GRU.
- **CNN-LSTM-Multilevel & multitype fusion:** A model with pre-trained word-vectors from GloVe, to perform combined multilevel and multitype feature fusion based on CNN and LSTM.
- **CNN-GRU-Multitype fusion:** A model with pre-trained word-vectors from GloVe, to perform multitype feature fusion based on CNN and GRU.
- **CNN-LSTM-Multitype fusion:** A model with pre-trained word-vectors from GloVe, to perform multitype feature fusion based on CNN and LSTM.
- **CNN-Multilevel fusion:** A model with pre-trained word-vectors from GloVe, to perform multilevel feature fusion based on CNN.
- **CNN-GRU-Multilevel & multitype fusion-rand:** A model with random initialized word-vectors, to perform combined multilevel and multitype feature fusion based on CNN and GRU.

C. TRAINING PROCEDURE AND HYPERPARAMETERS

1) HYPERPARAMETERS

For all dataset, we used activation function rectified linear units, window size 3, 4, and 5 for convolution layers, feature maps 100 for each CNN layer, RNN layer output size 100, dropout 0.50, and minibatch size 50. The training was performed through a stochastic gradient descent algorithm with the Adadelta update method [39].

2) PRE-TRAINED WORD VECTORS

We initialized word-vectors with pre-trained set obtained from Glove, an unsupervised natural language algorithm to represent words into its numerical vector form which uses in lack of the large training dataset to improve performance of the model [20]. Publically available GloVe vectors with dimension 50 were used in experiments that trained on data from Wikipedia 2014 and Gigaword 5. Words which were absent in pre-train set were initialized randomly.

V. RESULTS AND DISCUSSION

To evaluate the proposed model, we used the system with specification 16GB RAM and 14Core CPU. Several variations of the model are tested during the experiment as

[1] https://www.cs.cornell.edu/people/pabo/movie-review-data/
[2] https://nlp.stanford.edu/sentiment/
[3] https://nlp.stanford.edu/projects/glove/

TABLE 1. Statistical details of the datasets after tokenization used in experiment. c: Number of classes, A_l: Average sentence length, M_l: Maximum sentence length, V: Dataset size, N: Vocabulary size, V_{Train}: Training set, V_{Test}: Test set, V_{Valid}:Validation set, CV indicates that there was no standard train/test/dev split for dataset so 10-fold Cross validation was used.

Dataset	c	A_l	M_l	V	N	V_{Train}	V_{Test}	V_{Valid}
MR	2	20	51	10662	18983	8655	1064	CV
SST1	5	18	56	11855	18784	8544	2210	1101
SST2	2	19	56	9613	18784	6920	1821	872

TABLE 2. Results of the proposed features fusion approach against existing works.

Group	Models	MR	SST1	SST2
Machine learning	MNB [22]	79.0	-	-
	NBSVMachine [22]	79.4	-	-
	bowwvSVM [14]	79.7	43.2	83.3
Recursive	Tree-CRF [40]	77.3	-	-
	RAE [20]	77.7	43.2	82.4
	MV-RNN [19]	79.0	44.4	82.9
	RNTN [18]	-	45.7	85.4
Recurrent	Tree LSTM1 [15]	-	48.0	-
	Tree bi-LSTM [16]	0.79	-	-
	LSTM [21]	-	46.4	84.9
Convolutional	Non-static CNN [17]	**81.5**	48.0	87.2
	Multi-channel CNN [17]	81.1	47.4	88.1
	Non-static GloVe+word2vec CNN [14]	81.0	45.9	85.6
Joint architecture	ConvLstm [27]	-	47.5	**88.3**
Others	VecAvg [18]	-	32.7	80.1
	CCAE [5]	77.8	-	-
	Gaussian Dropout [23]	79.0	-	-
	Fast Dropout [23]	79.1	-	-
	Sent-parser [34]	79.5	-	-
Proposed models	CNN-GRU-multilevel & multitype fusion	80.2	47.9	87.0
	CNN-LSTM-Multilevel & multitype fusion	79.8	47.4	86.5
	CNN-GRU-multitype fusion	79.6	**48.2**	86.2
	CNN-LSTM-multitype fusion	78.9	45.9	85.7
	CNN-multilevel fusion	79.9	48.1	86.9
	CNN-GRU-multilevel & multitype fusion-rand	78.4	46.2	85.6

For numeric values, bold represent the best results. **Tree-CRF**: Dependency tree with Conditional Random Fields. **RAE**: Recursive Autoencoders. **MV-RNN**: Matrix-Vector Recursive Neural Network. **MNB**: Multinomial Naive Bayes. **NBSVM**: Naive Bayes Support Vector Machine. **RNTN**: Recursive Neural Tensor Network. **VecAvg**: Word vector averages. **CCAE**: Combinatorial Category Autoencoders. **Sent-Parser**: Sentiment analysis-specific parser. **Tree bi-LSTM**: Tree bi-directional LSTM. **CNN-Non-static**: CNN with fine-tuned word2vec pre-trained vectors for each task. **CNN-Multi-channel**:Same as CNN-Non-static with two word vectors sets. **bowwvSVM**: SVM with concatenation of bow vectors and word2vec representations. **Non-static GloVe+word2vec CNN**: CNN trained with word2vec and GloVe. **ConvLstm**:CNN used LSTM as a pooling layer.

explained above. Results are listed in the table 2. As we expected, model variant with all word vectors randomly initialized (CNN-GRU-Multilevel & multitype fusion-rand) does not perform better among its model variation. While we got expected results by using pre-trained word vectors, even pre-trained model variants with only multilevel fusion approach (CNN-Multilevel fusion) perform amazing and achieve competitive performance again some sophisticated natural language models which utilized complex structure such as [18], [19]. The achieved results suggest that pre-trained vector performs well, better extracted the features than randomly initialized, and can be used for all the datasets and model variations. There can be an accuracy gain of 1-2% by using pre-trained vectors than random initialized.

A. EXPERIMENT RESULTS

1) RESULTS ANALYSIS ON MR DATASET

This dataset contains one movie review per sentence and challenge is to predict positive or negative sentiment associated with the sentence. As shown in Table 2 machine learning model [14] achieved accuracy up to 79.7% on this dataset which is a little lower than proposed model variation (CNN-GRU-multilevel & multitype fusion). Some of the works also done on this dataset through recurrent and recursive models (Table 2). There is not much difference found when we compared these results with machine learning works. But compared to the proposed model, our model performs better than all recurrent and recursive models and achieves 1.2% gain in accuracy approximate compared to recursive [20] and recurrent [16] respectively. Some researcher used a convolutional network-based model and reported state-of art-results against existing works. Even compared to proposed work, convolutional model [17] perform better with 1.3% gain in the accuracy. In conclusion, we can tell that the proposed feature fusion approach performs better than all existing works except the convolutional models on MR dataset.

2) RESULTS ANALYSIS ON SST1 DATASET

This fine-grain dataset having the challenge to predict sentiment according to the original category labeled as positive, negative, very positive, very negative, and neutral. There is one early existing work [14] based on machine learning achieve equal results to one of the recursive model [20] while other recursive models [22] reported better results than machine learning work. Some of the researchers also works with recurrent and convolutional model over this dataset and reported mix results but in particular recurrent [15] and convolutional model [17] reported the same accuracy i.e. 48.0%. Moreover, all recurrent and convolutional model performs better compared to recursive and machine learning models and proposed models performs better than recurrent and convolutional models. Reference [27] used joint architecture model and reported accuracy 47.5% which is better compared to all existing works except [16] and [17]. But compared to the joint architecture model, proposed model variation (CNN-GRU-multitype fusion) work better with 0.7% gain in accuracy. Among all reference [18] reported much worst results on this dataset i.e. 32.7%. In conclusion, we can tell that the proposed model variant (CNN-GRU-multitype fusion) performs better than all existing works and achieve state-of-art results on this dataset.

3) RESULTS ANALYSIS ON SST2 DATASET

SST2 is the reduced version of SST1 dataset with removed neutral reviews and labeled as positive and negative only. Unlike SST1 dataset, machine learning model bowwwSVM [14] perform better than recursive model [19], [20] while one of the recursive model RNTN [18] achieve better results with 2.1% gain in accuracy against machine learning model. Compared to the recursive model, the recurrent model has mixed results too. Recurrent model [21] achieve better results than recursive models [19], [20] while lower than [18]. Moreover, all proposed model variants perform better with higher accuracy against machine learning, recurrent, and recursive models. Some existing works on this dataset also done by convolutional model. As shown in table 2 all convolutional model achieve better results than existing recurrent, recursive, and machine learning model. Reference [27] used joint architecture model by combining convolutional and LSTM together, and reported even better results than convolutional models. In overall we can say that the proposed model performs better than many existing works but could not achieve state-of-the-art results over SST2 dataset.

B. DISCUSSION

1) FUSION APPROACH: MULTILEVEL VS MULTITYPE

Initially, we hoped that multitype fusion approach would perform better (by using the different features from different architectures, i.e., contextual features from CNN and temporal features from RNN) than multilevel fusion approach, especially on small datasets. However, the results achieved are mixed. For MR and SST2 datasets, multilevel fusion performs better than multitype fusion (for both combination CNN-LSTM and CNN-GRU) while for SST1 dataset, multilevel fusion perform better than multitype fusion only with CNN-LSTM. Multitype fusion with CNN-GRU achieves little better results than multilevel fusion on SST1 dataset.

2) FUSION APPROACH: LSTM VS GRU

In the case with the joint architecture approach of CNN with LSTM and GRU, as we initially expected, the multitype fusion approach with CNN-GRU performs better than multitype fusion with CNN-LSTM approach in all datasets. Similarly, the combined multilevel and multitype fusion approach with CNN-GRU performs better than combined multilevel and multitype fusion with CNN-LSTM in all datasets. In overall, the CNN-GRU architecture performs better than the CNN-LSTM architecture for both approaches, i.e., multitype fusion and combined multilevel and multitype fusion on all datasets.

3) FURTHER OBSERVATIONS

Experiments validate that proposed approach of multilevel and multitype features fusion from two different networks performs better than existing state-of-art works on the one out of three dataset for sentiment classification of the short text. Thus, we take benefits of both CNN and RNN architecture and proposed a join feature fusion model. Where CNN extract the multilevel local features from input data, and RNN learns long-term dependency by processing input data sequentially. Finally, we merge them to get multilevel and multitype feature fusion. There are some further observations related to the experiments as follows:

- In general, the proposed approach of feature fusion from two different architectures CNN and RNN performs better than CNN and RNN models alone. Moreover, CNN with GRU performs better than CNN with LSTM in the sentiment classification task.

- Especially, our approach of combined multilevel and multitype feature fusion with CNN and GRU (CNN-GRU-Multilevel & multitype fusion) is better performs on MR and SST2 dataset, while multitype fusion approach with CNN and GRU (CNN-GRU-Multitype fusion) is better perform on SST1 dataset. Sentiment classification accuracy is increased by 0.20% on SST1 dataset compared to existing works.
- Even our simplest weighted features fusion approach (multilevel fusion) achieve better results than some complex existing architecture [5], [40].
- During literature reading, we found that most of the researcher use word2vec pre-trained vectors having dimension 300 trained on words from Google News. However, in experiments, we used GloVe pre-trained vectors having dimension 50 trained on words from Wikipedia 2014 and Gigaword 5. The reason for doing this is to decrease the computation burden, and we still got comparable results.
- In general combined multilevel and multitype fusion performs better than multilevel fusion, and multilevel fusion performs better than multitype fusion.

VI. CONCLUSION

This paper presents a new approach of multilevel and multitype weighted features fusion from two different neural networks i.e. CNN and RNN variants. The proposed framework can extract not only contextual features by CNN but also temporal features by RNN variants to learn the long-term dependency in sentences. Experiment results demonstrate that the proposed approach achieves better results than existing state-of-art works on one out of three datasets in sentiment analysis task. In the future study, we will test our approach of weighted feature fusion in other tasks of natural language processing.

ACKNOWLEDGMENT

The authors are grateful to King Saud University, Riyadh, Saudi Arabia for funding this work through Researchers Supporting Project number RSP-2019/18.

REFERENCES

[1] S. M. Kim and E. Hovy, "Automatic detection of opinion bearing words and sentences," in *Proc. IJCNLP*, 2005, pp. 61–66.
[2] M. Chen, Y. Hao, H. Gharavi, and V. C. M. Leung, "Cognitive information measurements: A new perspective," *Inf. Sci.*, vol. 505, pp. 487–497, Dec. 2019.
[3] B. Pang, L. Lee, and S. Vaithyanathan, "Thumbs up?: Sentiment classification using machine learning techniques," in *Proc. EMNLP*, 2002, pp. 79–86.
[4] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. NIPS*, 2012, pp. 1097–1105.
[5] K. M. Hermann and P. Blunsom, "The role of syntax in vector space models of compositional semantics," in *Proc. ACL*, 2013, pp. 894–904.
[6] A. Graves, A.-R. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *Proc. ICASSP*, May 2013, pp. 6645–6649.
[7] M. Usama, B. Ahmad, J. Wan, M. S. Hossain, M. F. Alhamid, and M. A. Hossain, "Deep feature learning for disease risk assessment based on convolutional neural network with intra-layer recurrent connection by using hospital big data," *IEEE Access*, vol. 6, pp. 67927–67939, 2018. doi: 10.1109/ACCESS.2018.2879158.
[8] Y. Hao, M. Usama, J. Yang, M. S. Hossain, and A. Ghoneim, "Recurrent convolutional neural network based multimodal disease risk prediction," *Future Gener. Comput. Syst.*, vol. 92, pp. 76–83, Mar. 2019. doi: 10.1016/j.future.2018.09.031.
[9] M. Chen, W. Li, Y. Hao, Y. Qian, and I. Humar, "Edge cognitive computing based smart healthcare system," *Future Gener. Comput. Syst.*, vol. 86, pp. 403–411, Sep. 2018.
[10] M. Chen, J. Yang, X. Zhu, X. Wang, M. Liu, and J. Song, "Smart home 2.0: Innovative smart home system powered by botanical IoT and emotion detection," *Mobile Netw. Appl.*, vol. 22, no. 6, pp. 1159–1169, 2017.
[11] M. Chen, Y. Hao, C. Lai, D. Wu, Y. Li, and K. Hwang, "Opportunistic task scheduling over co-located clouds in mobile environment," *IEEE Trans. Services Comput.*, vol. 11, no. 3, pp. 549–561, 2018.
[12] M. Chen, J. Zhou, G. Tao, J. Yang, and L. Hu, "Wearable affective robot," *IEEE Access*, vol. 6, pp. 64766–64776, 2018.
[13] M. Chen, J. Yang, J. Zhou, Y. Hao, C.-H. Youn, and J. Zhang, "5G-smart diabetes: Toward personalized diabetes diagnosis with healthcare big data clouds," *IEEE Commun. Mag.*, vol. 56, no. 4, pp. 16–23, Apr. 2018. doi: 10.1109/MCOM.2018.1700788.
[14] Y. Zhang and B. Wallace, "A sensitivity analysis of (and Practitioners' Guide to) convolutional neural networks for sentence classification," 2016, *arXiv:1510.03820*. [Online]. Available: https://arxiv.org/abs/1510.03820
[15] X. Zhu, P. Sobhani, and H. Guo, "Long short-term memory over tree structures," 2015, *arXiv:1503.04881*. [Online]. Available: https://arxiv.org/abs/1503.04881
[16] J. Li, D. Jurafsky, M.-T. Luong, and E. Hovy, "When are tree structures necessary for deep learning of representations?" 2015, *arXiv:1503.00185*. [Online]. Available: https://arxiv.org/abs/1503.00185
[17] Y. Kim, "Convolutional neural networks for sentence classification," 2014, *arXiv:1408.5882*. [Online]. Available: https://arxiv.org/abs/1408.5882
[18] R. Socher, A. Perelygin, J. Wu, J. Chuang, C. D. Manning, A. Ng, and C. Potts, "Recursive deep models for semantic compositionality over a sentiment treebank," in *Proc. EMNLP*, Oct. 2013, pp. 1631–1642.
[19] R. Socher, B. Huval, C. D. Manning, and A. Y. Ng, "Semantic compositionality through recursive matrix-vector spaces," in *Proc. EMNLP*, Jul. 2012, pp. 1201–1211.
[20] R. Socher, J. Pennington, E. H. Huang, A. Y. Ng, and C. D. Manning, "Semi-supervised recursive autoencoders for predicting sentiment distributions," in *Proc. EMNLP*, Jul. 2011, pp. 151–161.
[21] K. S. Tai, R. Socher, and C. D. Manning, "Improved semantic representations from tree-structured long short-term memory networks," 2015, *arXiv:1503.00075*. [Online]. Available: https://arxiv.org/abs/1503.00075
[22] S. Wang and C. D. Manning, "Baselines and bigrams: Simple, good sentiment and topic classification," in *Proc. ACL*, Jul. 2012, pp. 90–94.
[23] S. Wang and C. Manning, "Fast dropout training," in *Proc. ICML*, Feb. 2013, pp. 118–126.
[24] Y. Kim, Y. Jernite, D. Sontag, and A. M. Rush, "Character-aware neural language models," 2015, *arXiv:1508.06615*. [Online]. Available: https://arxiv.org/abs/1508.06615
[25] M. Chen and Y. Hao, "Label-less learning for emotion cognition," *IEEE Trans. Neural Netw. Learn. Syst.*, to be published. doi: 10.1109/TNNLS.2019.2929071.
[26] N. T. Vu, H. Adel, P. Gupta, and H. Schütze, "Combining recurrent and convolutional neural networks for relation classification," in *Proc. NAACL-HLT*, 2016, pp. 534–539.
[27] A. Hassan and A. Mahmood, "Deep learning approach for sentiment analysis of short texts," in *Proc. ICCAR*, Apr. 2017, pp. 705–710.
[28] R. Moraes, J. F. Valiati, and W. P. G. Neto, "Document-level sentiment classification: An empirical comparison between SVM and ANN," *Expert Syst. Appl.*, vol. 40, no. 2, pp. 621–633, Feb. 2013.
[29] J. Barnes, "LTG-Oslo hierarchical multi-task network: The importance of negation for document-level sentiment in Spanish," 2019, *arXiv:1906.07599*. [Online]. Available: https://arxiv.org/abs/1906.07599
[30] V. S. Shirsat, R. S. Jagdale, and S. N. Deshmukh, "Document level sentiment analysis from news articles," in *Proc. Int. Conf. Comput., Commun., Control Automat. (ICCUBEA)*, Pune, India, Aug. 2017, pp. 1–40. doi: 10.1109/ICCUBEA.2017.8463638.
[31] D. Tang, B. Qin, and T. Liu, "Document modeling with gated recurrent neural network for sentiment classification," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, Sep. 2015, pp. 1422–1432.

[32] Z. Yang, D. Yang, C. Dyer, X. He, A. Smola, and E. Hovy, ''Hierarchical attention networks for document classification,'' in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol.*, Jun. 2016, pp. 1480–1489.

[33] E. Cambria, S. Poria, A. Gelbukh, and M. Thelwall, ''Sentiment analysis is a big suitcase,'' *IEEE Intell. Syst.*, vol. 32, no. 6, pp. 74–80, Nov./Dec. 2017.

[34] L. Dong, F. Wei, S. Liu, M. Zhou, and K. Xu, ''A statistical parsing framework for sentiment classification,'' 2014, *arXiv:1401.6330*. [Online]. Available: https://arxiv.org/abs/1401.6330

[35] S. Liu, N. Yang, M. Li, and M. Zhou, ''A recursive recurrent neural network for statistical machine translation,'' in *Proc. ACL*, Jun. 2014, pp. 1491–1500.

[36] S. Hochreiter and J. Schmidhuber, ''Long short-term memory,'' *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[37] K. Cho, B. van Merrienboer, D. Bahdanau, and Y. Bengio, ''On the properties of neural machine translation: Encoder-decoder approaches,'' 2014, *arXiv:1409.1259*. [Online]. Available: https://arxiv.org/abs/1409.1259

[38] B. Pang and L. Lee, ''Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales,'' in *Proc. ACL*, Jun. 2005, pp. 115–124.

[39] M. D. Zeiler, ''ADADELTA: An adaptive learning rate method,'' 2012, *arXiv:1212.5701*. [Online]. Available: https://arxiv.org/abs/1212.5701

[40] T. Nakagawa, K. Inui, and S. Kurohashi, ''Dependency tree-based sentiment classification using CRFs with hidden variables,'' in *Proc. ACL*, Jun. 2010, pp. 786–794.

MOHD USAMA received the B.Sc. degree (Hons.) in statistics and the master's degree in computer science from Aligarh Muslim University (AMU), India. He is currently pursuing the Ph.D. degree with the School of Computer Science and Technology, Huazhong University of Science and Technology (HUST), China. His current research interests include deep learning, machine learning, natural language processing, and Internet of Medical Things. He is also a member of Quarterly Franklin at *London Journal of Research in Computer Science and Technology* (LJRCST) (London Journal Press, U.K.). He was a recipient of the HUST Academic Excellence Award, in 2019.

WENJING XIAO is currently pursuing the Ph.D. degree with the Embedded and Pervasive Computing (EPIC) Laboratory, School of Computer Science and Technology, Huazhong University of Science and Technology. Her current research interests include cognitive computing, Internet of Things, and cloud computing.

BELAL AHMAD received the B.Sc. degree (Hons.) from the Department of Statistics and Operation Research, Aligarh Muslim University (AMU), Aligarh, India, in 2009, and the MCA degree from the Department of Computer Science, AMU, in 2013. He is currently pursuing the Ph.D. degree with the School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan, China. His current research interests include network security and machine learning.

JIAFU WAN received the Ph.D. degree in mechatronic engineering from the South China University of Technology (SCUT), in 2008. From 2003 to 2014, he served as an Assistant Lecturer, a Lecturer, and an Associate Professor with the Guangdong Mechanical and Electrical College, Guangzhou, China. In 2010, he became an Associate Research Fellow and a Provincial Talent cultivated by Thousand-Hundred-Ten Program of Guangdong Province, China. From 2008 to 2012, he was a Postdoctoral Researcher in computer science and engineering with SCUT. He has been a Professor with the School of Mechanical and Automotive Engineering, SCUT, since 2015. His current research interests include cyber-physical systems, intelligent manufacturing, big data analytics, Industry 4.0, smart factory, and cloud robotics.

MOHAMMAD MEHEDI HASSAN (M'12) received the Ph.D. degree in computer engineering from Kyung Hee University, South Korea, in 2011. He is currently an Associate Professor with the Information Systems Department, College of Computer and Information Sciences (CCIS), King Saud University (KSU), Riyadh, Saudi Arabia. He received the Best Journal Paper Award from the IEEE Systems Journal, in 2018, the Best Paper Award from CloudComp Conference, China, in 2014, and the Excellence in Research Award from CCIS, KSU, in 2015 and 2016, respectively. He has published over 130 research articles in the journals and conferences of international repute. He has also played a role of the guest editor for several international ISI-indexed journals. His current research interests include cloud federation, multimedia cloud, sensor-cloud, Internet of Things, big data, mobile cloud, sensor network, publish/subscribe system, and recommender system.

ABDULHAMEED ALELAIWI received the Ph.D. degree in software engineering from the College of Engineering, Florida Institute of Technology, Melbourne, FL, USA, in 2002. He is currently an Associate Professor with the Software Engineering Department, College of Computer and Information Sciences (CCIS), King Saud University (KSU), Riyadh, Saudi Arabia. He is currently the Vice Dean of Research Chairs Program with KSU. He has authored and coauthored many publications. He has published over 70 research articles in the ISI-indexed journals of international repute. His current research interests include software testing analysis and design, cloud computing, multimedia, Internet of Things, big data, and mobile cloud. He has served as a Technical Program Committee Member in numerous reputed international conferences/workshops.