This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.2989428, IEEE Access

IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.Doi Number

Multi-task learning model based on Multi-scale CNN and LSTM for sentiment classification

Ning Jin[1], Jiaxian Wu[1], Xiang Ma[1], Ke Yan[1], (*Member, IEEE*), Yuchang Mo[2]

[1]Key Laboratory of Electromagnetic Wave Information Technology and Metrology of Zhejiang Province, College of Information Engineering, China Jiliang University, Hangzhou, China, 310018.
[2]Fujian Province University Key Laboratory of Computational Science, School of Mathematical Sciences, Huaqiao University, Quanzhou, China, 362021.

Corresponding author: Ke Yan (e-mail: yanke@cjlu.edu.cn).

This work was supported by Zhejiang Provincial Natural Science Foundation of China under Grant No. LY19F020016 and in part by the National Natural Science Foundation of China under Grant 61850410531, 61602431 and 61972156.

ABSTRACT Sentiment classification is an interesting and crucial research topic in the field of natural language processing (NLP). Data-driven methods, including machine learning and deep learning techniques, provide one direct and effective solution to solve the sentiment classification problem. However, the classification performance declines when the input includes review comments for multiple tasks. The most appropriate way of constructing a sentiment classification model under multi-tasking circumstances remains questionable in the related field. In this study, aiming at the multi-tasking sentiment classification problem, we propose a multi-task learning model based on a multi-scale convolutional neural network (CNN) and long short term memory (LSTM) for multi-task multi-scale sentiment classification (MTL-MSCNN-LSTM). The model comprehensively utilizes and properly handles global features and local features of different scales of text to model and represent sentences. The multi-task learning framework improves the encoder quality, simultaneously improving the results of emotion classification. Six different types of commodity review datasets were employed in the experiment. Using accuracy and F1-score as the metrics to evaluate the performance of the proposed model, comparing with methods such as single-task learning and LSTM encoder, the proposed MTL-MSCNN-LSTM model outperforms most of the existing methods.

INDEX TERMS Sentiment classification; Multi-Task Learning Model; Long Short Term Memory; Multi-Scale Convolutional Neural Network.

I. INTRODUCTION

In the information era, Internet technology has greatly influenced human lives [1]. There are review comments available for every service or product that exists in our daily life on the Internet. The sentiments included in the review comments become increasingly important for merchants to analyze and expand their business [2]. On one hand, user reviews reflect the quality and problems of products, which is important for the merchants. One the other hand, the number review reports can be tremendous and from different perspectives, it is difficult to conclude the review reports manually. As a traditional research topic in the field of natural language processing (NLP), sentiment classification aims at mining users' attitudes and perceptions (such as positive, negative) from text descriptions generated by user awareness [3]. Therefore, sentiment analysis is also called *opinion mining* and *point of view analysis*, which has been widely studied in e-commerce sites, social networks and other fields [4].

Through the ages, the methods of sentiment classification have undergone many changes, from the initial sentiment dictionary-based methods [5] to machine learning methods, such as Support Vector Machine (SVM), Naive Bayes (NB), decision tree, random forest, logistic regression, etc. [6, 7]. Although some machine learning methods can achieve good results on some tasks, due to the complexity of feature engineering [8], the effects of these methods are very dependent on feature representation, and difficult to achieve acceptable classification results [9, 10]. With the popularity of deep learning, many deep learning methods are applied to sentiment classification tasks [11, 12, 13]. Compared with machine learning methods, deep learning does not need to manually extract features. However, the robustness and generalization capability of the deep learning models largely depends on the amount of data available in the training phase.

With the development and maturity of deep learning methods, various structured deep learning networks have been proposed for different applications [14, 15, 16, 17]. In order to improve the effectiveness of the sentiment classification task, this paper proposes a model based on Multi-scale CNN and LSTM. It treats the review text of different commodities differently, while retaining the features that can be shared between each text, and combines the local features and temporal features (global features) of the text to improve the classification effect. The model consists of two main parts. The first part is a multi-task learning framework, which is used to capture shared features (that is, features independent of the type of commodity) and private features (that is, features closely related to the type of commodity) from reviews of different types of commodity. More specifically, it provides a private encoding scheme for each type of commodity comment, a shared encoding scheme for all comments, and a separate classifier for each commodity comment. The second part is the sentence encoder. Since the input of the sentiment classification model is represented by the combination of word embeddings [18], the model needs to extract the text features from it to obtain the sentence representation and then carry out the classification task. We use a multi-scale CNN network to extract the local features of different levels of sentences, and LSTM network to obtain the global features of sentences, and fuse the local features and global features of sentences to generate the sentence representation.

The main contributions of our study can be summarized as follows:

1. The multi-task learning (MTL) approach has been adopted to conduct sentiment analysis for the reviews under various types of commodities. Compared to the single task learning (STL) approach, the MTL uses an extra shared encoder to help improve the classification performance.
2. The shared encoder in MTL has been customized with multi-scale CNN (MSCNN) network combining the LSTM network and the Fusion net. The improved shared encoder, namely MSCNN-LSTM, largely improves the sentiment analysis in multi-task conditions.
3. A comprehensive comparative study has been conducted on the proposed method, compared with the existing sentiment analysis approaches available in the literature. The experimental results demonstrate that the proposed multi-task multi-scale CNN-LSTM (MTL-MSCNN-LSTM) framework outperforms all compared methods.

The remaining sections are arranged as follows: Section 2 introduces the related work. Section 3 introduces our method in detail. Section 4 shows the comparison experiment of this method. Section 5 summarizes the article and arranges the future work.

II. RELATED WORKS

As an important subject in the NLP field, sentiment classification always is an interesting and crucial research topic [6, 19, 20]. With the increasing popularity of deep learning, an increasing number of NLP tasks are based on word embeddings [21]. In 2013, Google opened word2vec, a tool for calculating word vectors [22, 23]. It can map words to low-dimensional vector space, reduce the cost of calculation, and make words to be related to each other rather than independent. It is a good distributed representation method [24], which overcomes the defect of a one-hot encoding discrete representation.

The purpose of the sentiment classification task is to obtain the sentiment polarity contained in the sentence, while the sentence information contained in the word is not complete, so the sentence encoder is needed to extract the features of the sentence, so as to generate the vector representation of the sentence. Convolutional neural network (CNN) was originally applied in the field of the image, but it has also been widely applied in the field of NLP in recent years [11, 25] and has achieved good results. The network structure of incomplete connection and weight sharing reduces the complexity of the model. The existence of a convolutional layer makes CNN good at capturing spatial local correlation, and the pooling layer in the network can greatly reduce the computation [26]. The appearance of word2vec makes it possible for CNN to apply to the text, enabling CNN to obtain local information in the text and conduct sentiment classification [27]. Zhou et al. proposed a goal-driven deep learning technique exploring user correlations in the task-oriented process [28]. Attardi and Santos proposed a char-CNN model, which is a two-layer convolutional network extracting relevant features from text sentences [29]. Similar works utilize CNN extensions to classify sentiment and completed multi-class classification tasks with good results [30, 31, 32].

It is not rigorous enough to define the sentiment polarity of a sentence by considering only the local features of the sentence. The global features of the sentence are an important indicator to determine the sentiment polarity. Recurrent neural networks (RNN) are suitable for extracting features of sequence-type data, and Long-Short-Term Memory Neural Network (LSTM) is improved based on RNN [33]. Although RNN is suitable for processing time-series information, it has the problem of gradient explosion or gradient disappearance [34, 35]. To solve this problem, LSTM introduces a gating mechanism into the traditional network structure of RNN to support the long-term dependence of sequence information [36, 37]. In natural language tasks, since the text itself has the property of sequence, it is very suitable to use LSTM for processing theoretically. Many works of literature have also proved that LSTM can effectively extract text semantic information [38, 39, 40]. Tang et al. used two LSTM networks to capture sentiment characteristics from the front and back context of an aspect and strung them together to

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.2989428, IEEE Access

predict the sentiment polarity of that aspect [41]. Ren et al. used LSTM combined with subject features to extract features of Twitter short texts and employed a bidirectional LSTM network for sentiment classification [42].

It is evident that the amount of data used in training affects the quality of the deep learning model, which is also true for the sentiment classification task. Multi-task learning (MTL) is an effective method to improve the performance of sentiment analysis, while there is not enough training data for any single task and related tasks' datasets are available [43]. There are various multi-task learning (MTL) architectures available and have been applied to different areas of NLP [44, 45, 46]. Liu et al. adopted the adversarial multi-tasking learning (AMTL) framework for multiple task text classification [47]. Luong et al. also used the multi-task learning method for machine translation tasks and achieved good results [48]. Yousif et al. completed the task of citation sentiment classification with a fully-shared multi-task learning framework [49]. Lu et al. utilized multi-task learning combined with VAE and achieved good results for multi-task sentiment classification [50].

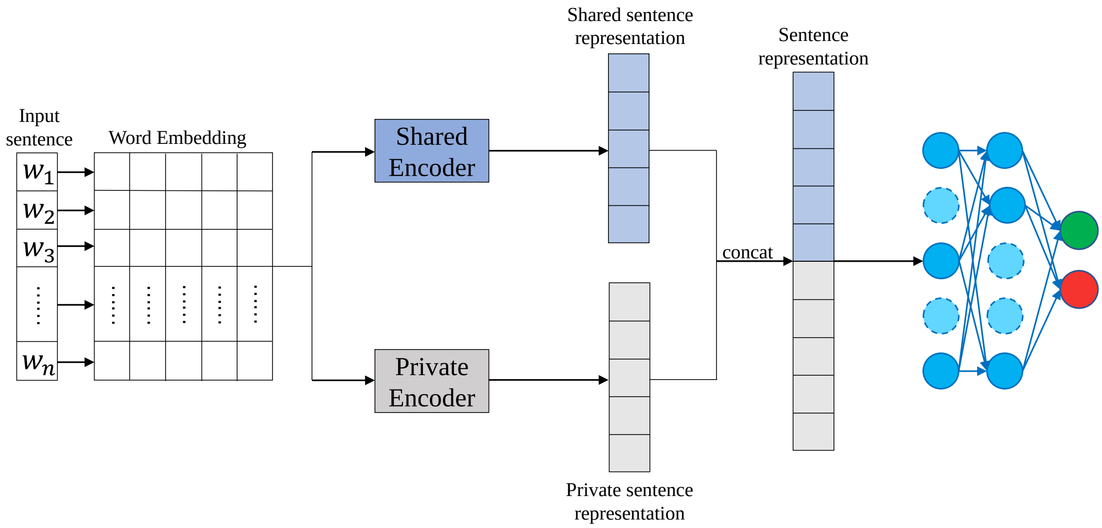

Fig. 1. The overall framework of the proposed MTL-MSCNN-LSTM model.

III. METHODOLOGY

We proposed MTL-MSCNN-LSTM to perform the sentiment analysis task of commodity review text and achieved good results. The overall structure of the model is shown in Fig. 1. We will describe the proposed deep learning framework in detail in the rest of this section.

A. MULTI-TASK LEARNING FRAMEWORK

Multi-task learning shows high efficiency in many natural language tasks. Due to the semantic differences between different types of commodity reviews, although the goal is to classify emotions, we think it is also applicable to the multi-task learning method. Our model adopts the adversarial multi-tasking learning (AMTL) framework [47] and improves the structure of its encoder. We will introduce the adversarial multi-tasking learning framework in detail in this part.

Adversarial multi-task learning is based on shared-private multi-task learning (SP-MTL), as shown in Fig. 2. For SP-MTL, the sentences in each task are assigned to two feature spaces to encode, shared space and a private space. Shared encoders are used to extract sentence features that are independent of tasks, while private encoders are used to extract sentence features that are more relevant to tasks. For sentence x_k in each task k, the shared feature S_k and private feature P_k of the sentence will be obtained through the two encoders, and the shared representation s_k and private representation p_k of the sentence will be obtained after pooling. Inspired by the generative adversarial network (GAN) [51], adversarial multi-task learning tries to make the shared space contain less information of private space, and adds a task discriminator after the shared encoder. It maps the shared representation of the sentence to a probability distribution to determine which task the sentence comes from, as shown in Equation (1). During the training process, the shared encoder prevents the discriminator from making a correct judgment on the task, ensuring that the shared representation part of the sentence is independent of the task, while the discriminator tries its best to identify which task the current sentence representation comes from. The discriminator loss L_D is included in the loss function, as shown in Equation (2).

$$D(s_k, \theta_D) = softmax(b + Us_k) \tag{1}$$

Where $U \in \mathbb{R}^{d\times d}$ is the parameter that can be learned, $b \in R^{d}$ is the bias, and θ_D is the parameter of the discriminator.

$$L_D = \min_{\theta_s}\left(\lambda \max_{\theta_D}\left(\sum_{k=1}^{K}\sum_{i=1}^{N_k} d_i^k \log\left[D\left(E\left(x_k\right)\right)\right]\right)\right) \quad (2)$$

Where d_i^k represents the true label of the current task and θ_s is the parameter of the shared encoder.

While ensuring that the shared features of a sentence include as few private features of the sentence as possible, it is also necessary to ensure that the private features of the sentence do not include those shared features as much as possible. To this end, the adversarial multi-task learning framework introduces orthogonal constraints to punish redundant features and guides the encoder to extract features in different aspects. Similarly, the loss function contains the orthogonal constraint loss term L_O , as shown in Equation (3).

$$L_O = \sum_{k=1}^{K}\left\| S_k^T P_k \right\|_F^2 \quad (3)$$

The main purpose of the multi-task learning method is to improve the performance of a single task with the help of other related tasks. The reason why we adopt the multi-task learning method as the base of our proposed model is that we take into account the semantic differences between different types of commodity review data. These differences might become noise to interfere with the training process. This will undoubtedly increase the learning difficulty for text encoders and classifiers. The purpose of introducing multi-task learning is to distinguish the review data of different types of commodities. A shared encoder was employed to learn the common features (shared features) between reviews from different tasks. The advantage of using the shared encoder is that since all the data samples pass through the shared encoder, the shared encoder ultimately learns enriched semantic features compared to individual encoders in single task leaning. The task discriminator set by the adversarial multi-task learning framework ensures that the learning content of the shared encoder is as independent as possible from the data source. On the other hand, we set up a private encoder for each type of commodity review data to learn the semantic features related to the source of the review (private feature). The existence of orthogonal constraints makes the learning content of the private encoder as relevant to the task as possible. Finally, we set up a classifier for each task for sentiment classification tasks. The input of the classifier is a sentence representation obtained by the shared feature and the private features of the review, and the output is the classification result.

It should be noted that in order to make the general structure of the model clearer, we did not give the details of the adversarial multi-task framework in Fig. 1, but chose to put it in this part for a detailed description. Similarly, the specific structure of the sentence encoder will be given in the next part.

B. SENTENCE ENCODER

The original adversarial multi-task learning framework uses LSTM encoder to encode sentences. We believe that this only considers the global features of the sentence, and ignores the local features of the sentence to some extent. To this end, we use multi-scale convolution combined with LSTM to encode sentences, using not only the global features of the sentence for classification but also the local features of different scales of the sentence. The structure of our encoder is shown in Fig. 3. As can be seen from Fig. 3, our proposed encoder extracts local text features of different scales from a multi-scale CNN encoder, and simultaneously extracts global features of the text from an LSTM encoder, and fuses the two features to generate a completely private or shared sentence representation.

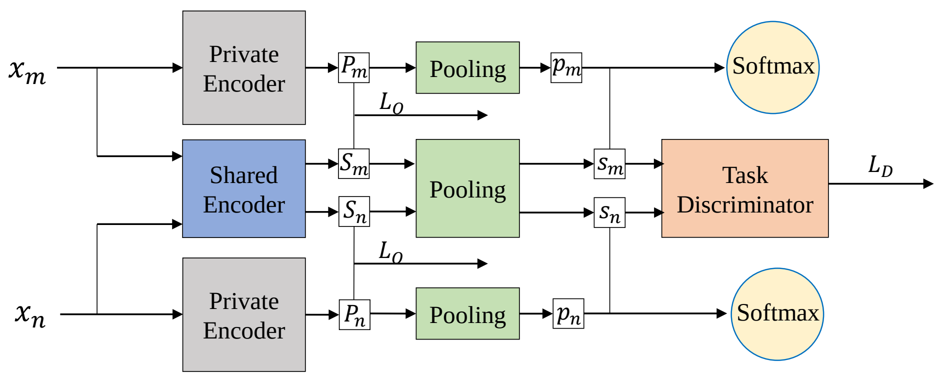

Fig. 2. The adversarial multi-task learning framework [47].

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.2989428, IEEE Access

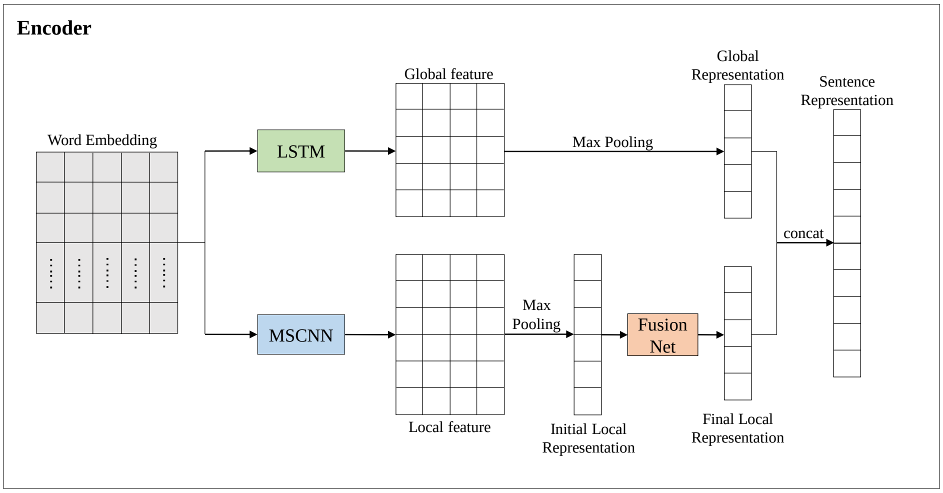

Fig. 3. Structure of the proposed MSCNN-LSTM encoder

1) LSTM EXTRACTS GLOBAL FEATURE

Our encoder uses LSTM to extract the global features of a sentence and get the global representation of the sentence. Compared with RNN, LSTM adds three gate control mechanisms, namely forget gate, input gate and output gate. At the same time, it introduces the selection of dependent information on cell state control, which effectively avoids the problem of gradient explosion and gradient disappearance. Its structure is shown in Fig. 4.

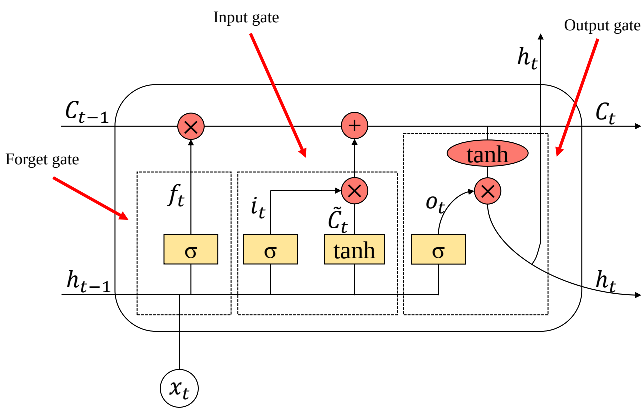

Fig. 4. Structure of the LSTM cell

The existence of the forget gate is to determine the degree of forgetting of the information flow before the current cell. The calculation is shown in Equation (4):

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{4}$$

The function of the input gate is to determine how much current information is added to the information flow. The calculation is shown in Equations (5) and (6):

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{5}$$

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \tag{6}$$

After the information passes through the input gate and the forget gate, the LSTM updates the cell state to calculate the output of the current LSTM cell and transfer it to the next LSTM cell. The calculation is shown in Equation (7):

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_{t-1} \tag{7}$$

The output gate combines the current input and cell state to determine the output of the current LSTM cell. The calculation is shown in Equations (8) and (9):

$$o_t = \sigma(W_o [h_{t-1}, x_t] + b_o) \tag{8}$$

$$h_t = o_t * \tanh(C_t) \tag{9}$$

The work of our encoder to extract the global features of the sentence and obtain global representation is shown in Fig. 5a. Each word in the sentence is represented by a word vector trained by word2vec. The first word vector is input to the first LSTM cell, and the output obtains the sentence feature of the current time-step and is passed to the next LSTM cell. The same is true for each subsequent word vector until all words pass through the LSTM cell and get their corresponding current sentence features. Since the input of the current time-step of the LSTM includes the output of the previous time-step, the output of the last time-step can be used as the global representation of the sentence, as shown in Fig. 5b.

2) MULTI-SCALE CNN EXTRACTS LOCAL FEATURE

CNN captures local features of sentences by scanning parts of the text through a convolution kernel. Considering the phrases, transitions, and other factors in the sentence, we choose 3, 4, and 5 sizes of convolution kernels to extract the local features of the sentence on different scales. The calculation is shown in Equation (10):

$$c_i^r = f(f \cdot V(w(i : i + r - 1)) + b) \tag{10}$$

Where F represents a convolution kernel of r * k size. In our experiment, the size of r is set to 3, 4, 5, and k is the dimension of the word vector. In our experiment, k is set to 256. f represents ReLU activation function. *V(w(i:i+r-1)* indicates that there are *r* word vectors from the *i*th word to the (*i* + *r-1*)th word in the sentence. c_i^r represents the ith

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.2989428, IEEE Access

local feature of the sentence extracted by the convolution kernel with width r. The work of our encoder to extract the local features of the sentence and get the local representations of the sentence is shown in Fig. 6.

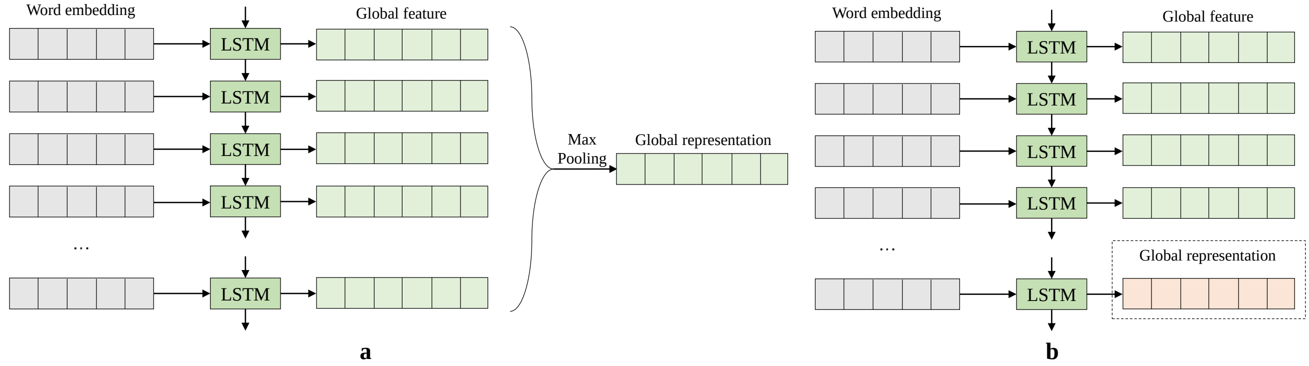

Fig. 5. LSTM extracts global features to generate global sentence representation

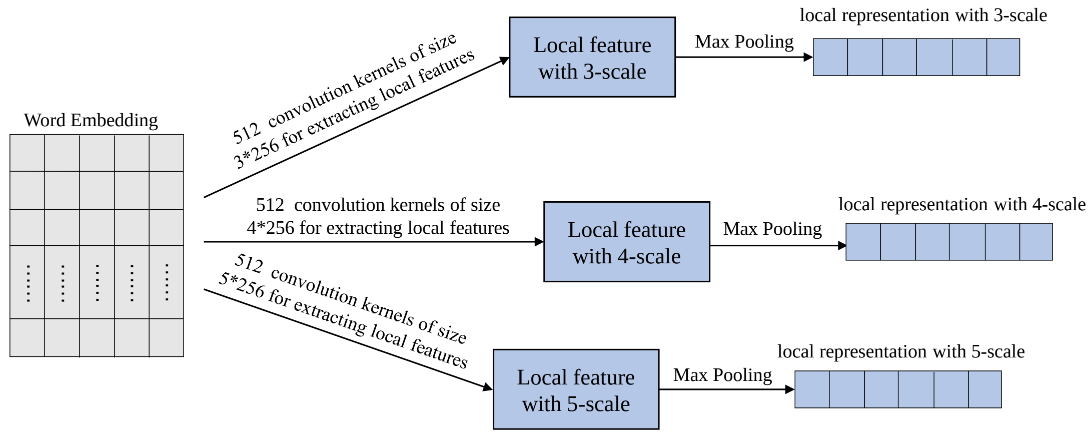

Fig. 6. MSCNN extracts multi-scale features to generate multi-scale local sentence representations

3) FUSION OF GLOBAL AND LOCAL REPRESENTATIONS OF SENTENCES

In the work described above, the word vector has passed through the LSTM encoder and the multi-scale CNN encoder to obtain the global and local representations of the sentence. Next, the two outputs (which were two sentences) were concatenated to get the complete private or shared sentence representation. But because these two sentence representations are not the same size. On the other hand, the output of the LSTM network finally passed the tanh activation function, and the feature size of each dimension was controlled in the interval (-1, 1), while the activation function used in the CNN network was the ReLU function. For the next classification task, the classifier can treat the sentence representations obtained by the two encoders equally. We added Fusion-Net after the sentence representations obtained by CNNs at three different scales to ensure that the local and global information contained in the sentence representations are in an "equal" state, as shown in Fig. 7. In Fig. 7, you can see that dropout was added after the initial local representation of the sentence, which is a measure to prevent overfitting. Similarly, we also set dropout in the LSTM cells. In order to verify the effectiveness of Fusion-Net, we set up comparative experiments in the experimental part to demonstrate this problem. The fused local representation will be concatenated with the global representation as to the final output of the sentence encoder.

Finally, the final sentence representation obtained from the output of the shared encoder and the output of the private encoder will be sent to the Softmax classifier after a number of fully connected layers (set to 3 in the experiment) for dimensionality reduction. We define the loss of classifier as Equation (11), and the final loss function is shown in Equation (12).

$$L(\hat{y}, y) = -\sum_{i=1}^{N}\sum_{j=1}^{C} y_i^j \log(\hat{y}_i^j) \tag{10}$$

$$L_C = \sum_{k=1}^{K} \alpha_k L(\hat{y}^{(k)}, y^{(k)}) \tag{11}$$

$$L = L_C + \beta L_D + \gamma L_O \tag{12}$$

Where λ and γ are hyperparameters, and our model is trained by SGD.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.2989428, IEEE Access

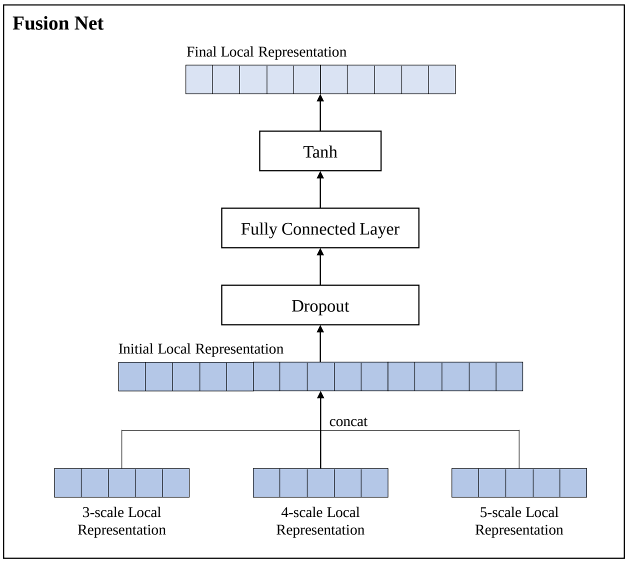

Fig. 7. Structure of Fusion-Net

IV. EXPERIMENT

A. DATASETS AND METRICS

As shown in Table I, the dataset I used in this experiment are apparel, camera, electronics, housewares, magazines, and sports commodity reviews. There are about 2,000 reviews for each commodity, for a total of about 12,000 reviews. This is a binary classification task. These review texts contain sentiment labels (Negative or Positive). These data were collected from raw data provided by Blitzer et al. [52]. We divided the training set, validation set, and test set to the ratio of 7: 1: 2, and ensured that the number of positive and negative samples in each set did not differ much. The dataset statistics are shown in Table II.

In addition, we collected four types of commodity review datasets of books, daily necessities, entertainment and media from raw data provided by Blitzer et al. [52] to participate in our experiments as dataset II. Dataset II has more data entries than dataset I, with a total of 23,742 entries. As with dataset I, each review text contains a sentiment label (Negative or Positive). We also divided the training set, validation set, and test set for dataset II, and ensured that the number of positive and negative samples in each set didn't differ much. Instances and statistics of dataset II are shown in Table III and Table IV.

In the experiment, we use the same evaluation criteria for each commodity review data set and each method, which are accuracy and F1-score. Our experiments were performed in the Pytorch environment.

TABLE I
INSTANCES OF THE TESTING DATASET I

Commodity type	Example	Label
Apparel	love hush puppies. they are comfortable and i can get a heel low enough for me to be comfortable	1
	my son wripped his right away trying to put it on his psp he said it was hard to get to the buttons to play	0
Camera	this is an elegant leather case for the canon elph cameras. it fits in the pocket easily. i 'm satisfied	1
	do not buy the case canon sells for this camera. it takes 20 seconds to take the camera out or put it in	0
Electronics	this mouse has never let me down. really works on most of the surfaces	1
	product arrived damaged. company (antonline) was unwilling to exchange for new item. would never buy from them again	0
Housewares	great for the money. feel like expensive sheet. very happy with it. just hope it is available with other colors	1
	these lights are junk , but the vendor i purchased them from is excellent	0
Magazines	go into high gear man! great business man! what you do not know you will know, if you buy this mag. read on man!	1
	I still have not received this magazine, what is taking so long! !	0
Sports	if you can stick to a workout routine, this home gym is perfect for you. after a bit of assembly , you 'll be able to do all your exercises at home	1
	the advertising was not clear as to what you're getting ; it does not include the stand for the ball assembly	0

TABLE II
DATASET I STATISTICS

Commodity type	Training set		Validation set		Test set		Total
	Positive	Negative	Positive	Negative	Positive	Negative	
Apparel	690	710	95	105	215	185	2000
Camera	706	692	99	100	194	206	1997
Electronics	705	693	97	103	198	202	1998
Housewares	706	694	102	98	192	208	2000

Magazines	682	688	101	99	217	183	1970
Sports	712	687	98	102	190	210	1999
Total	4201	4164	592	607	1206	1194	11964

TABLE III
INSTANCES OF THE TESTING DATASET II

Commodity type	Example	Label
Books	this is a resource used by all nps i have talked to . great addition to your library .	1
	this book made me feel like i was being patronized . they said the same thing over and over . i wish i had n't bought it	0
Daily necessities	item is exactly as pictured/described . cute design and good quality . adds a nice touch . highly recommend	1
	definately not a potty for boys . as with other reviewers , the shield does not stay on . looking for another simple and basic potty now : o	0
Entertainment	it was what i expected and it came in time for my grandsons birthday	1
	i ordered the doll because it said it will swim when put in water . when i recieved the doll it was just a barbie . it is not the swimming mermaid .	0
Media	this is a good 1st season set for the series of criminal intent . i strongly recommend this if you are a fan	1
	... but the movie was just plain stupid . do n't waste your time	0

TABLE IV
DATASET II STATISTICS

Commodity type	Training set		Validation set		Test set		Total
	Positive	Negative	Positive	Negative	Positive	Negative	
Books	2305	2257	292	308	379	421	5962
Daily necessities	1486	1609	198	201	212	188	3894
Entertainment	2219	2285	300	300	392	408	5904
Media	3007	2978	410	390	583	614	7982
Total	9017	9129	1200	1199	1566	1631	23742

B. COMPARED TO STL AND LSTM ENCODER

In order to verify that the proposed model structure meaningful, we compared the proposed model with the single-task method and the method using LSTM encoder. The experimental results of the dataset I are shown in Table V, and the experimental results of dataset II are shown in Table VI.

TABLE V
PERFORMANCE OF STL-LSTM, MTL-LSTM, STL-MSCNN-LSTM AND MTL-MSCNN-LSTM ON DATASET I

Model	STL-LSTM		MTL-LSTM		STL-MSCNN-LSTM		MTL-MSCNN-LSTM	
	Acc	F1-score	Acc	F1-score	Acc	F1-score	Acc	F1-score
Apparel	66.50%	0.6611	62.25%	0.5866	83.50%	0.8352	**86.25%**	**0.8627**
Camera	67.00%	0.6700	64.00%	0.6370	80.25%	0.8025	**88.50%**	**0.8850**
Electronic	66.75%	0.6674	70.50%	0.7031	80.25%	0.8024	**87.25%**	**0.8725**
Housewares	68.25%	0.6826	66.25%	0.6626	83.75%	0.8376	**85.50%**	**0.8550**
Magazines	70.50%	0.6969	60.00%	0.5629	82.75%	0.8267	**89.25%**	**0.8927**
Sports	67.50%	0.6751	60.25%	0.5542	84.75%	0.8474	**87.25%**	**0.8726**
Average	67.75%	0.6755	63.88%	0.6177	82.54%	0.8253	**87.33%**	**0.8734**

TABLE VI
PERFORMANCE OF STL-LSTM, MTL-LSTM, STL-MSCNN-LSTM AND MTL-MSCNN-LSTM ON DATASET II

Model	STL-LSTM		MTL-LSTM		STL-MSCNN-LSTM		MTL-MSCNN-LSTM	
	Acc	F1-score	Acc	F1-score	Acc	F1-score	Acc	F1-score
Books	74.50%	0.7446	82.88%	0.8288	74.50%	0.7446	**82.88%**	**0.8288**
Daily necessities	82.25%	0.8226	88.25%	0.8826	82.25%	0.8226	**88.25%**	**0.8826**

Entertainment	82.75%	0.8275	86.63%	0.8660	82.75%	0.8275	**86.63%**	**0.8660**
Media	75.36%	0.7514	86.72%	0.8672	75.36%	0.7514	**86.72%**	**0.8672**
Average	77.86%	0.7777	85.93%	0.8592	77.86%	0.7777	**85.93%**	**0.8592**

Observing Tables V and VI, we can see that our proposed MTL-MSCNN-LSTM model performs optimally on both datasets I and dataset II. This is the result of the combined action of both the multi-task learning framework and MSCNN-LSTM encoder. The multi-task learning model allows the model to learn common knowledge from related corpora and apply it to the training process of shared encoders. At the same time, the private encoder of each commodity review data and its independent classifier reduce the noise impact of different texts. On the other hand, the MSCNN-LSTM encoder we designed not only considers the global features of the sentence, but also considers the local features on different scales, and combines these two features to classify the sentiment of the reviews. In Tables V and VI, the experimental results of MSCNN-LSTM are better than LSTM in both STL and MTL frameworks. Only on the Media dataset, the performance of the MSCNN-LSTM encoder is slightly worse than the LSTM encoder, and it is still in the case of STL. This proves that its ability to extract text features is obviously more powerful than LSTM encoders, and has good generality. The MSCNN-LSTM encoder we designed has undoubtedly contributed to the improvement of sentiment classification tasks.

However, in the experiment of the dataset I, when LSTM is used as the encoder, the performance of STL and MTL is not ideal, and some results even show that the effect of STL is better than MTL. Generally speaking, MTL is better than STL, but when the encoder is not optimal, the performance of MTL may appear to be less stable. Especially in the case where the performance of both is not ideal, even if STL is slightly better, it doesn't make much sense. Of course, it is also related to the size of the data set to some extent. When our encoder is adjusted to MSCNN-LSTM, the performance of MTL is significantly better than STL. This also shows from another aspect that the MSCNN-LSTM encoder is better than the LSTM encoder, and it has some improvements in the performance of the MTL framework.

C. MODEL SELF-COMPARISON

As mentioned in Section 3, the Fusion-Net was added after the local representation of the multi-scale CNN encoded sentence for the classifier treating the global and local representation of the sentence equally, and consequently to avoid the problem of gradient disappearance in the later fully connected layer. Based on this problem, we put a comparative experiment on our proposed MTL-MSCNN-LSTM model. Table VII and Table VIII reflect the fact that the Fusion-Net we added further improves the classification effect of the MTL-MSCNN-LSTM model.

After the encoder extracts the sentence features, these features pass through the pooling layer to produce a partial representation of the sentence. Due to the characteristics of the LSTM encoder structure, it is determined that the subsequent pooling layer needs to use the Max-pooling method. For the pooling method of the pooling layer after the MSCNN encoder, we conducted a comparison experiment on the Max-pooling and the Mean-pooling, as shown in Table IX and Table X.

TABLE VII
PERFORMANCE IMPROVEMENT USING FUSION-NET FOR MTL-MSCNN-LSTM MODEL USED ON DATASET I

Fusion-Net	without Fusion-Net		with Fusion-Net	
	Acc	F1-score	Acc	F1-score
Apparel	86.25%	0.8627	86.25%	0.8627
Camera	86.75%	0.8675	**88.50%**	**0.8850**
Electronic	83.50%	0.8347	**87.25%**	**0.8725**
Housewares	83.50%	0.8350	**85.50%**	**0.8550**
Magazines	88.75%	0.8877	**89.25%**	**0.8927**
Sports	84.75%	0.8475	**87.25%**	**0.8726**
Average	85.89%	0.8559	**87.33%**	**0.8734**

TABLE VIII
PERFORMANCE IMPROVEMENT USING FUSION-NET FOR MTL-MSCNN-LSTM MODEL USED ON DATASET II

Fusion-Net	without Fusion-Net		with Fusion-Net	
	Acc	F1-score	Acc	F1-score

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.2989428, IEEE Access

Books	78.38%	0.7822	**82.88%**	**0.8288**
Daily necessities	86.00%	0.8601	**88.25%**	**0.8826**
Entertainment	84.62%	0.8459	**86.63%**	**0.8660**
Media	83.96%	0.8390	**86.72%**	**0.8672**
Average	82.98%	0.8292	**85.93%**	**0.8592**

TABLE IX

RESULTS OF USING DIFFERENT POOLING TYPES FOR MTL-MSCNN-LSTM MODEL USED ON DATASET I

Pooling type	Mean Pooling		Max Pooling	
	Acc	F1-score	Acc	F1-score
Apparel	84.75%	0.8477	**86.25%**	**0.8627**
Camera	87.50%	0.8748	**88.50%**	**0.8850**
Electronic	83.75%	0.8375	**87.25%**	**0.8725**
Housewares	84.00%	0.8399	**85.50%**	**0.8550**
Magazines	86.75%	0.8676	**89.25%**	**0.8927**
Sports	86.00%	0.8600	**87.25%**	**0.8726**
Average	85.46%	0.8546	**87.33%**	**0.8734**

TABLE X

RESULTS OF USING DIFFERENT POOLING TYPES FOR MTL-MSCNN-LSTM MODEL USED ON DATASET II

Pooling type	Mean Pooling		Max Pooling	
	Acc	F1-score	Acc	F1-score
Books	80.88%	0.8087	**82.88%**	**0.8288**
Daily necessities	85.75%	0.8569	**88.25%**	**0.8826**
Entertainment	83.00%	0.8298	**86.63%**	**0.8660**
Media	84.63%	0.8462	**86.72%**	**0.8672**
Average	83.42%	0.8341	**85.93%**	**0.8592**

D. COMPARISON WITH OTHER SENTIMENT CLASSIFICATION METHODS

In order to verify the effectiveness of our proposed MTL-MSCNN-LSTM sentiment classification model, we compared this method with existing sentiment classification methods, including logistic regression [53], random forest [54], Naïve Bayesian [53], k-nearest-neighbors (KNN) [54], support vector machine (SVM) [54], extreme gradient boosting (XGBoost) [55], gradient boosting decision tree [56]. The experimental results are shown in Table XI and Table XII.

TABLE XI

COMPARISON WITH OTHER SENTIMENT CLASSIFICATION METHODS, DATASETS I AND II ARE SEPARATED USING A HORIZONTAL LINE (EVALUATION INDEX: ACCURACY (%))

Method	Logistics Regression	Random Forest	Naive Bayesian	KNN	SVM	XGBoost	Gradient Boosting Decision Tree	MTL-MSCNN-LSTM
Apparel	84.50%	81.00%	82.25%	65.00%	60.00%	79.75%	79.75%	**86.25%**
Camera	81.75%	81.25%	83.50%	59.75%	58.50%	77.50%	80.50%	**88.50%**
Electronic	81.75%	75.00%	67.25%	60.25%	60.00%	80.25%	76.75%	**87.25%**
Housewares	77.50%	76.75%	78.50%	58.75%	59.50%	77.00%	77.75%	**85.50%**
Magazines	83.50%	83.00%	82.75%	51.75%	78.00%	82.50%	83.50%	**89.25%**
Sports	81.75%	82.00%	75.00%	56.50%	52.25%	79.50%	79.00%	**87.25%**
Books	78.50%	78.13%	78.63%	57.63%	70.38%	75.13%	75.63%	**82.88%**
Daily necessities	84.75%	79.25%	68.75%	61.50%	53.00%	82.25%	79.75%	**88.25%**
Entertainment	83.63%	81.00%	81.88%	63.88%	57.63%	81.00%	79.63%	**86.63%**
Media	82.46%	79.95%	80.20%	57.56%	59.40%	78.61%	77.28%	**86.72%**

TABLE XII
COMPARISON WITH OTHER SENTIMENT CLASSIFICATION METHODS, DATASETS I AND II ARE SEPARATED USING A HORIZONTAL LINE (EVALUATION INDEX: F1-SCORE)

Method	Logistics Regression	Random Forest	Naive Bayesian	KNN	SVM	XGBoost	Gradient Boosting Decision Tree	MTL-MSCNN-LSTM
Apparel	0.8452	0.8100	0.8225	0.6442	0.5691	0.7977	0.7978	**0.8627**
Camera	0.8175	0.8123	0.8350	0.5768	0.5072	0.7775	0.8049	**0.8850**
Electronic	0.8175	0.7492	0.6357	0.5954	0.5603	0.8121	0.7674	**0.8725**
Housewares	0.7750	0.7667	0.7817	0.5587	0.5466	0.7697	0.7774	**0.8550**
Magazines	0.8352	0.8303	0.8275	0.4422	0.7676	0.8253	0.8353	**0.8927**
Sports	0.8176	0.8201	0.7436	0.5198	0.4096	0.7951	0.7901	**0.8726**
Books	0.7851	0.7814	0.7856	0.5334	0.6881	0.7513	0.7564	**0.8288**
Daily necessities	0.8476	0.7926	0.6647	0.6075	0.4259	0.8227	0.7977	**0.8826**
Entertainment	0.8361	0.8100	0.8184	0.6307	0.4834	0.8099	0.7956	**0.8660**
Media	0.8246	0.7995	0.8020	0.5720	0.5086	0.7858	0.7727	**0.8672**

According to Tables XI and XII, the proposed MTL-MSCNN-LSTM model outperforms all compared state-of-art classification methods, for both classification accuracy rates and F1-score values. The comparative results that are shown in Tables XI and XII further prove that the research we conducted is meaningful and our proposed method is effective and robust.

V. CONCLUSION, LIMITATION and FUTURE WORKS

In this paper, we propose a multi-task learning method for sentiment analysis of different types of commodity reviews. Taking multi-task learning as the framework, we use LSTM and multi-scale CNN to jointly perform sentence encoding, taking into account the global and local features of the text. In addition, the network structure of the CNN encoder has been customized to enhance the classification performance. We implemented model training, validation, and testing in the Pytorch environment. The experimental results show that the sentiment classification results of our proposed model are superior over existing state-of-art methods. This is because this model considers more comprehensive text features and properly handles them. In addition, the adversarial multi-task learning approach improves the encoding quality of the encoder.

It is noted that multiple deep learning techniques, including CNN and LSTM, were used in the proposed framework. Therefore, the efficiency of the proposed algorithm is not comparable with the state-of-art methods, such as Native Bayes, SVM and KNN. However, we insist that the sentiment classification accuracy has been largely improved. And the time complexity of the proposed method is not the main concern, since the sentiment analysis can always be performed offline

As one of the future works, the encoder on the existing basis will be improved to further improve the results of sentiment classification for multi-task learning. Moreover, we intend to perform some multi-class sentiment analysis for other related NLP tasks.

REFERENCES

[1] Xu, Yang, et al. "A Blockchain-based Non-Repudiation Network Computing Service Scheme for Industrial IoT." *IEEE Transactions on Industrial Informatics* (2019).

[2] O'Connor, Brendan, et al. "From tweets to polls: Linking text sentiment to public opinion time series." *Fourth International AAAI Conference on Weblogs and Social Media*. 2010.

[3] Zhou, Xiaokang, et al. "Social Recommendation With Large-Scale Group Decision-Making for Cyber-Enabled Online Service." *IEEE Transactions on Computational Social Systems* 6.5 (2019): 1073-1082.

[4] Zhou, Xiaokang, et al. "Multi-Modality Behavioral Influence Analysis for Personalized Recommendations in Health Social Media Environment." *IEEE Transactions on Computational Social Systems* 6.5 (2019): 888-897.

[5] Teng, Zhiyang, Duy Tin Vo, and Yue Zhang. "Context-sensitive lexicon features for neural sentiment analysis." *Proceedings of the 2016 conference on empirical methods in natural language processing*. 2016.

[6] Yan, Ke, et al. "Semi-supervised learning for early detection and diagnosis of various air handling unit faults." *Energy and Buildings* 181 (2018): 75-83.

[7] Lu, Huijuan, et al. "A cost-sensitive rotation forest algorithm for gene expression data classification." *Neurocomputing* 228 (2017): 270-276.

[8] Yan, Ke, et al. "Cost-sensitive and sequential feature selection for chiller fault detection and diagnosis." *International Journal of Refrigeration* 86 (2018): 401-409.

[9] Xia, H., Yang, Y., Pan, X., Zhang, Z., & An, W. (2019).

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.2989428, IEEE Access

Sentiment analysis for online reviews using conditional random fields and support vector machines. Electronic Commerce Research, 1-18.

[10] Qu, Lizhen, Georgiana Ifrim, and Gerhard Weikum. "The bag-of-opinions method for review rating prediction from sparse text patterns." *Proceedings of the 23rd international conference on computational linguistics*. Association for Computational Linguistics, 2010.

[11] Gómez-Adorno, H., Fuentes-Alba, R., Markov, I., Sidorov, G., & Gelbukh, A. (2019). A convolutional neural network approach for gender and language variety identification. *Journal of Intelligent & Fuzzy Systems*, *36*(5), 4845-4855.

[12] Zhang, D., Hong, M., Zou, L., Han, F., He, F., Tu, Z., & Ren, Y. (2019). Attention Pooling-Based Bidirectional Gated Recurrent Units Model for Sentimental Classification. *International Journal of Computational Intelligence Systems*, *12*(2), 723-732.

[13] Yan, Xiaodan, et al. "A Method of Information Protection for Collaborative Deep Learning under GAN Model Attack." *IEEE/ACM Transactions on Computational Biology and Bioinformatics* (2019).

[14] Yan, Ke, et al. "Multi-Step short-term power consumption forecasting with a hybrid deep learning strategy." *Energies*11.11 (2018): 3089.

[15] Yan, K., Chong, A., & Mo, Y. (2020). Generative adversarial network for fault detection diagnosis of chillers. Building and Environment, 172.

[16] Yan, K., Shen, H., Wang, L., Zhou, H., Xu, M., & Mo, Y. (2020). Short-Term Solar Irradiance Forecasting Based on a Hybrid Deep Learning Methodology. *Information*, *11*(1), 32.

[17] Yan, Ke, et al. "A Hybrid LSTM Neural Network for Energy Consumption Forecasting of Individual Households." *IEEE Access* 7 (2019): 157633-157642.

[18] Serrano-Guerrero, Jesus, et al. "Sentiment analysis: A review and comparative analysis of web services." *Information Sciences* 311 (2015): 18-38.

[19] Liu, Bing. "Sentiment analysis and opinion mining." *Synthesis lectures on human language technologies* 5.1 (2012): 1-167.

[20] Bengio, Yoshua, et al. "A neural probabilistic language model." *Journal of machine learning research* 3.Feb (2003): 1137-1155.

[21] Mikolov, Tomas, et al. "Efficient estimation of word representations in vector space." *arXiv preprint arXiv:1301.3781* (2013).

[22] Nyberg, Katariina, et al. "Document classification utilising ontologies and relations between documents." *Proceedings of the Eighth Workshop on Mining and Learning with Graphs*. ACM, 2010.

[23] Mikolov, Tomas, et al. "Distributed representations of words and phrases and their compositionality." *Advances in neural information processing systems*. 2013.

[24] Zhou, Xiaokang, Bo Wu, and Qun Jin. "User role identification based on social behavior and networking analysis for information dissemination." *Future Generation Computer Systems* (2017).

[25] Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. "Imagenet classification with deep convolutional neural networks." *Advances in neural information processing systems*. 2012.

[26] Kim, Yoon. "Convolutional neural networks for sentence classification." *arXiv preprint arXiv:1408.5882* (2014).

[27] Pang, Bo, and Lillian Lee. "Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales." *Proceedings of the 43rd annual meeting on association for computational linguistics*. Association for Computational Linguistics, 2005.

[28] Zhou, Xiaokang, et al. "Discovery of action patterns and user correlations in task-oriented processes for goal-driven learning recommendation." *IEEE Transactions on Learning Technologies* 7.3 (2014): 231-245.

[29] Attardi, Giuseppe and Daniele Sartiano. "UniPI at SemEval-2016 Task 4: Convolutional neural networks for sentiment classification." *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*. 2016.

[30] Dong Han, Chunhua Wang, and Min Xiao. "Improved CNN based on sentence-level supervised learning for short text classification." *Computer Engineering and Design* 2019, 40(01):264-268+292. (in Chinese)

[31] Yanmei, Liu, and Chen Yuda. "Research on Chinese Micro-Blog Sentiment Analysis Based on Deep Learning." *2015 8th International Symposium on Computational Intelligence and Design (ISCID)*. Vol. 1. IEEE, 2015.

[32] Hochreiter, Sepp, and Jürgen Schmidhuber. "Long short-term memory." *Neural computation* 9.8 (1997): 1735-1780.

[33] Hochreiter, Sepp. "The vanishing gradient problem during learning recurrent neural nets and problem solutions." *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* 6.02 (1998): 107-116.

[34] Pascanu, Razvan, Tomas Mikolov, and Yoshua Bengio. "On the difficulty of training recurrent neural networks." *International conference on machine learning*. 2013.

[35] Al-Smadi, M., Talafha, B., Al-Ayyoub, M., & Jararweh, Y. (2019). Using long short-term memory deep neural networks for aspect-based sentiment analysis of Arabic reviews. *International Journal of Machine Learning and Cybernetics*, *10*(8), 2163-2175.

[36] Graves, Alex, Navdeep Jaitly, and Abdel-rahman Mohamed. "Hybrid speech recognition with deep bidirectional LSTM." *2013 IEEE workshop on automatic speech recognition and understanding*. IEEE, 2013.

[37] Zhang, S., Xu, X., Pang, Y., & Han, J. (2019). Multi-layer Attention Based CNN for Target-Dependent

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.2989428, IEEE Access

Sentiment Classification. Neural Processing Letters, 1-15.
[38] Wen, S., Wei, H., Yang, Y., Guo, Z., Zeng, Z., Huang, T., & Chen, Y. (2019). Memristive LSTM network for sentiment analysis. IEEE Transactions on Systems, Man, and Cybernetics: Systems.
[39] Zhang, Meishan, Yue Zhang, and Duy-Tin Vo. "Gated neural networks for targeted sentiment analysis." *Thirtieth AAAI Conference on Artificial Intelligence*. 2016.
[40] Wang, J., Yu, L. C., Lai, K. R., & Zhang, X. (2019). Tree-Structured Regional CNN-LSTM Model for Dimensional Sentiment Analysis. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 28, 581-591.
[41] Tang, Duyu, et al. "Target-dependent sentiment classification with long short term memory." *arXiv preprint arXiv:1512.01100* (2015).
[42] Mian Ren, and Gang Gan. " Sentiment analysis of text based on bi-directional long short-term memory model." *Computer Engineering and Design* 2018, v.39; No.379(07):272-276. (in Chinese)
[43] Hessel, M., Soyer, H., Espeholt, L., Czarnecki, W., Schmitt, S., & van Hasselt, H. (2019, July). Multi-task deep reinforcement learning with popart. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 33, pp. 3796-3803).
[44] Liu, Pengfei, Xipeng Qiu, and Xuanjing Huang. "Deep multi-task learning with shared memory." *arXiv preprint arXiv:1609.07222* (2016).
[45] Liu, Pengfei, Xipeng Qiu, and Xuanjing Huang. "Recurrent neural network for text classification with multi-task learning." *arXiv preprint arXiv:1605.05101* (2016).
[46] Zhang, Honglun, et al. "A generalized recurrent neural architecture for text classification with multi-task learning." *arXiv preprint arXiv:1707.02892* (2017).
[47] Liu, Pengfei, Xipeng Qiu, and Xuanjing Huang. "Adversarial multi-task learning for text classification." *arXiv preprint arXiv:1704.05742* (2017).
[48] Luong, Minh-Thang, et al. "Multi-task sequence to sequence learning." *arXiv preprint arXiv:1511.06114* (2015).
[49] Yousif, Abdallah, et al. "Multi-task learning model based on recurrent convolutional neural networks for citation sentiment and purpose classification." *Neurocomputing* 335 (2019): 195-205.
[50] Lu, Guangquan, et al. "Multi-task learning using variational auto-encoder for sentiment classification." *Pattern Recognition Letters* (2018).
[51] Goodfellow, Ian, et al. "Generative adversarial nets." *Advances in neural information processing systems*. 2014.
[52] Blitzer, John, Mark Dredze, and Fernando Pereira. "Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification." *Proceedings of the 45th annual meeting of the association of computational linguistics*. 2007.
[53] Prabhat, A., & Khullar, V. (2017, January). Sentiment classification on big data using Naïve Bayes and logistic regression. In *2017 International Conference on Computer Communication and Informatics (ICCCI)* (pp. 1-5). IEEE.
[54] Yan, K., Huang, J., Shen, W., & Ji, Z. (2020). Unsupervised learning for fault detection and diagnosis of air handling units. Energy and Buildings, 210, 109689.
[55] Yan, K., Dai, Y., Xu, M., & Mo, Y. (2020). Tunnel Surface Settlement Forecasting with Ensemble Learning. Sustainability, 12(1), 232.
[56] Hew, K. F., Hu, X., Qiao, C., & Tang, Y. (2020). What predicts student satisfaction with MOOCs: a gradient boosting trees supervised machine learning and sentiment analysis approach. Computers & Education, 145, 103724.