Global-Local Mutual Attention Model for Text Classification

Qianli Ma, *Member, IEEE*, Liuhong Yu, Shuai Tian, Enhuan Chen, and Wing W. Y. Ng, *Senior Member, IEEE*

***Abstract*—Text classification is a central field of inquiry in natural language processing (NLP). Although some models learn local semantic features and global long-term dependencies simultaneously, they simply combine them through concatenation either in a cascade way or in parallel while mutual effects between them are ignored. In this paper, we propose the Global-Local Mutual Attention (GLMA) model for text classification problems, which introduces a mutual attention mechanism for mutual learning between local semantic features and global long-term dependencies. The mutual attention mechanism consists of a Local-Guided Global-Attention (LGGA) and a Global-Guided Local-Attention (GGLA). The LGGA allows to assign weights and combine global long-term dependencies of word positions that are semantic related. It captures combined semantics and alleviates the gradient vanishing problem. The GGLA automatically assigns more weights to relevant local semantic features, which captures key local semantic information and filters both noises and irrelevant words/phrases. Furthermore, a weighted-over-time pooling operation is developed to aggregate the most informative and discriminative features for classification. Extensive experiments demonstrate that our model obtains the state-of-the-art performance on seven benchmark datasets and sixteen Amazon product reviews datasets. Both the result analysis and the mutual attention weights visualization further demonstrate the effectiveness of the proposed model.**

***Index Terms*—Recurrent Neural Network (RNN), Convolutional Neural Network (CNN), Mutual Attention Mechanism, Weighted-over-time Pooling, Text Classification.**

I. Introduction

TEXT classification [1] is a fundamental and traditional task in natural language processing (NLP) which attracts considerable attention from researchers. Text classification is a task in NLP where a single or multiple predefined category(ies) is/are assigned to a sequence of texts. The aim of text classification is learning sequence representation for sentiment analysis [2], question classification [3], and topic classification [4].

Modeling of global long-term dependencies and local semantic features are two ways widely used for text classification. Recently, Recurrent Neural Networks (RNNs) are widely applied to capture global context features and model the long-term dependency of text sequences. However, RNNs tend to model the global long-term dependency of the entire text sequences, so that they may ignore some important local semantic information for correct classification [5]. On the other hand, Convolutional Neural Networks (CNNs) are often used to capture discriminative local semantic features for text classification [6]–[9], while they neglect global long-term dependency of text sequences. However, it has been proved that global contexts provide useful topical information [10], and several studies in psychology have also shown that global contexts help language comprehension [11]. Hence, it is necessary to model global long-term dependencies and local semantic features simultaneously for text classification tasks.

Therefore, researches try to model global long-term dependencies and local semantic features simultaneously and connect the two parts in a cascaded way [12]–[16]. However, the model will be deeper in this manner which may aggravate gradient vanishing problems and will have troubles in the training. Some other researches combine the global long-term dependency and local semantic features through concatenation [17]. However, concatenated features may lead to redundancies [18]. Moreover, they cannot capture the combined semantics of text sequences because RNNs are sequential models and the global long-term dependencies are accumulated sequentially [19]. On the other hand, not all the local semantic features extracted by CNN are useful because they often contain noises or redundancies.

To address these issues, we propose a novel model named Global-Local Mutual Attention (GLMA) model for text classification. Firstly, GLMA extracts long-term dependencies by a bidirectional long-short term memory (bi-LSTM) and local semantic features by a multi-scale CNN. After that, a GLMA mechanism consisting of local-guided global-attention (LGGA) and global-guided local-attention (GGLA) is designed.

Manuscript received March 24, 2019; revised July 20, 2019 and August 21, 2019; accepted September 11, 2019. Date of publication September 18, 2019; date of current version September 30, 2019. The work was supported in part by the National Natural Science Foundation of China under Grants 61502174, 61872148, 61876066, and 61572201, the Natural Science Foundation of Guangdong Province under Grants 2017A030313355 and 2017A030313358, in part the Guangzhou Science and Technology Planning Project under Grants 201704030051, 201902010020, and 201804010245, and in part by the Key R&D Program of Guangdong Province 2018B010107002. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Imed Zitouni. *(Corresponding author: Wing W. Y. Ng.)*

Q. Ma is with the School of Computer Science and Engineering, South China University of Technology, Guangzhou 510006, China, and also with the Guangdong Provincial Key Lab of Computational Intelligence and Cyberspace Information, Guangzhou 510006, China (e-mail: qianlima@scut.edu.cn).

L. Yu, S. Tian, and E. Chen are with the School of Computer Science and Engineering, South China University of Technology, Guangzhou 510006, China (e-mail: yu.liuhong@foxmail.com; 439797373@qq.com; ceh930603@gmail.com).

W. W. Y. Ng is with the Guangdong Provincial Key Lab of Computational Intelligence and Cyberspace Information, School of Computer Science and Engineering, South China University of Technology, Guangzhou 510006, China (e-mail: wingng@ieee.org).

This paper has supplementary downloadable material available at http://ieeexplore.ieee.org., provided by the authors. Contact [Qianli Ma qianlima@scut.edu.cn] for further questions about this work.

Digital Object Identifier 10.1109/TASLP.2019.2942160

LGGA treats local semantic features as guiding information and global long-term dependencies as target information. For the local semantic feature at each word position, LGGA learns to weigh global long-term dependencies of different positions with learned weights. The learned weights are significantly large for global long-term dependencies that are semantically correlated. Thus, the global long-term dependencies are captured and the combined semantics are obtained. The learned weights also create direct connections to global long-term dependencies (hidden states of bi-LSTM) and act as the weighted skip connections [20] to alleviate the gradient vanishing problem by shortening the path of gradient propagation in bi-LSTM. On the other hand, GGLA uses global long-term dependencies as guiding information and local semantic features as target information. For each word position of global long-term dependency, GGLA automatically assigns larger weights to relevant local semantic features. Therefore, the GLMA mechanism can model mutual effects between each position of inputs. Furthermore, a weighted-over-time pooling is proposed to aggregate the most informative global-local features. Finally, the global-local features are fed into a fully connected layer and a softmax layer to obtain classification results.

The main contributions can be summarized as follows:

1) We propose a novel Global-Local Mutual Attention (GLMA) model with a mutual attention mechanism consisting of a LGGA and a GGLA. The LGGA learns to weight long-term dependencies with learned weights which capture combined semantics. It creates direct connections to alleviate the gradient vanishing problem. GGLA automatically assigns larger weights to relevant local semantic features to capture key local semantic features.
2) A weighted-over-time pooling operation is proposed to aggregate the most informative and discriminative features. Our experiments prove that weighted-over-time pooling is more effective than max-over-time and average-over-time pooling operation.
3) GLMA is extensively evaluated on seven benchmark text classification datasets and sixteen datasets from Amazon product reviews. Experiment results demonstrate GLMA outperforms existing models. Visualization results of mutual attention weights further prove the effectiveness of our model.

The rest of this paper is organized as follows. Section II discusses related work. Section III formally describes the structure of the proposed model and the important components in detail. The results of the proposed method on text classification, the experimental analysis, qualitative analysis, attention weights visualization, and gradient analysis are presented in Section IV. Finally, we conclude the study and suggest future work in Section V.

II. Related Work

In this paper, we will focus on introducing the deep learning approaches for text classification. Approaches for modeling local semantic features and global long-term dependencies fall into three categories, including CNN-based, RNN-based, and combined approaches. Moreover, we will also introduce the relevant attention mechanism.

A. CNN-Based Approaches

CNN has a good performance on extracting local semantic features at different positions of a text sequence. A multi-channel CNN with two sets of word vectors, static vectors, and fine-tuned vectors is proposed for text classification [6]. To capture both short and long-range relations over a sentence, The Dynamic Convolutional Neural Network (DCNN) which has a global pooling operation with different pooling rate is proposed [7]. Shallow CNN can only extract local features with limit window size. Therefore, a very deep CNN is used in text classification to extract hierarchical local features [21]. Similarly, a deep pyramid CNN [22] which carefully studies the deepening of word-level CNN is proposed to enable the discovery of long-range associations in text. By increasing the depth of network, this approach achieves good performance and reduces training time. However, most of the models use a fixed window size in CNN so that they cannot learn variable n-gram features. A densely connected CNN with multi-scale feature attention is proposed to extract variable n-gram features for text classification [9]. The dense connections build short-cut paths between upstream and downstream convolutional blocks, which enable the model to compose features of larger scale from those of smaller scale to produce variable n-gram features.

Although CNN-based approaches emphasize the extracting of variable n-gram features, they cannot learn sequential correlations. Moreover, the local semantic features extracted by CNN may contain redundancies.

B. RNN-Based Approaches

RNN is suitable for handling text sequences and modeling long-term dependencies sequentially [23]. Particularly, bidirectional recurrent neural network is able to capture global long-term dependencies. Therefore, many RNN variants are proposed for text classification. The gated recurrent neural network models the semantics of sentences and their relations adaptively [24]. The approach first learns representation with CNN or Long Short-Term Memory (LSTM). Afterwards, semantics of sentences and their relations are adaptively encoded in document representation with gated recurrent neural network. A hierarchical attention model which incorporates attention mechanism into hierarchical Gate Recurrent Unit (GRU) is proposed to capture the important information of a document [25]. The residual networks are incorporated into RNN to model longer text sequences and alleviate gradient vanishing problem [26]. An RNN variant Cached Long Short-Term Memory model is proposed to capture local and global semantic features of the long text sequence [27]. A memory with low forgetting rate captures the global semantic features while a memory with high forgetting rate captures the local semantic features.

The aforementioned RNN-based modes are specialized for sequential modeling with a recurrent hidden state whose activation at each time step depends on that of the previous time step's.

In sequential model, each hidden state is greatly affected by surrounding inputs and could not model long-term dependencies with a skipped span. It is problematic to capture combined semantics of text sequences.

C. Combined Approaches

Some researchers attempt to combine the advantages of CNN and RNN by using them to extract global long-term dependencies and local semantic features. C-LSTM [12] captures both local features of phrases as well as temporal sentence semantics. It utilizes CNN to extract a sequence of higher-level phrase representation which is then fed into a LSTM to obtain the sentence representation. A regional CNN-LSTM considers both the regional information within sentences and long-term dependency across sentences [13]. Similarly, Wang *et al.* [14] use CNN to capture local semantic features of text sequences and feed them into a RNN model to learn long-term dependency. On the contrary, global long-term dependency can firstly be extracted by RNN which is then fed into CNN to get the final representation [15]. These models capture global long-term dependencies and local semantic features simultaneously in a cascaded way. A hybrid conv-RNN framework combines the long-term dependencies and local semantic features through concatenation using both recurrent and convolutional neural networks. It seamlessly integrates merits on extracting different aspects of linguistic information from both structures [17]. Self-Attention Sandwich Neural Network (SA-SNN) [16] is proposed to extract local semantic representation and global structure representation simultaneously. It considers effective fusion of both with self-attention mechanism.

Combined approaches learn local semantic features and global long-term dependencies of text sequences simultaneously, but both are connected in a cascaded way [12]–[16] or combined through concatenation [17]. The cascaded way will deepen the depth of model and aggravate gradient vanishing problems. Furthermore, concatenated features may have redundancies. Moreover, on one hand, the global long-term dependencies extracted by a sequential recurrent neural network are combined sequentially so the combined semantics cannot be extracted well. On the other hand, local semantic features extracted by CNN may contain redundant features. The combined semantics and key local features are essential for text classification and they can interact with each other. However, exiting methods seldom model the mutual effect between global long-term dependencies and local features. Therefore, in this paper, we design a mutual attention mechanism consisting of a LGGA for capturing complex combined semantics and a GGLA for extracting key local semantic features.

D. Attention Mechanism

The attention mechanism [28] is proposed to compute alignment scores between source vectors $\mathbf{s} = [\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_n]$ and a target vector $\mathbf{t}$ by a compatibility function $f(\cdot)$ which measures the similarity between $\mathbf{s}$ and $\mathbf{t}$. Then, a softmax function is used to calculate a probability distribution $p(z = i|\mathbf{s}_i, \mathbf{t})(i = 1, 2, \ldots, n)$ by normalizing over all n elements of $\mathbf{s}$. A large $p(z = i|\mathbf{s}_i, \mathbf{t})$ means that $\mathbf{s}_i$ contributes important information to $\mathbf{t}$. The attention mechanism can be summarized as follows:

$$p(z = i|\mathbf{s}_i, \mathbf{t}) = \frac{\exp(f(\mathbf{s}_i, \mathbf{t}))}{\sum_{j=1}^{n} \exp(f(\mathbf{s}_j, \mathbf{t}))}. \tag{1}$$

The output $\mathbf{o}$ of this attention mechanism is a weighted sum of all elements in source s.

$$\mathbf{o} = \sum_{i=1}^{n} p(z = i|\mathbf{s}_i, \mathbf{t})\mathbf{s}_i. \tag{2}$$

There are many variants of attention mechanism, such as co-attention mechanism [29], [30] and self-attention [31], [32]. The co-attention mechanism is a special case of the aforementioned attention mechanism. It computes attention weights between two input sequences from different modalities, such as image and text, for visual question answering. The co-attention mechanism uses one of the modalities as source vectors and the other is compressed into a vector as target vector. Self-attention is another special case of the attention mechanism which replaces target vector $\mathbf{t}$ with an element $\mathbf{s}_i$ from the source input. It relates elements at different positions from a single sequence by computing the attention between each pair of elements $\mathbf{s}_i$ and $\mathbf{s}_j$. Self-attention is very expressive for modeling long-term dependency in a variety of NLP tasks. Most recently, the Transformer [31] based on self-attention has achieved state-of-the-art performance on machine translation. As a transformer-based approach, BERT (Bidirectional Encoder Representations from Transformers) [33] has achieved amazing results in many language understanding tasks, including the tasks of text classification.

III. Global-Local Mutual Attention Model

The overall framework of the GLMA model is shown in Fig. 1. It consists of two branches: the upper one captures global long-term dependencies with a bi-LSTM while the lower one extracts local semantic features with a convolution from the shared word embedding of the input text sequences. Both features are fed into the local-guided global attention and global-guided local attention to obtain global and local attention contexts, respectively. After that, both global and local attention contexts are fed into two weighted-over-time pooling operations, respectively. Then the two branches are combined by a fully connected layer and the final predictions are made by a softmax function layer.

For a text classification task, a training set of pair-wise data $\mathbf{S} = (\mathbf{W}_n, y_n)_{n=1}^{N}$ is given, where $\mathbf{W}_n = w_1, w_2, \ldots, w_T$, y_n, T, and N denote the text sequence, its corresponding label, the length of a text sequence, and the number of samples in the dataset, respectively. Let $\mathbf{x}_i \in R^d$ be the d-dimensional pre-trained word embedding vector of the i^{th} word w_i in a text sequence and then the input text sequence can be represented as an embedding:

$$\mathbf{x}_{1:T} = \mathbf{x}_1 \oplus \mathbf{x}_2 \oplus \ldots \oplus \mathbf{x}_T, \tag{3}$$

where $\oplus$ and $\mathbf{x}_{1:T} \in R^{T \times d}$ denote the concatenation operation and the input of bi-LSTM and CNN in our proposed model, respectively.

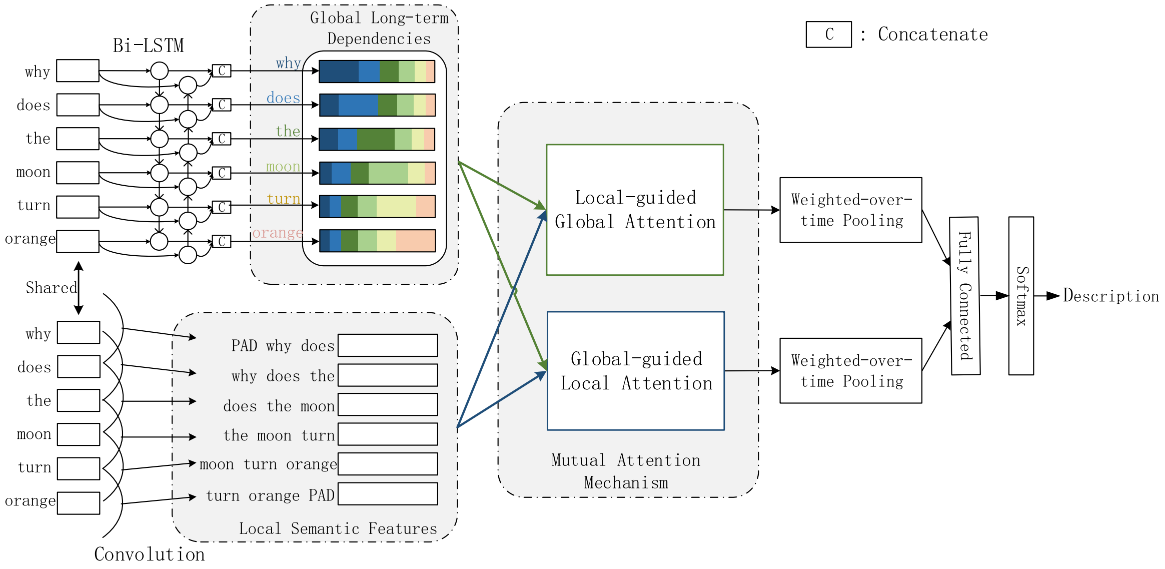

Fig. 1. The framework of the GLMA model. For the upper branch, different colors represent proportions of words at each position.

Global Long-Term Dependencies Extraction: The Long Short-Term Memory (LSTM) [34] is a popular RNN model and has been widely applied in various NLP tasks [35]. However, a single-direction LSTM is insufficient for learning long-term dependencies without utilizing the contextual information from future words. For modeling the global long-term dependency, a bi-LSTM [36] is employed to utilize both previous and future contexts by processing the sequence in both forward and backward directions. At each time step t, the output vectors of the two directions are concatenated.

Firstly, let k_{glo} be the hidden state dimension of a single direction LSTM. The hidden state $\mathbf{h}_t \in R^{k_{glo}}$ of a single direction LSTM at time step t is updated as follows:

$$\begin{pmatrix} \mathbf{i}_t \\ \mathbf{f}_t \\ \mathbf{o}_t \\ \mathbf{g}_t \end{pmatrix} = \begin{pmatrix} \sigma \\ \sigma \\ \sigma \\ \tanh \end{pmatrix} \mathbf{M} \begin{pmatrix} \mathbf{h}_{t-1} \\ \mathbf{x}_t \end{pmatrix}, \tag{4}$$

$$\mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \mathbf{g}_t, \tag{5}$$

$$\mathbf{h}_t = \mathbf{o}_t \odot \tanh\left(\mathbf{c}_t\right), \tag{6}$$

where $\odot$, $\mathbf{c}_t$, $\mathbf{g}_t$, $\sigma(\cdot)$, $\mathbf{M}$, $\mathbf{i}_t$, $\mathbf{f}_t$ and $\mathbf{o}_t$ denote the element-wise production operation, the cell memory vector, the intermediate calculation, the sigmoid function, the affine transform function consisting of trainable parameters, the input gate, the forget gate, and the output gate of a single directional LSTM, respectively.

Then, we feed the input text sequence $\mathbf{x}_{1:T}$ to the LSTM in the forward direction and obtain forward hidden state $\overrightarrow{\mathbf{h}}_t$ with Equations (4) to (6). We also update the backward hidden state $\overleftarrow{\mathbf{h}}_t$ by feeding the sequences into LSTM in a reverse direction. The hidden states of the two directions are concatenated as follows:

$$\mathbf{h}_t^{fb} = \overrightarrow{\mathbf{h}}_t \oplus \overleftarrow{\mathbf{h}}_t, \tag{7}$$

where $t = 1, 2, \ldots, T$ and $\mathbf{h}_t^{fb}$ represents the global long-term dependency at time step t as it contains text sequence information from both directions. All the hidden states are collected into a matrix, which is defined as

$$\mathbf{H} = [\mathbf{h}_1^{fb}, \mathbf{h}_2^{fb}, \ldots, \mathbf{h}_T^{fb}], \tag{8}$$

where $\mathbf{H} \in R^{T \times 2k_{glo}}$ and each row of $\mathbf{H}$ represents the global long-term dependency at the corresponding position of the input text sequence. Finally, the $\mathbf{H}$ and the local semantic features to be described below will be fed into the mutual attention mechanism as inputs.

Local Semantic Features Extraction: A one-dimensional convolution is employed to extract local features [6], which involves filter vectors sliding over a sequence and detects local semantic features at different positions. We denote $F \in R^{w \times d \times k_{loc}}$ as the convolution filter of the convolution operation where w, d, and k_{loc} denote the width of the convolution filter, the number of input dimensions, the number of convolution filters, respectively. Note that the height of convolution filter is equal to the input dimension d. For a word at position i, we take the text subsequence of word embedding $\mathbf{x}_{i-w/2+1:i+w/2}$ if w is even or $\mathbf{x}_{i-\lfloor w/2 \rfloor:i+\lfloor w/2 \rfloor}$ otherwise as inputs. Zeros are padded if the text subsequence has the number of elements less than w. The convolution operation is formulated as follows:

$$\tilde{\mathbf{c}}_i = \begin{cases} f(\mathbf{x}_{i-w/2+1:i+w/2} * F + b), & \text{if } w \text{ is even} \\ f(\mathbf{x}_{i-\lfloor w/2 \rfloor:i+\lfloor w/2 \rfloor} * F + b), & \text{if } w \text{ is odd} \end{cases} \tag{9}$$

where $\mathbf{x}_{i-w/2+1:i+w/2}$ refers to the concatenation of word embedding vectors $\mathbf{x}_{i-w/2+1}, \ldots, x_i, \ldots, x_{i+w/2}$, $*$, b, f, and $\tilde{\mathbf{c}}_i$ denote the convolution operation, the bias term, a nonlinear transformation function (can be either sigmoid, hyperbolic tangent, etc...), and the k_{loc} dimension local w-gram feature vector at i^{th} position of the text sequence, respectively. The filter is applied to each position of the text sequence with zero-padding to produce a feature map with the same length as the input as

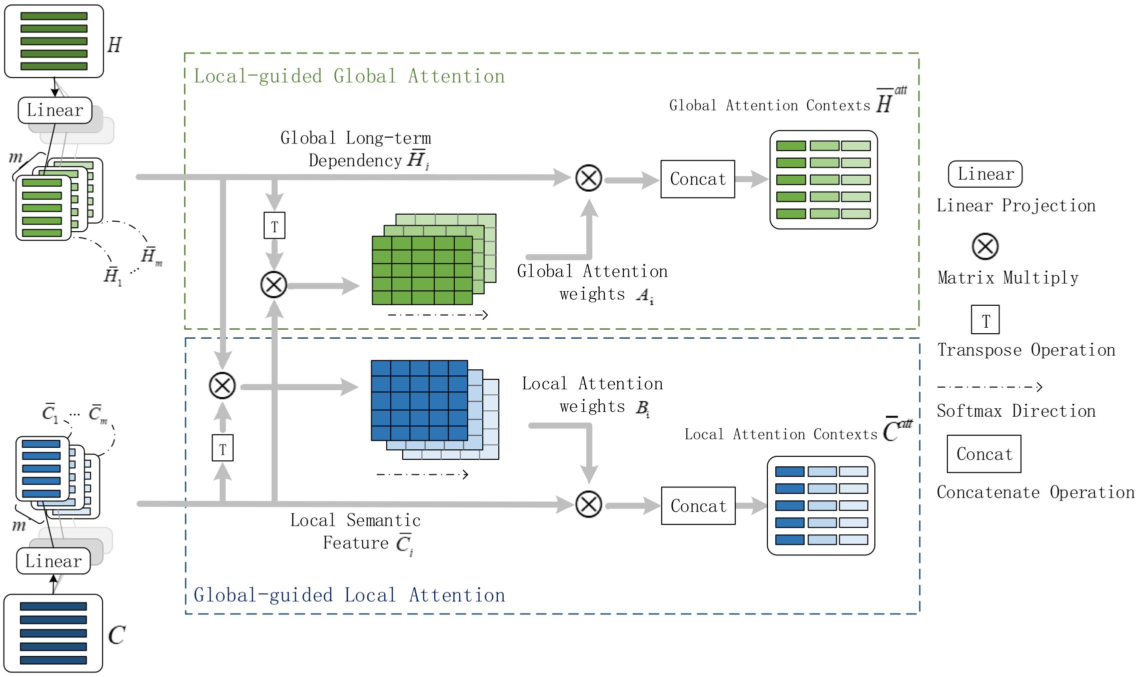
H
Linear
m
$\bar{H}_1$
$\bar{H}_m$
Local-guided Global Attention
Global Long-term Dependency $\bar{H}_i$
T
Global Attention weights A_i
Concat
Global Attention Contexts $\bar{H}^{att}$
Linear
Linear Projection
Matrix Multiply
T
Transpose Operation
Softmax Direction
Concat
Concatenate Operation
$\bar{C}_1 \cdots \bar{C}_m$
T
Local Attention weights B_i
Local Attention Contexts $\bar{C}^{att}$
Concat
Local Semantic Feature $\bar{C}_i$
m
Linear
Global-guided Local Attention
C

Fig. 2. The proposed global-local mutual attention (GLMA) mechanism which consists of the local-guided global attention (LGGA, the upper part) and the global-guided local attention (GGLA, the bottom part).

follows:

$$\tilde{\mathbf{C}} = [\tilde{\mathbf{c}}_1, \tilde{\mathbf{c}}_2, \ldots, \tilde{\mathbf{c}}_T], \tag{10}$$

where $\tilde{\mathbf{C}} \in R^{T \times k_{loc}}$.

The local semantic features are obtained as follows if multi-scale filters (r denotes the number of scales with varying scale w) are used to generate variable local w-gram features at each position:

$$\mathbf{C} = [\tilde{\mathbf{C}}^{(1)}, \tilde{\mathbf{C}}^{(2)}, \ldots, \tilde{\mathbf{C}}^{(r)}], \tag{11}$$

where $\mathbf{C} \in R^{T \times rk_{loc}}$.

A. Mutual Attention Mechanism

Both the global long-term dependencies ($\mathbf{H}$) and local semantic features ($\mathbf{C}$) at each word position are extracted independently from the same text sequence. However, the global long-term dependencies or the local semantic features being extracted alone may not be optimal for classification. On one hand, although the bi-LSTM is able to capture long-term information, it is a sequential model and cannot extract combined semantics because hidden states are sequentially obtained and equally combined at different time steps. On the other hand, text sequences often contain noises or irrelevant words/phrases so that not all extracted local semantic features are useful. Consequently, a further refinement on both the global dependencies and the local semantic features is desirable for obtaining the optimal features.

Inspired by the word attention techniques for neural machine translation tasks [31], [37] and co-attention mechanism for visual question answering tasks [29], we propose the global-local mutual attention mechanism containing the local-guided global attention (LGGA) and global-guided local attention (GGLA) to model the mutual effect between $\mathbf{H}$ and $\mathbf{C}$ of text sequences. The details of the proposed mutual attention mechanism are shown in Fig 2.

As shown in Fig. 2, $\mathbf{H}$ and $\mathbf{C}$ are linearly embedded into m different subspaces with lower dimension of k before being fed to the LGGA and GGLA. Here, m denotes the parallel mutual attention layers, or heads which enables the model to attend features in m different representation subspaces [31]. Note that the dimension k is much smaller than feature dimensions $2k_{glo}$ and rk_{loc}. The total computational cost is similar to that of single-head's with the original feature dimension because the dimension of each head is reduced. $\mathbf{H}$ and $\mathbf{C}$ are projected to each subspace i ($i = 1, 2, \ldots, m$) as follows:

$$\bar{\mathbf{H}}_i = \mathbf{H}\mathbf{W}_i^{glo}, \tag{12}$$

$$\bar{\mathbf{C}}_i = \mathbf{C}\mathbf{W}_i^{loc}, \tag{13}$$

where both $\mathbf{W}_i^{glo} \in R^{2k_{glo} \times k}$ and $\mathbf{W}_i^{loc} \in R^{rk_{loc} \times k}$ denote the weighting matrices.

1) Local-Guided Global Attention (LGGA): In LGGA, the global attention contexts of the global long-term dependencies $\bar{\mathbf{H}}_i$ are obtained according to the local semantic features $\bar{\mathbf{C}}_i$ of each word position. Specifically, given source vectors ($\bar{\mathbf{C}}_i$ as guiding information) and target vectors ($\bar{\mathbf{H}}_i$ as target information), the dot-product between the source vectors and the target vectors is computed. After it is divided by $\sqrt{k}$, a softmax function is applied to each row to obtain the weights of the target $\bar{\mathbf{H}}_i$. In other words, it allows every position of local semantic features to attend over all positions of global long-term dependencies. The LGGA computes the global attention weights (A_i) as follows:

$$\mathbf{A}_i = \text{softmax}\left(\frac{\bar{\mathbf{C}}_i \bar{\mathbf{H}}_i^{\mathrm{T}}}{\sqrt{k}}\right), \quad (14)$$

$$\bar{\mathbf{H}}_i^{att} = \mathbf{A}_i \bar{\mathbf{H}}_i, \quad (15)$$

where $\mathbf{A}_i \in R^{T \times T}$ and $\bar{\mathbf{H}}_i^{att} (i = 1, 2, \ldots, m)$ denotes the global attention weights between LGGA and the i^{th} head of global attention contexts. Note that we apply a softmax function along each row (the sum of each row in $\mathbf{A}_i$ is equal to 1) to obtain the global attention weights. With global attention weights, the proposed model keeps the useful distant information of H and obtains sufficient meaningful combined semantics from H that are semantically related to the learned weights.

After that, all the heads of global attention contexts are concatenated as follows:

$$\bar{\mathbf{H}}^{att} = [\bar{\mathbf{H}}_1^{att}, \bar{\mathbf{H}}_2^{att}, \ldots, \bar{\mathbf{H}}_m^{att}], \quad (16)$$

where $\bar{\mathbf{H}}^{att} \in R^{T \times mk}$. Through the LGGA, there are paths directly connecting all of the hidden states of the bi-LSTM which play a similar role as the weighted skip connections [20] so the LGGA alleviates the gradient vanishing problem by shortening the path of gradient propagation.

2) Global-Guided Local Attention (GGLA): Similarly, in the GGLA, the local attention contexts of the local semantic features are extracted according to the global long-term dependency at each word position. For the global long-term dependency at each word position, the GGLA automatically assigns larger weights to the more relevant and informative local semantic features as follows:

$$\mathbf{B}_i = \text{softmax}\left(\frac{\bar{\mathbf{H}}_i \bar{\mathbf{C}}_i^{\mathrm{T}}}{\sqrt{k}}\right), \quad (17)$$

$$\bar{\mathbf{C}}_i^{att} = \mathbf{B}_i \bar{\mathbf{C}}_i, \quad (18)$$

where $\mathbf{B}_i \in R^{T \times T}$ and $\bar{\mathbf{C}}_i^{att}$(i = 1, 2, ..., m) denotes the local attention weights between the GGLA and the i^{th} head of local attention contexts. Note that we apply a softmax function along each row (the sum of each row of $\mathbf{B}_i$ is equal to 1) to obtain local attention weights. All heads of local attention contexts are concatenated as follows:

$$\bar{\mathbf{C}}^{att} = [\bar{\mathbf{C}}_1^{att}, \bar{\mathbf{C}}_2^{att}, \ldots, \bar{\mathbf{C}}_m^{att}], \quad (19)$$

where $\bar{\mathbf{C}}^{att} \in R^{T \times mk}$.

There are two major differences in our work from previous co-attention mechanisms. Firstly, existing co-attention mechanisms [29], [30] only consider the calculation of attention weights on target information from the whole guiding information. We argue that this may limit the amount of possible interactions between guiding and target information. The GLMA has fine-grain interactions as it computes attention weights between any guiding information and any target information. Secondly, existing co-attention mechanisms calculate attention weights in the original feature space. In contrast, the GLMA has multi-head and calculates attention weights on different feature representation subspaces to obtain diversified features.

Here, we discuss the difference between the proposed attention and the multi-head attention mechanism [31]. As mentioned in [31], attention mechanism can be described as a mapping of a query and a set of key-value pairs to an output. The output is computed as a weighted sum of the values, where the weight assigned to each value is computed by a compatibility function of the query with the corresponding key. The calculation process of our proposed attention mechanism is similar to the multi-head attention mechanism. However, they have some essential differences. First, the multi-head attention mechanism calculates attention weights between encoder and decoder features, which are both extracted from self-attention layers. However, our proposed attention mechanism calculates attention weights between global and local features, which are both encoder features from different encoder networks. In other words, our attention explores the mutual effect between two different scales of information (global features and local features) from different network architectures. Second, weighting matrices of key and value in our proposed attention mechanism are designed to be shared while the ones in the multi-head attention are different. The number of parameters can thus be reduced in our proposed attention mechanism comparing to the multi-head attention. In addition, since the weighting matrices of key and value are shared, key and value can be regarded as one scale of features (global or local), while the query can represent the other one (local or global). Therefore, it enables our model to learn the mutual effects of the two different scales of features with the proposed mutual attention mechanism.

B. Weighted-Over-Time Pooling

Both global attention contexts and local attention contexts are obtained by the global-local mutual attention mechanism. Most existing work employs either the max-over-time or the average-over-time pooling to aggregate the sequence vectors into one vector over the time dimension [15], [38]. However, both the max-over-time and the average-over-time pooling lose position and intensity information of features at different time dimensions.

In this paper, we propose a weighted-over-time pooling which adaptively assigns a scalar score over the time dimension to each feature vector of a sequence and compresses the sequence into a single vector. Fig. 3 shows the detail of weighted-over-time pooling operation. The proposed weighted-over-time pooling is applied to the global attention contexts and the local attention contexts, respectively. The weighted-over-time pooling on the global attention contexts is as follows:

$$\boldsymbol{\alpha} = \sigma(\bar{\mathbf{H}}^{att}\mathbf{W}^{(1)} + \mathbf{B}^{(1)})\mathbf{w}^{(2)} + \mathbf{b}^{(2)}, \quad (20)$$

$$\mathbf{p}_i^H = \frac{exp(\boldsymbol{\alpha}_i)}{\sum_{j=1}^{T} exp(\boldsymbol{\alpha}_j)}, \quad (21)$$

$$\mathbf{z}^H = \mathbf{p}^H \bar{\mathbf{H}}^{att}, \quad (22)$$

where $\boldsymbol{\alpha} \in R^T$ and p_i^H $(i = 1, 2, \ldots, T)$. $\mathbf{W}^{(1)} \in R^{mk \times mk}$, $\mathbf{w}^{(2)} \in R^{mk}$, $\mathbf{B}^{(1)} \in R^{T \times mk}$, and $\mathbf{b}^{(2)} \in R^T$ are learnable parameters. $\sigma(\cdot)$, $\mathbf{p}^H \in R^T$, and $\mathbf{z}^H \in R^{mk}$ denote a sigmoid

TABLE I
ACCURACIES (%) AND P-VALUES OF THE GLMA AND OTHER STATE-OF-THE-ART MODELS ON SEVEN BENCHMARK DATASETS

Method	Movie Review	SUBJ	TREC	CR	20New	MPQA	AG	P-value
CNN-multichannel [6]	81.10	93.20	92.20	85.00	96.93*	89.40	92.05*	0.0180
bi-LSTM [16]	79.70	92.80	93.00*	83.88*	95.20*	90.29*	91.60	0.0180
RCNN [48]	80.03*	93.29*	93.20*	84.48*	96.49	90.32*	92.21*	0.0180
C-LSTM [12]	80.57*	93.94*	94.60	82.29*	96.89*	90.34*	93.59*	0.0180
CNN-LSTM-word2vec [14]	81.52	93.17*	93.40*	83.87*	96.31*	90.30*	92.24*	0.0180
conv-RNN [17]	81.99	94.13	95.40*	86.20*	95.68*	90.23*	92.46*	0.0180
SA-SNN [16]	82.10	93.90	96.00	86.76*	96.80*	90.59*	93.67*	0.0180
Self-Attentive [49]	80.10	92.50	96.00*	82.08*	95.45*	**90.81***	91.10	0.0280
HS-LSTM [50]	82.10	93.70	-	-	-	-	92.50	0.1088
GLMA	**82.15**	**94.47**	**96.80**	**87.18**	**97.48**	90.72	**94.01**	

1. The results with * are obtained by their published source code or our re-implementation. We are confident that our implementations are correct because they achieve classification accuracies similar to the reported results in the original papers.
2. The - indicates that the authors of HS-LSTM did not publish the source code of how to pre-process sentences by using some heuristic methods so that we could not re-implement the results.

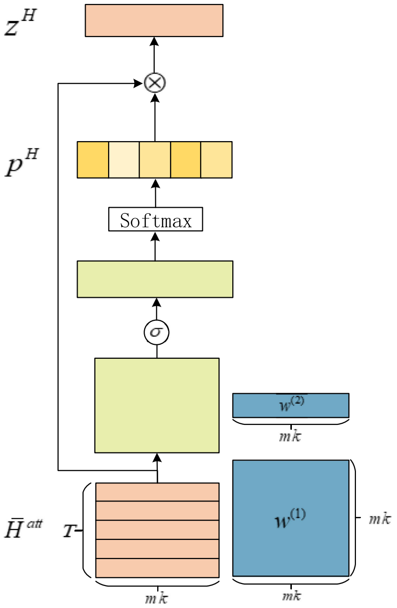

Fig. 3. The weighted-over-time pooling operation on global attention contexts $\bar{\mathbf{H}}^{att}$. Pink color shapes represent input and output, green color shapes are hidden representation, yellow color shapes are scalar scores $\mathbf{p}^H$ of each position of input, and blue color shapes stand for weight matrix. Here, the bias term is omitted for concision.

activation function, the scalar scores of global attention contexts, and the final global representation vector, respectively.

Let $\mathbf{p}^C$ be the scalar scores of local attention contexts and $\mathbf{z}^C$ be the final local representation vector. The weighted-over-time pooling on the local attention contexts is applied in the same way as in Equations (20) to (22), but with the local attention contexts as inputs, to obtain $\mathbf{p}^C$ and $\mathbf{z}^C$.

Finally, we feed the final global representation vectors ($\mathbf{z}^H$) and final local representation vectors ($\mathbf{z}^C$) to a fully connected layer with rectified linear unit (ReLU) activation. The output of fully connected layer is then fed to a softmax function to predict the probability distributions of categories. The cross-entropy loss between predicted probability distribution and the referenced distribution of categories is minimized to learn the model parameters.

IV. EXPERIMENTS

The effectiveness of the proposed model for text classification is tested on 23 datasets, including 7 benchmark datasets (Movie Review [39], SUBJ [40], TREC [41], CR [42], 20New [43], MPQA [44], and AG [45]) and 16 Amazon product review datasets [46]. A summary statistics and validation protocols of these datasets are listed in the Table I of the Supplementary Material. In addition, these datasets are briefly described in the Supplementary Material.

A. Experimental Setup

The 300-dimensional pre-trained $word2vec$[1] [47] vectors are used as word embedding while Rectified Linear Units (RELU) are used as the nonlinear function in the convolutional layer. The number of convolution filters k_{loc} is set to 128. The number of hidden state dimension of bi-LSTM is set to 192 ($k_{gloc} = 192$). We choose the head number $m = 8$ and the subspace dimensions $k = 64$. Both word embedding and fully connected layer employ Dropout operation with dropout rate of 0.5 and we do not perform any l_2 regularization over the parameters. The gradient-based optimizer Adam is used to minimize the cross-entropy loss between predicted and true distributions, and the training stopped early when the accuracy on development set starts to drop. The batch size is chosen from 32, 64, and 128 according to the size of datasets. The learning rate is set to 0.001. The same parameter configuration is used for all datasets.

B. Comparison Models

For seven benchmark datasets, the experimental results are compared with other start-of-the-art models including CNN-multichannel [6], bi-LSTM [16], RCNN [48], C-LSTM [12], CNN-LSTM-word2vec [14], conv-RNN [17], SA-SNN [16], Self-Attentive [49], and HS-LSTM [50]. For the 16 Amazon product review datasets, the experimental results are compared with SA-SNN [16] and SLSTM [51] models. The description of these models are listed in the Supplementary Material.

[1] https://code.google.com/p/word2vec/

TABLE II
ABLATION STUDIES ON DIFFERENT COMPONENTS OF THE PROPOSED MODEL

Dataset	global only	local only	W/O mutual-att	W/O LGGA	W/O GGLA	**GLMA**
Movie Review	80.90	81.15	81.50	81.95	81.60	**82.15**
SUBJ	92.45	92.39	93.47	93.83	93.61	**94.47**
TREC	93.20	94.40	95.80	96.00	96.20	**96.80**
CR	85.28	85.64	85.96	86.40	86.63	**87.18**
20New	95.77	95.51	96.06	97.00	96.63	**97.48**
P-value	0.0431	0.0431	0.0422	0.0431	0.0422	-

C. Experimental Results

Classification accuracies of the proposed GLMA compared with other approaches are shown in the Table I in the main text and Table II in Supplementary Material for 7 benchmark datasets and 16 Amazon product review datasets, respectively. We also use the Wilcoxon signed-rank test method [52] (a paired comparison) to verify the significance of differences between GLMA and other approaches. For the general standard of Wilcoxon signed-rank test (i.e., P-value < 0.05), the GLMA achieves a significantly better performance than other approaches.

The Table I shows that accuracies of the GLMA outperform other state-of-the-art baselines on seven benchmark datasets. Compared with the CNN-multichannel exploring local semantic features only or the bi-LSTM exploring global long-term dependencies only, the GLMA performs much better because it learns both simultaneously. When compared with the C-LSTM, the CNN-LSTM-word2ve, the conv-RNN, and the SA-SNN model, the GLMA achieves better results, although they all extract local semantic features and global long-term dependencies by CNN and RNN simultaneously. The main reason is that the proposed GLMA extracts more discriminative global and local features of text sequences by exploring the mutual relationship between local semantic features and global long-term dependencies. The SA-SNN yields similar accuracies as that of the GLMA, but the SA-SNN relies more on prior knowledge, i.e., two versions of word embedding: the word2vec and the Glove [53]. The GLMA uses the word2vec as word embedding only but outperforms SA-SNN. Moreover, the SA-SNN compresses global or local features into a single vector which is not sufficient to capture all important information of the whole text sequence [17]. In contrast, the GLMA keeps global long-term dependencies and local semantic features of each position and explores the fine-grained mutual effect of them to capture useful combined semantics and filter irrelevant features.

The Table II in the Supplementary Material shows that the GLMA outperforms other state-of-the-art baseline methods in 14 out of 16 Amazon product review datasets. According to the Table I in Supplementary Materials and the Table II in the Supplementary Material, the GLMA achieves a better performance on datasets with long text sequences, e.g., the 20New, the Books, the Electronics, the DVD, and the IMDB datasets. One reason is that the proposed mutual attention mechanism reduces the depth of the model and is beneficial to the gradient propagation for strong long-term dependency learning. Moreover, the LGGA has the advantages of extracting combined semantic features which is especially important for long text sequence classification tasks. Moreover, we find that the length of each sample varies greatly in the Camera and the Magazines datasets. The GLMA models mutual effects between global and local features subject to the maximal length of inputs. However, the SLSTM is able to handle text sequences with variable lengths and thus performs better on both the Camera and the Magazines datasets.

D. Ablation Analysis

In this section, we conduct ablation studies to quantify the influence of each component in the GLMA on five benchmark datasets.

"global only": This model extracts global long-term dependency of text sequences only with a bi-LSTM. After that, a weighted-over-time pooling operation and a fully connected layer are applied to obtain final global representations. In other words, this model corresponds to the upper branch in Fig. 1.

"local only": This model extracts local semantic features of text sequences only with a multi-scale CNN. After that, a weighted-over-time pooling operation and a fully connected layer are applied to obtain the final local features. This model corresponds to the bottom branch in Fig. 1.

"W/O mutual-att": The architecture of this model is the same as that of the GLMA without the mutual attention mechanism (without cross arrow lines between the two branches in Fig. 1).

"W/O LGGA": A model that almost the same as the GLMA but not using the local-guided global attention (without the arrow line from the bottom branch to the upper branch in Fig. 1).

"W/O GGLA": A model that almost the same as the GLMA but not using the global-guided local attention (without the arrow line from the upper branch to the bottom branch in Fig. 1).

"GLMA-max" and "GLMA-avg": "GLMA-max" and "GLMA-avg" are models replacing the weighted-over-time pooling in GLMA with max-over-time pooling or average-over-time pooling.

Comparison results and the P-values of Wilcoxon signed-rank tests between GLMA and other ablation models show in Table II. With P-value less than 0.05, it means that GLMA achieves a significantly better performance than other ablation models. The GLMA outperforms "local only" and "global only" models which demonstrates the necessity of modeling both global and local features simultaneously. According to the "W/O mutual-att" column of Table II, the performance drops compared with the GLMA which shows the effectiveness of the mutual attention mechanism in the GLMA. Comparing the "W/O LGGA" and the "W/O GGLA" with the GLMA model, the performance drop demonstrates the effectiveness of the two parts in GLMA. The model "W/O LGGA" combines long-term dependencies sequentially without considering the semantic relatedness so

TABLE III
COMPARISON RESULTS OF AVERAGE-OVER-TIME POOLING, MAX-OVER-TIME POOLING AND WEIGHTED-OVER-TIME POOLING

Dataset	GLMA-avg	GLMA-max	**GLMA**
Movie Review	81.58	81.95	**82.15**
SUBJ	94.25	94.11	**94.47**
TREC	96.20	96.60	**96.80**
CR	86.31	86.68	**87.18**
20New	96.72	96.74	**97.48**
P-value	0.0431	0.0422	-

the combined semantics cannot be extracted. On the other hand, local semantic features extracted by the "W/O GGLA" may have noises or redundancies but the GLMA addresses this problem by learning global long-term dependencies to weight local semantic features at different positions. The effectiveness of the GLMA is further demonstrated through attention visualization in Section G.

Moreover, we compare the GLMA with the GLMA using max-over-time pooling ("GLMA-max") and the GLMA using average-over-time pooling ("GLMA-avg") instead of the weighted-over-time pooling to verify the effectiveness of the weighted-over-time pooling in the GLMA. Comparison results and the P-values between GLMA and "GLMA-max"/"GLMA-avg" are shown in Table III. It shows that GLMA with weighted-over-time pooling mechanism improves about 1% comparing to models with max-over-time pooling and average-over-time pooling mechanism. In Table III, the P-values less than 0.05 further proves that the weighted-over-time pooling mechanism significantly outperforms max-over-time and average-over-time pooling mechanisms. These results demonstrate the effectiveness of aggregating the most informative and discriminative features into a single vector representation using the proposed weighted-over-time pooling.

E. Qualitative Analysis

In this section, error analysis is performed on the GLMA, the "W/O GGLA," and the "W/O LGGA" using the Movie Review dataset to investigate the necessity of the local-guided global attention and the global-guided local attention in GLMA. Table IV shows two examples that have been classified correctly by the GLMA model. The first example is misclassified by "W/O GGLA" but correctly classified by "W/O LGGA". On the contrary, the second example is misclassified by the "W/O LGGA" while the "W/O GGLA" classified it correctly. For the first example, the "W/O GGLA" fails to extract the key local negative features *soft and stinky* and is misled by other obvious local positive feature *so ripe*, which cause the misclassification. However, the global-guided local attention in both the GLMA and the "W/O LGGA" uses global long-term dependencies as guiding information to learn to assign more weights to key local semantic features *soft and stinky* in this example which helps to correctly classify it. For the second example, the "W/O LGGA" fails to capture the combined semantic *but manages old problems* because the global long-term dependencies are extracted from a sequential recurrent network and combined equally. It extracts local features with negative polarities only such as *it isn't*, *is unfamiliar*, *old problems* which result in misclassification. However, both the GLAM and the "W/O GGLA" classified it correctly because they use the local-guided global attention to capture global long-term dependencies and combine them with learned weights to obtain distant combined semantics (*but manages old problems*).

F. Mutual Attention Visualization

In this section, mutual attention weights of each head and scalar scores of weighted-over-time pooling are visualized to investigate how global long-term dependencies and local semantic features are aligned and affect each other as well as the effectiveness of weighted-over-time pooling operation. The proposed mutual attention mechanism contains 8 heads of each local-guided global attention and global-guided local attention, respectively. After mutual attention, both global long-term dependencies and local semantic features are fed into the classification layer through the weighted-over-time pooling. So for a position, the bigger the scalar value of the global long-term dependency or the local semantic feature is, the greater the impact on the loss function is. Thus, we could cautiously interpret the classification results using our mutual attention weights and the scalar scores of weighted-over-time pooling.

The visualization of mutual attention and weighted-over-pooling of a sample from the Movie Review dataset (the first sample in Table IV) are shown in Figs. 4 and 5, which is correctly classified to the Negative category.

Fig. 4 shows the heat maps of the scalar scores' distribution (p^H, subfigure (a)) and global attention weights (A_i in eq. (14), subfigures from (b) to (i)) of 8 heads. In subfigure (a), the position at words *narrative* and *film* are chosen by the weighted-over-time pooling because a darker color indicates higher importance for the GLMA final prediction. In subfigure (b) to (i), the row of local semantic features at the position of *narrative* assigns weights for global long-term dependency at all positions, and especially assigns more weights for global long-term dependency at the position of *stinky* in subfigure (b) to (d), *ripe* in subfigure (e) and (g), and *can't help* in subfigure (h) to (i), respectively. Similarly, the row of local semantic features at the position of *film* assigns larger weights to global long-term dependencies at the position of *stinky* in subfigure (b) to (d), *ripe* in subfigure (e) to (f), and *can't help* in subfigure (g) to (h), respectively. That is the local-guided global attention tries to combine distant semantics (such as, *ripe can't help stinky* in this sample) which are discriminative features for correct classifications.

Similarly, Fig. 5 shows the heat maps of the scalar scores distribution (p^C, subfigure (a)) and local attention weights (B_i in eq. (17), subfigures from (b) to (i)) of 8 heads. Informative positions at *pivotal narrative point*, *so ripe*, and *soft and stinky* are chosen by weighted-over-time pooling in subfigure (a) for the final prediction. In subfigure (b), the global long-term dependencies of these chosen positions distribute more weights to local semantic features *narrative*, *film*, and *stinky*. Similar to subfigures from (c) to (i), the global long-term dependencies also help to distribute weights focusing on *narrative*, *film*, and *stinky* which are the most key local features for a correct prediction in this sample.

TABLE IV
ERROR ANALYSIS FOR THE NECESSITY OF GLOBAL-GUIDED LOCAL ATTENTION AND LOCAL-GUIDED GLOBAL ATTENTION. EXAMPLES IN THIS TABLE ARE SELECTED FROM THE MOVIE REVIEW DATASET

Examples	ground truth	Our model	W/O GGLA	W/O LGGA
1: The pivotal narrative point is **so ripe** the film can't help but go **soft and stinky**	Negative	Negative	Positive	Negative
2: It isn't that the picture is unfamiliar **but** that it **manages** to find new avenues of discourse on **old problems**	Positive	Positive	Positive	Negative

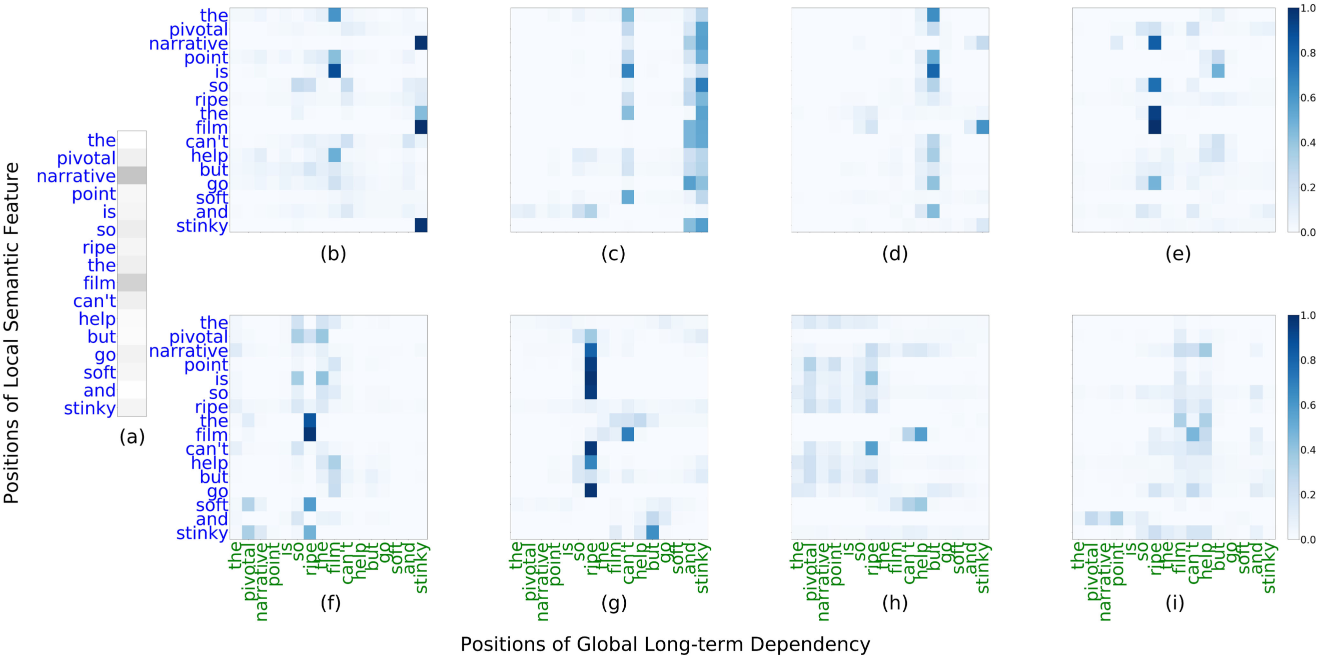

Fig. 4. Visualization of local-guided global attention (LGGA) weights and the scalar scores of weighted-over-time pooling of a sample from Movie Review dataset. Blue colors represent local semantic features at each position and green colors represent global long-term dependency at each position.

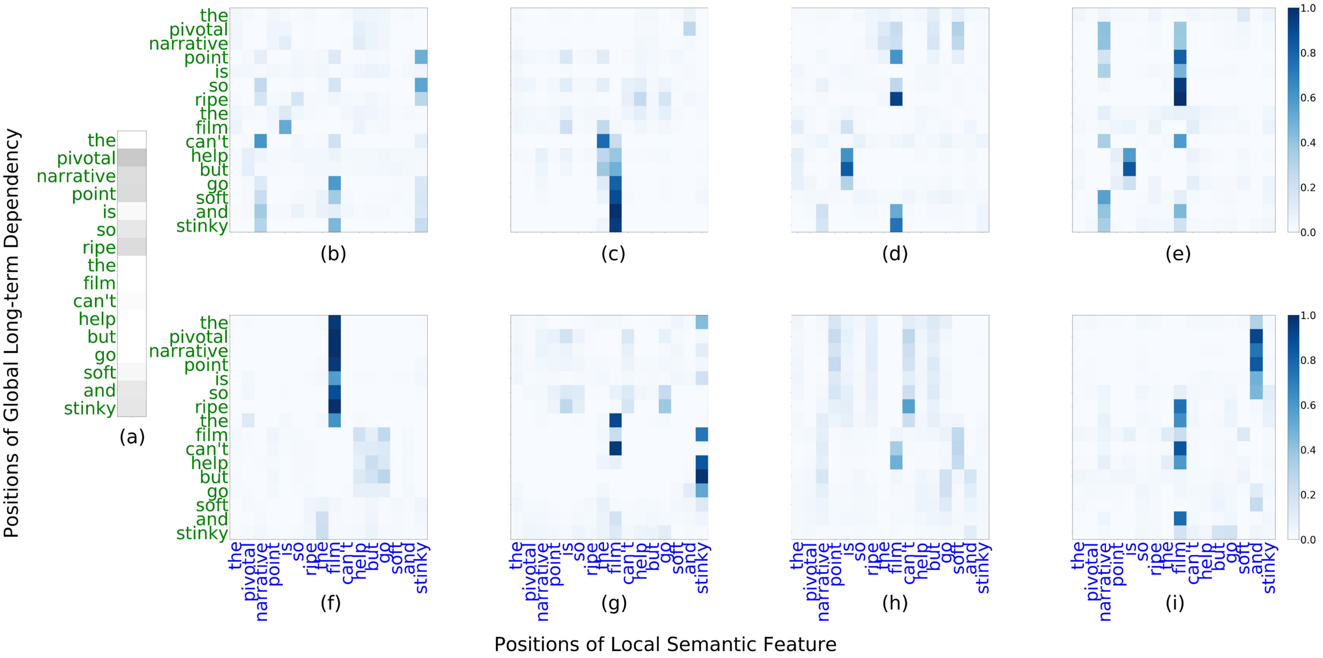

Fig. 5. Visualization of global-guided local attention (GGLA) weights and the scalar scores of weighted-over-time pooling of a sample from Movie Review dataset. Blue colors represent local semantic features at each position and green colors represent global long-term dependency at each position.

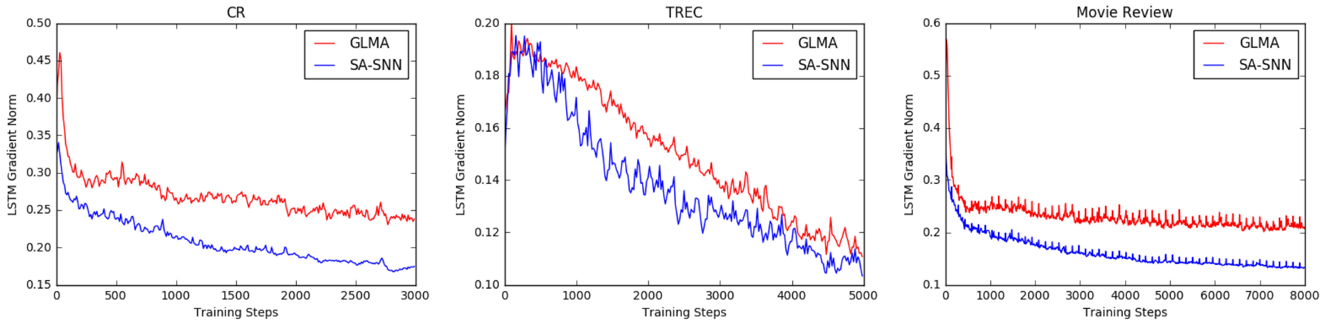

Fig. 6. Average gradient norms of the first $L/2$ positions of the forward LSTM of the GLMA and the SA-SNN.

Judging by the combined semantics in Fig. 4 and key local features in Fig. 5, the GLMA is undoubtedly able to classify the sample into a Negative category.

G. Gradient Analysis

In this section, we compare the variations of average gradient norm of the LSTM in the GLMA and the SA-SNN during the training stage. It demonstrates the abilities of the GLMA in reducing the model depth and alleviating the gradient vanishing problem. We re-implement the SA-SNN with strong confidence that the replicated results are almost the same as the original paper. Note that we calculate the average gradient norm of the first L/2 positions of training samples where gradients vanish easily. L denotes the average sentence length of samples. The average gradient norm is calculated in the forward direction LSTM of the GLMA only and the first forward direction LSTM layer of the SA-SNN, respectively. Both models employ *word2vec* as word embedding and the same training configurations (including learning rate, batch-size, and 50 training epochs for each dataset).

Fig. 6 shows the average gradient norms of the first $L/2$ positions of forward LSTM of the loss function at each training steps of the forward LSTM in the GLMA and the SA-SNN on the CR, the TREC, and the Movie Review datasets. According to Fig. 6, the average gradient norm of the first $L/2$ positions of the GLMA is bigger than that of the SA-SNN. The propagation of more gradients to the first L/2 positions of the forward LSTM in the GLMA helps to alleviate the gradient vanishing problem.

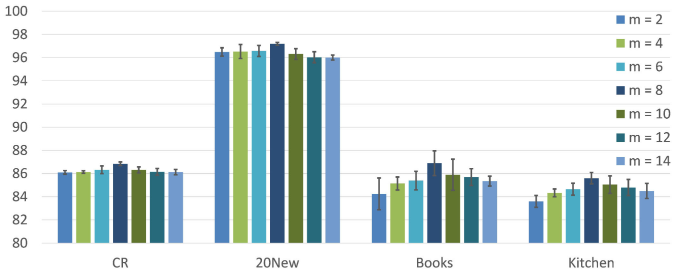

Fig. 7. Average and stand deviations of accuracies (%) of the GLMA with $m = 2$, 4, 6, 8, 10, 12 and 14 on CR, 20New, Books, and Kitchen datasets.

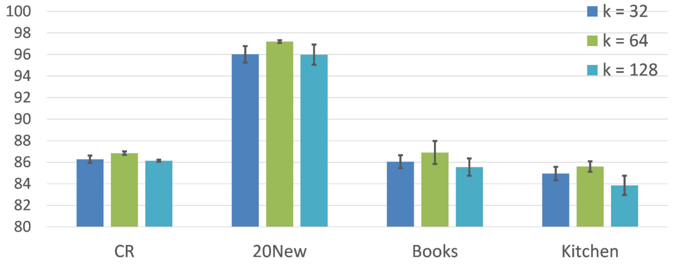

Fig. 8. Average accuracies (%) and stand deviations of GLMA with different value of the dimension of head (k) on CR, 20New, Books and Kitchen datasets.

H. Influences of Hyper-Parameters

Experiments on the CR, the 20New, and two Amazon product reviews including the Books and the Kitchen datasets are conducted to study influences of two key hyper-parameters: the number of heads (m) and the dimension of heads (k).

1) Various Numbers of Heads (m): m varies from 2 to 14 with interval of 2 and other hyper-parameters are kept unchanged. Experimental results are shown in Fig. 7. One can observe the following facts: 1) The performance improves when m is less than or equal to 8 which shows that the GLMA is able to attend features in different representation subspaces and aggregation complex features with more heads. 2) However, the performance drops when m is larger than 8, such as $m > 10$, because the model is too complicated for the datasets and suffers from over-fitting.

2) Dimension of Head (k): In the experiments, dimension k varies but other hyper parameters remain unchanged. Experimental results are listed in Fig. 8, it shows that $k = 64$ is the best choice which always achieves the best accuracies. The experimental results are consistent with the setting in [31]. When the subspace dimension of k is too small, the mutual attention's attending features in different subspaces may be insufficient. In contrast, the model may become too complicated and lead to over-fitting for large values of k.

I. Computational Efficiency Analysis

One of the cons of the GLMA is that the CNN block has to wait for the sequential process of the Bi-LSTM block, especially for long text sequences. However, in our practical experiments, the average length of text sequences is not so long that the effect of the processing time of Bi-LSTM is little. The computation of the proposed mutual attention mechanism can be parallelized so that it does not require much time for train or test. We conduct an experiment on the 20New dataset to test the computational time consumption of both the GLMA and the SA-SNN. The 20New dataset consists of 7520 training and 5563 testing samples with an average length of 429. The GLMA spends 0.42 seconds in training for each sample and 6.544 seconds in testing on average. However, the SA-SNN spends 0.52 seconds in training for each sample and 7.868 seconds in testing on average. These results demonstrate that the GLMA has a higher computational efficiency in comparison with the SA-SNN.

V. Conclusion

In this work, a global-local mutual attention model is proposed to capture both local semantic features and global long-term dependencies effectively. The mutual attention mechanism contains a local-guided global attention which keeps the useful information of global long-term dependencies and extracts combined semantics. Moreover, it also has a global-guided local attention which extracts the most relevant and informative local semantic features. A weighted-over-time pooling is developed for distinguishing discriminative features of text sequences for classification which is more effective than the average-over-time pooling and the max-over-time pooling. Our model demonstrates a better performance on seven benchmark datasets and sixteen Amazon product reviews datasets. Additionally, ablation studies, qualitative analysis, and attention weights visualization are provided to further prove the effectiveness of the proposed model. In this paper, we find a kind of mutual effects between global features and local features in text sequences. Our future work will focus on exploring mutual effects between other different information from the same data. For example, interactions among high-level features (e.g. shape information in images) and low-level features (e.g. texture information in images).

References

[1] C. C. Aggarwal and C. Zhai, “A survey of text classification algorithms,” in *Mining Text Data*, 2012, vol. 9781461432234, pp. 163–222.

[2] D. Tang, B. Qin, F. Wei, L. Dong, T. Liu, and M. Zhou, “A joint segmentation and classification framework for sentence level sentiment classification,” *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 23, no. 11, pp. 1750–1761, Nov. 2015.

[3] D. Zhang and W. S. Lee, “Question classification using support vector machines,” in *Proc. 26th Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval.*, 2003, pp. 26–32.

[4] S. Tong and D. Koller, “Support vector machine active learning with applications to text classification.” *Mach. Learn. Res.*, vol. 2, no. 1, pp. 999–1006, 2002.

[5] W. Yin, K. Kann, M. Yu, and H. Schütze, “Comparative study of CNN and RNN for natural language processing,” 2017, *arXiv:1702.01923*.

[6] Y. Kim, “Convolutional neural networks for sentence classification,” in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2014, pp. 1746–1751.

[7] N. Kalchbrenner, E. Grefenstette, and P. Blunsom, “A convolutional neural network for modelling sentences,” in *Proc. 52nd Annu. Meeting Assoc. Comput. Linguistics*, 2014, vol. 1, pp. 655–665.

[8] S. Li, Z. Zhao, T. Liu, R. Hu, and X. Du, “Initializing convolutional filters with semantic features for text classification,” in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2017, pp. 1884–1889.

[9] S. Wang, M. Huang, and Z. Deng, “Densely connected CNN with multi-scale feature attention for text classification,” in *Proc. Int. Joint Conf. Artif. Intell.*, 2018, pp. 4468–4474.

[10] H. T. Ng and J. Zelle, “Corpus-based approaches to semantic interpretation in natural language processing,” *AI Mag.*, vol. 18, no. 4, pp. 45–64, 1997.

[11] D. J. Hess, D. J. Foss, and P. Carroll, “Effects of global and local context on lexical processing during language comprehension,” *J. Exp. Psychol., Gen.*, vol. 124, no. 1, pp. 62–82, 1995.

[12] C. Zhou, C. Sun, Z. Liu, and F. C. M. Lau, “A C-LSTM neural network for text classification,” *Comput. Sci.*, vol. 1, no. 4, pp. 39–44, 2015.

[13] J. Wang, L.-C. Yu, K. R. Lai, and X. Zhang, “Dimensional sentiment analysis using a regional CNN-LSTM model,” in *Proc. 54th Annu. Meeting Assoc. Comput. Linguistics*, 2016, vol. 2, pp. 225–230.

[14] X. Wang, W. Jiang, and Z. Luo, “Combination of convolutional and recurrent neural network for sentiment analysis of short texts,” in *Proc. 26th Int. Conf. Comput. Linguistics, Tech. Papers*, 2016, pp. 2428–2437.

[15] R. Zhang, H. Lee, and D. Radev, “Dependency sensitive convolutional neural networks for modeling sentences and documents,” in *Proc. NAACL-HLT*, 2016, pp. 1512–1521.

[16] J. Zhao *et al.*, “Adaptive learning of local semantic and global structure representations for text classification,” in *Proc. 27th Int. Conf. Comput. Linguistics*, 2018, pp. 2033–2043.

[17] C. Wang, F. Jiang, and H. Yang, “A hybrid framework for text modeling with convolutional RNN,” in *Proc. 23rd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2017, pp. 2061–2069.

[18] Y. Yaslan and Z. Cataltepe, “Co-training with relevant random subspaces,” *Neurocomputing*, vol. 73, no. 10/12, pp. 1652–1661, 2010.

[19] S. R. Bowman, J. Gauthier, A. Rastogi, R. Gupta, C. Manning, and C. Potts, “A fast unified model for parsing and sentence understanding,” in *Proc. 54th Annu. Meeting Assoc. Comput. Linguistics*, 2016, pp. 1466–1477.

[20] D. Britz, A. Goldie, M.-T. Luong, and Q. Le, “Massive exploration of neural machine translation architectures,” in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2017, pp. 1442–1451.

[21] A. Conneau, H. Schwenk, L. Barrault, and Y. Lecun, “Very deep convolutional networks for text classification,” in *Proc. 15th Conf. Eur. Chapter Assoc. Comput. Linguistics*, 2017, pp. 1107–1116.

[22] R. Johnson and T. Zhang, “Deep pyramid convolutional neural networks for text categorization,” in *Proc. 55th Annu. Meeting Assoc. Comput. Linguistics*, 2017, vol. 1, pp. 562–570.

[23] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, “Empirical evaluation of gated recurrent neural networks on sequence modeling,” in *NIPS Workshop Deep Learn.*, Dec. 2014.

[24] 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.*, 2015, pp. 1422–1432.

[25] 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, Human Lang. Technol.*, 2016, pp. 1480–1489.

[26] Y. Wang and F. Tian, “Recurrent residual learning for sequence classification,” in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2016, pp. 938–943.

[27] J. Xu, C. Danlu, X. Qiu, and X. Huang, “Cached long short-term memory neural networks for document-level sentiment classification,” in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2016, pp. 1660–1669.

[28] M.-T. Luong, H. Pham, and C. D. Manning, “Effective approaches to attention-based neural machine translation,” in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2015, pp. 1412–1421.

[29] J. Lu, J. Yang, D. Batra, and D. Parikh, “Hierarchical question-image co-attention for visual question answering,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 289–297.

[30] D. Yu, J. Fu, T. Mei, and Y. Rui, “Multi-level attention networks for visual question answering,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 4187–4195.

[31] A. Vaswani *et al.*, “Attention is all you need,” in *Proc. Adv. Neural Inform. Process. Syst.*, 2017, pp. 5998–6008.

[32] J. Fu *et al.*, “Dual attention network for scene segmentation,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 3146–3154.

[33] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “Bert: Pretraining of deep bidirectional transformers for language understanding,” in *Proc. Conf. North American Chapter Assoc. Comput. Linguistics: Human Lang. Technologies*, vol. 1, Jun. 2019, pp. 4171–4186.

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

[35] H. Palangi *et al.*, “Deep sentence embedding using long short-term memory networks: Analysis and application to information retrieval,” *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 24, no. 4, pp. 694–707, Apr. 2016.

[36] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Trans. Signal Process.*, vol. 45, no. 11, pp. 2673–2681, Nov. 1997.

[37] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," 2016, *arXiv:1409.0473v2*.

[38] B. Wang, "Disconnected recurrent neural networks for text categorization," in *Proc. 56th Annu. Meeting Assoc. Comput. Linguistics*, 2018, vol. 1, pp. 2311–2320.

[39] B. Pang and L. Lee, "Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales," in *Proc. 43rd Annu. Meeting Assoc. Comput. Linguistics*, 2005, pp. 115–124.

[40] B. Pang and L. Lee, "A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts," in *Proc. 42nd Annu. Meeting Assoc. Comput. Linguistics*, 2004, pp. 271–278.

[41] X. Li and D. Roth, "Learning question classifiers," in *Proc. 19th Int. Conf. Comput. Linguistics*, 2002, pp. 1–7.

[42] M. Hu and B. Liu, "Mining and summarizing customer reviews," in *Proc. 10th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2004, pp. 168–177.

[43] S. Hingmire, S. Chougule, G. K. Palshikar, and S. Chakraborti, "Document classification by topic labeling," in *Proc. 36th Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, 2013, pp. 877–880.

[44] J. Wiebe, T. Wilson, and C. Cardie, "Annotating expressions of opinions and emotions in language," *Lang. Resour. Eval*, vol. 39, no. 2–3, pp. 165–210, 2005.

[45] X. Zhang, J. Zhao, and Y. LeCun, "Character-level convolutional networks for text classification," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 649–657.

[46] J. Blitzer, M. Dredze, and F. Pereira, "Biographies, bollywood, boomboxes and blenders: Domain adaptation for sentiment classification," in *Proc. 45th Annu. Meeting Assoc. Comput. Linguistics*, 2007, pp. 440–447.

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

[48] S. Lai, L. Xu, K. Liu, and J. Zhao, "Recurrent convolutional neural networks for text classification." in *Proc. Conf. Assoc. Advancement Artif. Intell.*, 2015, 2015, pp. 2267–2273.

[49] Z. Lin *et al.*, "A structured self-attentive sentence embedding," in *5th Int. Conf. Learn. Representations, ICLR*, Toulon, France, Apr. 24-26, 2017.

[50] T. Zhang, M. Huang, and L. Zhao, "Learning structured representation for text classification via reinforcement learning," in *Proc. 32nd AAAI Conf. Artif. Intell.*, 2018.

[51] Y. Zhang, Q. Liu, and L. Song, "Sentence-state LSTM for text representation," in *Proc. 56th Annu. Meeting Assoc. Comput. Linguistics*, 2018, vol. 1, pp. 317–327.

[52] F. Wilcoxon, "Individual comparisons by ranking methods," *Biometrics Bull.*, vol. 1, no. 6, pp. 80–83, 1945.

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

Qianli Ma (M'17) received the Ph.D. degree in computer science from the South China University of Technology, Guangzhou, China, in 2008. He is an Associate Professor with the School of Computer Science and Engineering, South China University of Technology. From 2016 to 2017, he was a Visiting Scholar with the University of California at San Diego, La Jolla, CA, USA.

His current research interests include machine learning algorithms, data-mining methodologies, and their applications.

Liuhong Yu is currently working toward the master's degree in computer science and engineering from the South China University of Technology, Guangzhou, China.

Her current research interests include machine learning, deep learning, and natural-language processing.

Shuai Tian received the master's degree in computer science and engineering from the South China University of Technology, Guangzhou, China, in 2019.

His current research interests include machine learning, deep learning, and natural-language processing.

Enhuan Chen received the master's degree in computer science and engineering from the South China University of Technology, Guangzhou, China, in 2019.

His current research interests include machine learning, deep learning, and natural-language processing.

Wing W. Y. Ng (S'02–M'05–SM'15) received the B.Sc. and Ph.D. degrees in computer science from Hong Kong Polytechnic University, Hong Kong, in 2001 and 2006, respectively.

He is a Professor with the School of Computer Science and Engineering, South China University of Technology, Guangzhou, China, where he is currently the Deputy Director of the Guangdong Provincial Key Laboratory of Computational Intelligence and Cyberspace Information. His current research interests include neural networks, deep learning, smart grid, smart health care, smart manufacturing, and nonstationary information retrieval.

Dr. Ng is currently an Associate Editor for the International Journal of Machine Learning and Cybernetics. He is a Principle Investigator of four China National Nature Science Foundation projects and a Program for New Century Excellent Talents in University from China Ministry of Education. He was the Board of Governor of IEEE Systems, Man and Cybernetics Society from 2011 to 2013.