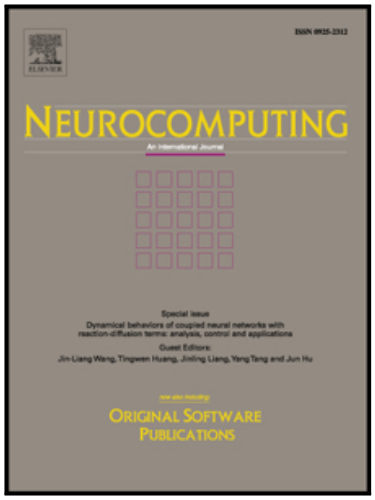
# Journal Pre-proof




Attention augmentation with multi-residual in Bidirectional LSTM

Ye Wang, Xinxiang Zhang, Mi Lu, Han Wang, Yoonsuck Choe

PII: S0925-2312(19)31506-1
DOI: https://doi.org/10.1016/j.neucom.2019.10.068
Reference: NEUCOM 21459

To appear in: *Neurocomputing*

Received date: 27 November 2018
Revised date: 8 May 2019
Accepted date: 18 October 2019

Please cite this article as: Ye Wang, Xinxiang Zhang, Mi Lu, Han Wang, Yoonsuck Choe, Attention augmentation with multi-residual in Bidirectional LSTM, *Neurocomputing* (2019), doi: https://doi.org/10.1016/j.neucom.2019.10.068

# Attention augmentation with multi-residual in Bidirectional LSTM

Ye Wang[a,*], Xinxiang Zhang[b], Mi Lu[a], Han Wang[c], Yoonsuck Choe[c]

[a]*Department of Electrical and Computer Engineering*
*Texas A&M University, College Station, TX, USA, 77840*
[b]*Department of Electrical Engineering*
*Southern Methodist University, Dallas, TX, USA, 75205*
[c]*Department of Computer Science*
*Texas A&M University, College Station, TX, USA, 77840*

**Abstract**

Recurrent neural networks (RNNs) have been proven to be efficient in processing sequential data. However, the traditional RNNs have suffered from the gradient diminishing problem until the advent of Long Short-Term Memory (LSTM). However, LSTM is weak in capturing long-time dependency in sequential data due to the inadequacy of memory capacity in LSTM cells. To address this challenge, we propose an Attention-augmentation Bidirectional Multi-residual Recurrent Neural Network (ABMRNN) to overcome the deficiency. We propose an algorithm which integrates both past and future information at every time step with omniscient attention model. The multi-residual mechanism has also been leveraged in the proposed model targeting the pattern of the relationship between current time step and further distant time steps instead of only one previous time step. The results of experiments show that our model outperforms the traditional statistical classifiers and other existing RNN architectures.

*Keywords:* Long Short-Term Memory, attention augmentation, natural language processing, multi-residual network

## 1. Introduction

Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) are the two primary architectures in neural networks. RNNs are often applied to sequential data such as natural language processing and speech processing [1, 2], while CNNs are more employed in image processing areas [3, 4, 5]. Among the existing RNN models, LSTM [6] is one of the most popular approaches since it initially solves gradient vanishing and exploding problems during RNN training by introducing forget gate and memory cell. After the literature review, we found that numerous RNN variations have been proposed to achieve the state-of-the-art performance in different tasks, where LSTM is the cornerstone of those structures. However, due to the limited memory cell in LSTM, when the time sequence is long, the LSTM performance is heavily influenced.

With the increase of the depth of layers and the length of the sequences, residual networks have proved their advantages in both CNNs [7, 8] and RNNs [9]. Residual networks connect current and distant previous time steps for optimizing of the layer information. [7] and [10] propose similar residual ideas to randomly connect one previous distant time step to current time step, where the problem of long time dependency is solved partially. Therefore, the residual networks motivate us to combine the residual network with LSTM, where the information in current time step has been updated dynamically based on the attained correlation between previous time steps and current time step.

To better attain the correlation between current time step and previous time steps, the attention model is widely

*Corresponding author: Tel.: +1-469-321-1300;
*Email address:* wangye0523@tamu.edu (Ye Wang)

applied in image processing, speech processing and natural language processing. The objective of the attention model is to ultimately optimize the training procedure when the amount of attention is limited. [11] initially leverages the attention model from image processing to natural language processing. [12] proposes a model as a decoder network between previous states and current state. [13] simplifies the model as an attention-based weighted pooling RNN to acquire utterance representation in speech processing. Since the attention is limited, the way to effectively distribute those attention becomes considerably important. Inspired by the attention-based approaches, we leverage the attention model [14] to strengthen the correlation between the current state and both previous and future distant states. As for [14], they focus on the relationship between current time step with previous information. Since the objective is to allocate the attention properly, we regard the past and the future as the same importance, which means we integrate both the previous and the future time steps to refine the information of current time step instead of only relying on previous time steps.

Therefore, to address the aforementioned challenges in long time dependency and optimizing the text correlations, this paper develops an Attention-augmentation Bidirectional Multi-residual Recurrent Neural Network (ABMRNN). The proposed ABMRNN achieves the state-of-the-art performance among several existing sequential classification tasks. The main contributions of this work are summarized as following:

- Our algorithm overcomes the deficiency of LSTM in weak modeling the long-time dependency, so we can handle much longer sequential data and obtain higher accuracy rate in longer sequential tasks. In this algorithm, we design a novel bi-directional layer to dynamically acquire and allocate attention from both previous and future time steps. Bi-directional layers help us focus not only on the past but also on the future, so that we can attain the better correlation between current steps and distant time steps (In STCT, after incorporating the bi-direction layer, the accuracy has improved from 93.01% to 94.10%).

- To better supplement the acquired attention from bi-directional layer, we also leverage the multi-residual mechanism to the recurrent networks. Compared with traditional residual networks, the advantages of ours are more obvious because the proposed multi-residual mechanism is more rational than previous residual networks in sequence learning, since the traditional residual networks connect current time step with only one randomly previous time step.

- The proposed model contains fewer parameters than current popular models such as [7] and [15], which indicates that the architecture of the proposed model is less complicated than those popular models. The proposed model also achieves the state-of-the-art performance in sequence learning of STCT from 93.01% to 96.50%.

## 2. Related Work

Three major directions have been studied in recent years towards the structure exploration for sequence learning, including capturing better feature representation, optimizing cell memory usage and improving the capacity of long time dependency. First, an increasing number of layers and numerous embedding techniques are employed for feature extraction [16][17][18]. However, with the increase of layers, the computational complexity becomes critical and unaffordable in current neural networks. Second, a wide range of variations towards the RNN interior cell structure units such as LSTM and GRU [19] are proposed. Nonetheless, those RNN variations are confronted with the same imperfection due to the limited cell memory. Third, the attention-based approaches [11][20] are proposed to improve the capacity of long time dependency in RNN variations. Nevertheless, partial information has been obtained because the proposed attention models only focus on previous states.

Therefore, the contribution of our work integrates the advantages of residual networks and attention-augmentation mechanism for the tasks of interest. Unlike the popular trend of combining deeper and wider neural networks [21], we propose a novel RNN variation with forward and backward layers. The proposed model attains the text information from both past and future time steps based on the limited memory cell of LSTM and improves the capacity of long time dependency.

## 3. Recurrent Neural Networks Preliminaries

The basic LSTM architecture of solving the problem of gradient diminishing in traditional RNNs is described and the illustrated equations are given in Section 3.1. Besides, the fundamental residual mechanism employed in recurrent neural networks is presented in Section 3.2.

### *3.1. Long Short-Term Memory (LSTM)*

The traditional RNNs have suffered from gradient diminishing problem until the advent of LSTM. The appearance of LSTM is meaningful because the authors introduced the gates' mechanism by adding nonlinear activation functions. Activation functions squash the values of these vectors between 0 and 1. Figure 1 shows the structure of LSTM cell, where Equations (1-6) follows the data flow: The input gate activation vector $i_t$ (forget gate activation vector $f_t$, output gate activation vector $o_t$ respectively) is obtained by the sigmoid function of the updated current input vector $x_t$ and updated hidden state vector $h_{t-1}$. The update of $x_t$ is by a weight matrix $U$ converting the input to the current hidden layer in the input gate (forget gate, output gate respectively). The update of $h_{t-1}$ is through matrix $W^i$, representing a recurrent connection between the previous hidden layer and the current layer in the input gate (forget gate, output gate respectively).

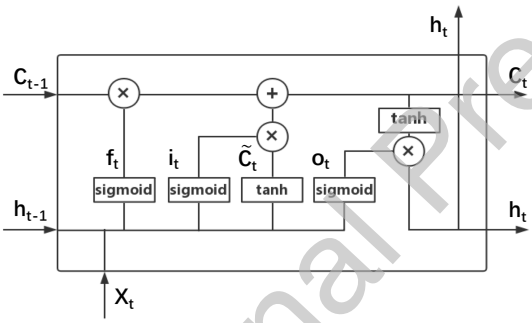



Figure 1: LSTM cell

$$i_t = \sigma\left(x_t U^i + h_{t-1} W^i\right) \tag{1}$$

$$f_t = \sigma\left(x_t U^f + h_{t-1} W^f\right) \tag{2}$$

$$o_t = \sigma\left(x_t U^o + h_{t-1} W^o\right) \tag{3}$$

In the forget gate, by the element-wise multiplication activation function with input and hidden state vectors, we can control the amount of information from the previous state. Similarly, in the output gate, we can control the amount of information between internal state and external network.

$$\tilde{C}_t = \tanh\left(x_t U^c + h_{t-1} W^c\right) \tag{4}$$

$$C_t = \sigma\left(f_t * C_{t-1} + i_t * \tilde{C}_t\right) \tag{5}$$

$$h_t = \tanh(C_t) * o_t \tag{6}$$

For cell state vector $C_t$, $\tilde{C}_t$ is the candidate value to update it. $\tilde{C}_t$ can be obtained by the tanh function of updated $x_t$ and update $h_t$, the output vector of the LSTM unit in Equation 6. $C_t$ is obtained by the sigmoid function of two Hadamard products (element-wise products), the Hadamard product of $f_t$ and $C_{t-1}$, and that of $i_t$ and $\tilde{C}_t$.

### *3.2. Recurrent Residual Network*

LSTM solves gradient vanishing and exploding problems. However, the dependency between the past and the current information is neglected because current time step only depends on previous time step if the time sequence is too long. To enhance such a distant relationship, residual recurrent neural networks have been proposed [9][10].

Figure 2 shows the general structure of a residual recurrent network. Residual recurrent network introduces a direct shortcut between different time steps to strengthen the connection.

Regarding the implementation of the basic residual network, for each current time step $t$, we consider both the previous time step $t-1$ and the additional specific previous time step (e.g. we assume it is $t_0$ as Figure 2 shows).

$$C_{t-1}^{new} = C_{t-1}^{old} + \alpha * C_0 \tag{7}$$

$$h_{t-1}^{new} = h_{t-1}^{old} + \alpha * h_0 \tag{8}$$

where $\alpha$ represents the specific scalar weight of how much information is imported from previous time step to current time step. Recurrent residual networks leverage the convolutional residual network [7] and improve the performance in particular sequential tasks.

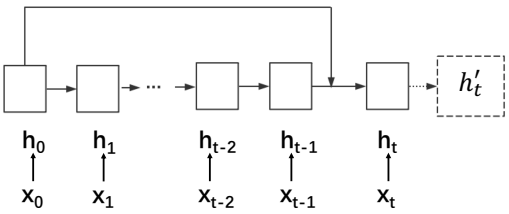



Figure 2: Residual Recurrent Network

## 4. Proposed Scheme

In this section, we propose an Attention-augmentation Bidirectional Multi-residual Recurrent Neural Network (ABMRNN). The algorithm will be illustrated and scheme described in details. First, a modified attention model is proposed which can resolve the memory cell limitation in LSTM, by the sliding window-based implementation. Also, efforts are made to identify the specific previous time step such that linking to it is most effective. Instead of randomly connecting to a previous time step, a multi-residual mechanism is developed to enhance the correlation of current time step and distant previous time steps. Additionally, a bidirectional multi-residual mechanism is proposed that can combine both past and future information comprehensively, rather than partially capturing the previous information only. Moreover, the detailed training procedure of ABMRNN is made available, and its function further explained.

### *4.1. Attention-augmentation mechanism*

The objective of the attention model is to ultimately optimize the training procedure when the amount of attention is limited. Therefore, the way to effectively distribute those attention becomes considerably crucial. We illustrate the equations as follows:

$$WS = \Sigma_{T=t1}^{tn}(a_T \times h_T) \tag{9}$$

$$a_T = \frac{exp(W \cdot h_T)}{\Sigma_{T=t1}^{tn} exp(W \cdot h_T)} \tag{10}$$

In Equation 9, we define a Weighted Summation (WS) at current time step $T$ as the whole attention from previous time steps. $h_T$ represents the value in hidden state at time step $T$. $a_T$ is a scalar value representing the weight at time step $T$. We compute $a_T$ through a softmax form, and $W$ is a parameter which needs to be learned during training. $exp(W \times h_T)$ represents the potential energy at time step $T$. Figure 3 illustrates one example of the attention model. To simplify the model, previous six steps are considered. At time step $t$, we compute the relationship between $h_t$ and previous time steps. The energy height represents the corresponding weight value. Therefore, we put more attention in specific time steps of which the energy is high. In Figure 3, we acquire the highest attention at $t-3$ and the lowest attention at $t-6$ when current time step is $t$. For every selected time step $T$, theoretically, we should review all the previous states to obtain the comprehensive relationship. However, the complexity of such an attention model is too high, being $O(N^2)$ if we compute all the previous states. Due to the limitation of computational power, we select the fixed $N$ past states to cover by a sliding window, complexity is hence reduced to $O(N)$.

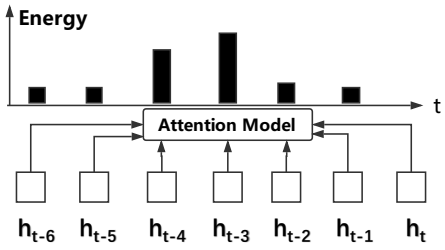



Figure 3: Attention Model

### *4.2. Multi-residual LSTM*

Due to the limited memory cell in LSTM, when the time sequence is long, the LSTM performance is heavily impeded. The residual networks introduced in Section 3.2 inspire us by combing residual network and LSTM with attention model. In this way, the information in current time step has been updated reasonably because we obtain

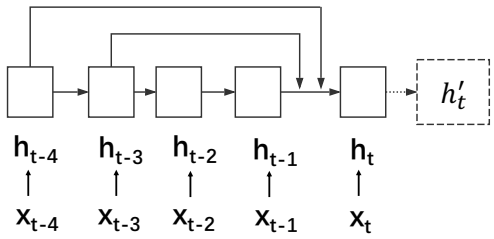



Figure 4: Multi-residual LSTM with attention Model, unrolling our model along the time axis. The dashed box indicates the updated state in current time step.

the correlation between previous time steps and current time step.

[7] initially proposes a residual learning framework. They attempt to build a block as:

$$y = \mathcal{F}(x, \{W_i\}) + x \tag{11}$$

where $x$ and $y$ are input and output vectors respectively. The function $\mathcal{F}(x, \{W_i\})$ represents the residual mapping to be learned. The operation $\mathcal{F} + x$ is performed by a shortcut connection and element-wise addition.

Our idea is illustrated in Figure 4. With the help of attention model, we are inferred that $h_{t-4}$ and $h_{t-3}$ gained more attentions compared with other states when we only pick top two attention. Therefore, we import the information from $h_{t-4}$ and $h_{t-3}$ to current time step $h_t$. The proposed equations below are introduced to explain our model:

$$\tilde{C}_t = \tanh\left(W_C \cdot [h'_{t-1}, x_t]\right) \tag{12}$$

$$C_t = \sigma\left(f_t * C_{t-1} + i_t * \tilde{C}_t\right) \tag{13}$$

$$C' = \Sigma_T(W_{C_T} \times C_T) \tag{14}$$

$$W_{C_T} = \frac{exp(W \cdot C_T)}{\Sigma_T exp(W \cdot C_T)} \tag{15}$$

$$h_t = \tanh(C_t) * o_t \tag{16}$$

$$h' = \Sigma_T(W_{h_T} \times h_T) \tag{17}$$

$$W_{h_T} = \frac{exp(W \cdot h_T)}{\Sigma_T exp(W \cdot h_T)} \tag{18}$$

In Equation 12, we modify $\tilde{C}_t$ by updated previous hidden state $h'_{t-1}$. In Equation 13, we calculate $C_t$ by using the updated $\tilde{C}_t$. In our model, we update every $C_t$ and $h_t$ not only depending on time $t-1$, but also involving several past states by weighted summation of multi-residual schemes in terms of memory cells and hidden states. Therefore, in Equation 14 and 17, the updated $C'$ and $h'$ are introduced by the weighted summation of the previous time steps. $T$ is the candidate set of previous time steps. $W_{C_T}$ and $W_{h_T}$ are both scalar weighted numbers at the corresponding specific time steps. In Equation 15 and 18, $W$ is a shared candidate vector by the model which needs to be learned from data.

### *4.3. Bidirectional Multi-residual network*

Compared with the previous models, We add one more layer as shown in Figure 5. The forward sequence and backward sequence are fed into training processing simultaneously. The advantage is to enable updating the weights combining both previous and future time steps. The equations are shown as follows:

$$h'_t = [\overrightarrow{h}_t, \overleftarrow{h}_t] \tag{19}$$

$$\overrightarrow{h'}_{t-1} = \overrightarrow{h}_{t-1} + \Sigma_{\overrightarrow{T}} \overrightarrow{a_T}(\tanh(C_T) \otimes \sigma(x_T)) \tag{20}$$

$$\overleftarrow{h'}_{t-1} = \overleftarrow{h}_{t-1} + \Sigma_{\overleftarrow{T}} \overleftarrow{a_T}(\tanh(C_T) \otimes \sigma(x_T)) \tag{21}$$

where $\overrightarrow{T}, \overleftarrow{T} \in \mathbb{N}, t-n \leq \overrightarrow{T} \leq t-1, t+1 \leq \overleftarrow{T} \leq t+n$.

In Equation 20 and Equation 21, $\overrightarrow{h}_{t-1}$ and $\overleftarrow{h}_{t-1}$ are the original forward and backward hidden states at time step $t-1$ of bidirectional LSTM respectively. $\overrightarrow{h'}_{t-1}$ and $\overleftarrow{h'}_{t-1}$ are updated forward and backward hidden states at time step $t-1$ of the proposed ABMRNN. The concept of residual is the weighted summation of the hidden states from selected time steps $T$ based on the attention scalar $a_T$ obtained in Equation 10. The updated hidden states at time step $t-1$ are the input to compute the output at time step $t$.

Since the forward sequence and backward sequence are fed into training processing simultaneously, both of the past and future information have been considered together with the current time step. Besides, our model allows more time flexibilities which enable recalling past pieces of information, predicting the future time steps and evaluating the influences between each states and current state.

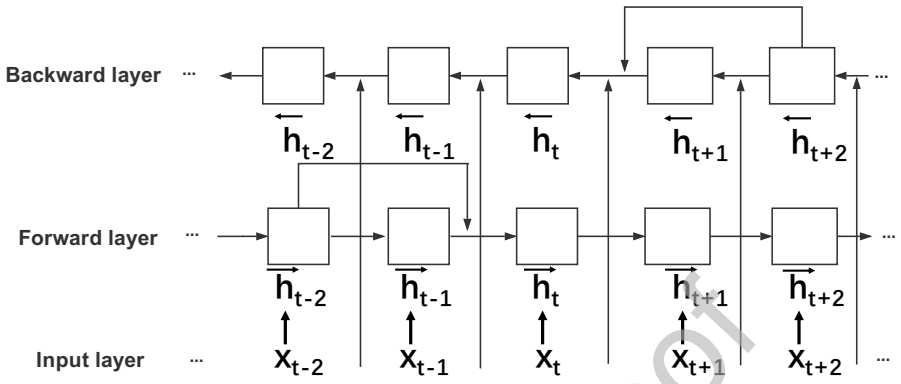



Figure 5: Bidirectional multi-residual LSTM with attention model

### 4.4. Training procedure

**Algorithm 1** ABMRNN training procedure

$x_{bi} \leftarrow \{\overrightarrow{x}, \overleftarrow{x}\}$ where $\overrightarrow{x}$ is the forward input, $t$ is the target value and $\overleftarrow{x}$ is the backward (reversed) input.
$\epsilon$ : number of epochs
$e \leftarrow 0$
**for** $e < \epsilon$ **do**
  **for** $x_T \in x_{bi}$ **do**
    $y \leftarrow \mathcal{F}(x_{bi}, \{W\})$
    $W_{tmp} \leftarrow \mathcal{A}(\{W\}), h^* \leftarrow \mathcal{H}(h, \{W_{tmp}\})$
    $y^* \leftarrow \mathcal{F}(y, h^*, \{W_{tmp}\})$
    error $E \leftarrow \|t - y^*\|$
    update $W \leftarrow backpropagate(W, E)$
  **end for**
  $e \leftarrow e + 1$
**end for**

We present the training procedure of ABMRNN as pseudo-code in Algorithm 1. The input sequence $x_{bi}$ is composed of a forward order sequence and a backward order sequence. The objective is to minimize the loss function by updating hidden states and the corresponding attention-augmentation dynamics.

$\mathcal{F}$ is denoted as the function of ABMRNN to obtain output states when the input is $x_{bi}$ and matrix weights is $W$. $\mathcal{A}$ represents the function of updated attention, where we can obtain the attention distribution. $\mathcal{H}$ stands for the function of updated hidden states which means we have imported the important information from previous and future to current time step. $W$ is the matrix weights while $W_{tmp}$ is the temporary updated weights from attention model $\mathcal{A}$. $h$ is defined as the initial hidden states while $h^*$ is denoted as the updated hidden states. $y$ is the initial output while $y^*$ is the updated output.

## 5. Experiments and Results

### 5.1. Task introduction

1. Table 1 shows the detailed statistics of each dataset. Short-term text classification task (STCT) is still a challenging task. Unlike traditional long text documents, short-term texts including headings and news titles are usually concise, which somehow impact the performance. [22] introduces STCT which is constructed by average 20 Chinese words in coarse and refined categories. There are eight labels in coarse class and 59 labels in refined class. The total number of texts is 400,000. Besides, the baseline performance has been provided by traditional statistical

| Dataset | Ave. Len | Max Len | #Classes | #Train : #Test |
|---|---|---|---|---|
| STCT | 20 | 30 | 8 | 28,000 : 12,000 |
| IMDB | 300 | 3000 | 2 | 25,000 : 25,000 |
| AG_NEWS | 30 | 200 | 4 | 8,000 : 1,000 |
| MNIST | 784 | 784 | 10 | 60,000 : 10,000 |

Table 1: Classification Datasets

| Model | IMDB | AG_NEWS | MNIST | STCT |
|---|---|---|---|---|
| Plain LSTM | 88.77% | 82.33% | 97.01% | 93.01% |
| Bi-LSTM | 89.91% | 83.13% | 98.31% | 94.10% |
| 2-layer LSTM | 88.42% | 82.27% | 98.03% | 93.16% |
| 1-layer IndRNN | 80.60% | 84.98% | 97.58% | 93.02% |
| 5-layer IndRNN | 76.39% | 84.74% | 97.71% | 88.89% |
| Plain RNN | 77.12% | 80.33% | 97.66% | 78.89% |
| 5-layer RNN | 50.00% | 77.76% | 97.45% | 87.23% |
| 1-D CNN | 88.70% | 84.61% | 98.01% | 94.50% |
| Attention-LSTM | 89.50% | 82.17% | 98.31% | 95.88% |
| Residual-LSTM | 90.80% | 84.71% | 98.03% | 93.55% |
| Our-model | **90.91%** | **86.31%** | **98.53%** | **96.50%** |

Table 2: Accuracy in classification Results

methods such as support vector machine, decision tree, logistic regression and so on.

2. AG_NEWS is a collection of more than 1 million news articles. All the titles have been labeled in four categories. We randomly select 8,000 samples for training and 1000 samples for testing.
3. IMDB movie review dataset is a binary classification task containing movie reviews with positive and negative labels. We select 5000 samples in total, half for training and the other half for testing. The maximum length in the review is up to 3000 and the average length is about 300.
4. Originally, MNIST is an image classification task (10 categories). However, we flat the pixels as the sequential data and feed into the training to predict the image label. Therefore, MNIST is assumed as a solid task for long time dependencies modeling (up to 784). There are 60000 training samples and 10000 testing samples.

### *5.2. System description*

Regarding the preprocessing, we leverage the mechanism [23] for word segmentation. Since English is delimited by whitespace, nevertheless, Chinese, Korean and any other Asian language are not. Therefore, segmentation is a fundamental step in the first step. For our study, we adopt the Viterbi algorithm [24] in segmentation.

The next step is word embedding. There are numerous methods for word embedding such as dense vector and sparse vector. We apply word2vec [18] for word embedding. Dense vector is more superior than sparse vector in terms of training speed and capacity of synonym capturing. Dense vectors are much shorter than sparse vectors, which means the computation power will be low. Besides, dense vectors obtained by word2vec reveal an attractive property that similar words in the vector matrix have a closer distance than others.

After selecting the features, the next step is to build the neural network. For better illustrating the improvement, various models are evaluated and compared such as plain RNNs, LSTMs, Bidirectional LSTM, 1-D CNNs, single residual and multi-residual networks. Two layers with 128 forward and 128 backward LSTM units are employed in our model. We also utilized gradient clipping [25]. All the weights are randomly initialized by the isotropic Gaussian distribution of variance 0.1. The dropout rate is

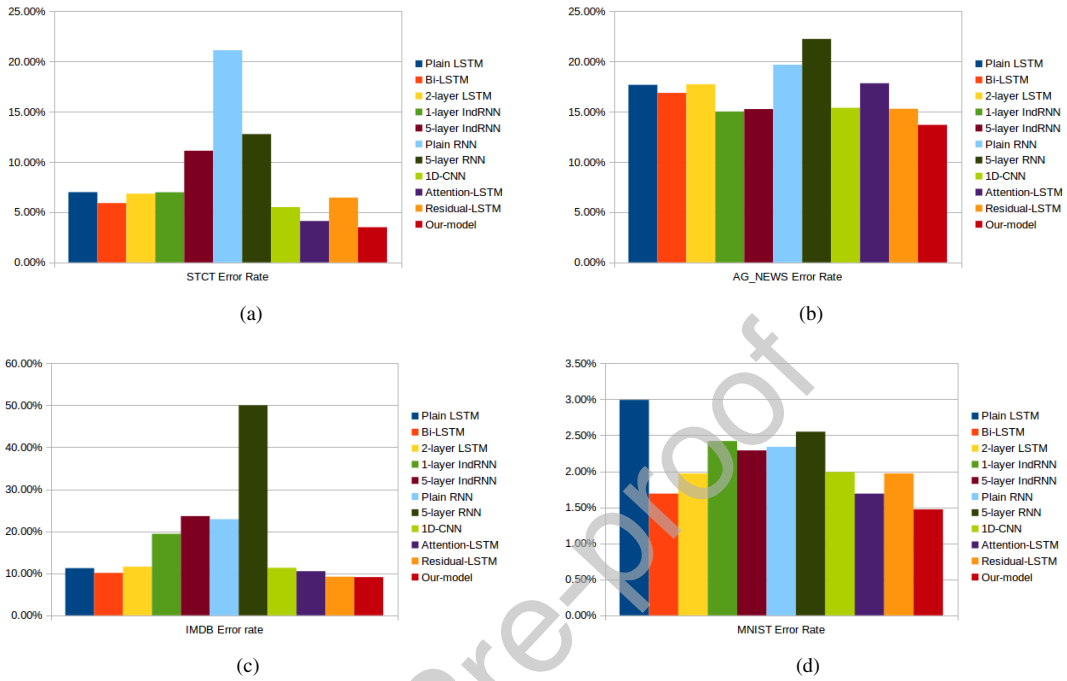



Figure 6: Error Rate

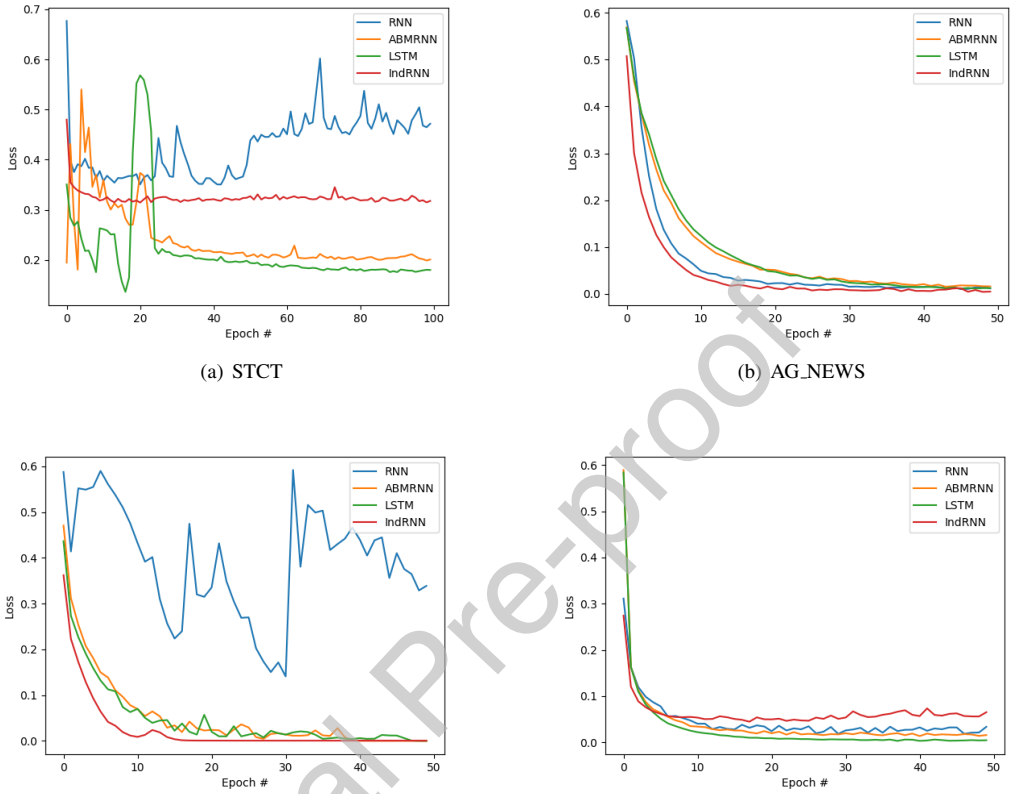
0.2 for each layer [26] and the batch size is 64. Regarding the other models, we keep the consistent setting with 128 hidden units in hidden layers.

### 5.3. *Result Analysis*

Overall results of the experiment are shown in Table 2. Our model achieves the state-of-the-art performance in STCT. In STCT, the highest accuracy rate is 96.5%, while the baseline performance provided is 69.03% [22]. We improve the ground truth about 39.7%, even the plain RNN model outperforms the statistical classification models (SVM 69.03% vs. Plain RNN 78.89%). Besides, with the model becoming more advanced, the corresponding accuracy rate is increasingly improved (Plain RNN 78.89% - LSTM 93.01% - Bi-LSTM 94.10%). We leverage the attention mechanism for optimizing the relationship between distant time steps, which improves the performance as well (plain LSTM 93.01% vs. attention LSTM 95.88%). We also attempt other architectures in STCT such as IndRNN [21], multi-layer RNNs. Our model outperforms all the existing methods.

We also concern about the training loss, because training loss can determine whether the model is converged or not, as well as provide the lower bound to estimate the performance. Those four models are selected because they represent the most typical structures. In Figure 7(a) and 7(c), plain RNNs keep oscillating, which means it is hard to converge. This is because compared with task AG_NEWS and MNIST, STCT and IMDB are much more complicated, which indicates RNNs cannot handle those hard tasks. However, ABMRNN, LSTM and IndRNN converge after only a few epochs in those four tasks, but LSTM converges slower than IndRNN. Although the training loss of ABMRNN is a little higher than that of LSTM, training loss can only guarantee the lower bound instead of the upper bound with regarding the accuracy rate.

AG's news corpus is a short-term text task which is similar to STCT. The accuracy rate in our model is 86.31%,




(a) STCT (b) AG_NEWS

(c) IMDB (d) MNIST

Figure 7: Training Loss

where outperforms the rest RNN models. However, 5-layer RNN obtains the lowest accuracy rate (77.76%). The accuracy rate in plain LSTM, 2-layer LSTM and bi-LSTM are 82.33%, 82.27% and 82.13% respectively. With adopted residual mechanism into LSTM, the performance improves correspondingly (from 82.33% to 84.71%).

In IMDB datasets, with the text length increasing up to 3000, compared with short-term text, the performance is impacted due to more redundancies and noises are introduced. We achieve the highest accuracy (90.91%) while the lowest accuracy is only 50% in 5-layer RNN. Because IMDB is a binary classification task, 50% accuracy rate means blind guess.

In MNIST, RNNs generally cannot perform as good as CNNs. We still apply models in MNIST because this is also a good task to test the robustness and reliability if we regard the images as sequential data. In sequential MNIST, the accuracy rate in 1-D CNN is only 98.01%. However, our model obtains the highest accuracy rate (98.53%) than other models because our model inherits the advantages from both residual network and LSTM.

For better illustrating the improvement of our model, we define Error Rate(ER) = 1 - accuracy and show the results with bar charts. Figure 6 shows the ER with four datasets. In STCT and AG's NEWS, ERs are decreased

about 503% (from 21.11% to 3.5%) and 62.5% (from 22.24% to 13.69%) respectively. In IMDB and MNIST, we finally improve the performance by 450% (from 50% to 9.09%) and 103% (from 2.99% to 1.47%) respectively.

Regarding recent neural networks towards AG's NEWS, IMDB and MNIST, all the models become increasingly complex with deeper layers. We illustrate the number of parameters of each model in Table 3. The number of parameters in [16] and [17] are 7.8M and 50M respectively. However, our model only contains 0.5M parameters. In AG's NEWS, [17] and [16] claim that their ER are 13.39% and 10.17% respectively. However, the ER in our model is 13.69%; our model demonstrates high efficiency in training (0.5M vs. 7.8M vs. 50M) and comparatively top performance (13.69% vs. 13.39% vs. 10.17%). Besides, in IMDB datasets, [27] and [28] declare that the ER are 11.52% and 11% respectively while the ER in our model is 9.19%. However, in [28], the depth of their model is 200 nevertheless ours is only two. In MNIST, although there are numerous famous CNN architectures such as ResNet (25.5M) AlexNet (60M) and VGGNet (138M), the depth of their models are considerably high (up to 1000). However, they perform from 95% to 97% variously because they mainly focus on the size of 224*224 while the size of the image in MNIST is 28*28. Therefore, [15] introduces a smaller architecture towards MNIST and achieves the state-of-the-art performance (ER = 0.23%). The number of parameters in [15] is 12M whereas ours is 0.5M. As we mentioned before, unlike all the 2D-CNN methods, we consider MNIST as a sequential classification task to test the model regarding robustness and reliability, and we outperform other RNN models.

| Model | #parameter |
|---|---|
| VDCNN[16] | 7.8M |
| CharCNN[17] | 50M |
| ResNet[7] | 25M |
| AlexNet [29] | 60M |
| APNN[15] | 12M |
| VGGNet [30] | 138M |
| **ABMRNN** | **0.5M** |

Table 3: The number of parameters

## 6. Conclusion and future work

In our research, an Attention-augmentation Bidirectional Multi-residual Recurrent Neural Network has been proposed. The memory cell limitation in LSTM has been resolved through a modified attention model. The specific previous time steps can be identified so that linking to it is most effective. Moreover, unlike randomly connecting to a previous time step, a multi-residue mechanism has been leveraged to enhance the correlation of current time step and distant previous time steps. In addition, a bidirectional multi-residual mechanism has been proposed which can combine both past and future information comprehensively, instead of partially capturing the previous information solely. Last but not the least, the detailed training procedure of ABMRNN has been made available. The results have shown that our model outperforms other RNN models such as plain RNN, LSTM and IndRNN, and achieved the state-of-the art performance in STCT. Compared with existing models, our model has demonstrated high efficiency in training and has achieved comparative top performance.

Our future work includes applying ABMRNN in other tasks, and further optimizing our model to handle longer sequence.

## References

[1] H. Zhang, J. Li, Y. Ji, H. Yue, Understanding subtitles by character-level sequence-to-sequence learning, IEEE Transactions on Industrial Informatics 13 (2) (2017) 616–624.

[2] F. Tao, C. Busso, Gating neural network for large vocabulary audiovisual speech recognition, IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP) 26 (7) (2018) 1286–1298.

[3] Y. Zhang, X. Zhang, Effective real-scenario video copy detection, in: Pattern Recognition (ICPR), 2016 23rd International Conference on, IEEE, 2016, pp. 3951–3956.

[4] Y. Wang, Y. Huang, W. Zheng, Z. Zhou, D. Liu, M. Lu, Combining convolutional neural network and self-adaptive algorithm to defeat synthetic

multi-digit text-based captcha, in: Industrial Technology (ICIT), 2017 IEEE International Conference on, IEEE, 2017, pp. 980–985.

[5] Y. Ji, H. Zhang, Q. J. Wu, Salient object detection via multi-scale attention cnn, Neurocomputing 322 (2018) 130–140.

[6] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural computation 9 (8) (1997) 1735–1780.

[7] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778.

[8] H. Wu, X. Zhang, B. Story, D. Rajan, Accurate vehicle detection using multi-camera data fusion and machine learning, in: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2019, pp. 3767–3771.

[9] Y. Wang, F. Tian, Recurrent residual learning for sequence classification, in: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, 2016, pp. 938–943.

[10] Y. Zhang, G. Chen, D. Yu, K. Yaco, S. Khudanpur, J. Glass, Highway long short-term memory rnns for distant speech recognition, in: Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on, IEEE, 2016, pp. 5755–5759.

[11] D. Bahdanau, K. Cho, Y. Bengio, Neural machine translation by jointly learning to align and translate, arXiv preprint arXiv:1409.0473.

[12] D. Bahdanau, J. Chorowski, D. Serdyuk, P. Brakel, Y. Bengio, End-to-end attention-based large vocabulary speech recognition, in: Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on, IEEE, 2016, pp. 4945–4949.

[13] F. Tao, G. Liu, Q. Zhao, An ensemble framework of voice-based emotion recognition system for films and tv programs, arXiv preprint arXiv:1803.01122.

[14] J. K. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, Y. Bengio, Attention-based models for speech recognition, in: Advances in neural information processing systems, 2015, pp. 577–585.

[15] I. Sato, H. Nishimura, K. Yokoi, Apac: Augmented pattern classification with neural networks, arXiv preprint arXiv:1505.03229.

[16] A. Conneau, H. Schwenk, L. Barrault, Y. Lecun, Very deep convolutional networks for text classification, arXiv preprint arXiv:1606.01781.

[17] X. Zhang, J. Zhao, Y. LeCun, Character-level convolutional networks for text classification, in: Advances in neural information processing systems, 2015, pp. 649–657.

[18] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, J. Dean, Distributed representations of words and phrases and their compositionality, in: Advances in neural information processing systems, 2013, pp. 3111–3119.

[19] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, Y. Bengio, Learning phrase representations using rnn encoder-decoder for statistical machine translation, arXiv preprint arXiv:1406.1078.

[20] F. Tao, G. Liu, Advanced lstm: A study about better time dependency modeling in emotion recognition, in: 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2018, pp. 2906–2910.

[21] S. Li, W. Li, C. Cook, C. Zhu, Y. Gao, Independently recurrent neural network (indrnn): Building a longer and deeper rnn, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 5457–5466.

[22] Y. Wang, Z. Zhou, S. Jin, D. Liu, M. Lu, Comparisons and selections of features and classifiers for short text classification, in: Materials Science and Engineering Conference Series, Vol. 261, 2017, p. 012018.

[23] C. Huang, H. Zhao, Chinese word segmentation: A decade review, Journal of Chinese Information Processing 21 (3) (2007) 8–20.

[24] G. D. Forney, The viterbi algorithm, Proceedings of the IEEE 61 (3) (1973) 268–278.

[25] G. Hinton, N. Srivastava, K. Swersky, Neural networks for machine learning lecture 6a overview of mini-batch gradient descent, Cited on (2012) 14.

[26] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, R. Salakhutdinov, Dropout: a simple way to prevent neural networks from overfitting, The Journal of Machine Learning Research 15 (1) (2014) 1929–1958.

[27] A. Tripathy, A. Agrawal, S. K. Rath, Classification of sentiment reviews using n-gram machine learning approach, Expert Systems with Applications 57 (2016) 117–126.

[28] S. Karimi, X. Dai, H. Hassanzadeh, A. Nguyen, Automatic diagnosis coding of radiology reports: a comparison of deep learning and conventional classification methods, BioNLP 2017 (2017) 328–332.

[29] A. Krizhevsky, I. Sutskever, G. E. Hinton, Imagenet classification with deep convolutional neural networks, in: Advances in neural information processing systems, 2012, pp. 1097–1105.

[30] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, arXiv preprint arXiv:1409.1556.

**Biography**

**Ye Wang** received B.S. degree in Microelectronics from Chongqing University of Posts and Telecommunications, Chongqing, China in 2011 and the M.S. degree in Electrical Engineering from University of Texas at Dallas, Richardson, TX, USA in 2014. He is currently pursuing the Ph.D. degree with the Department of Electrical and Computer Engineering, Texas A&M University, College Station, TX, USA. His current research is mainly focused on word embedding in natural language processing and machine learning.

**Xinxiang Zhang** received the B.E. Degree from Communication University of China, Beijing, China in 2014 and the M.S. Degree from Boston University, Boston, MA, USA in 2016, both in Electrical Engineering. He is currently working toward his Ph.D. degree in Electrical Engineering from Southern Methodist University, Dallas, TX, USA. His current research is mainly focused on computer vision applications on civil engineering and intelligent transportation system.

**Mi Lu (S'84-M'86-SM'94)** received the M.S. and Ph.D. degrees in electrical engineering from Rice University, Houston, in 1984 and 1987, respectively. She joined the Department of Electrical Engineering at Texas A&M University in 1987 where she is currently a professor. Her research interests include parallel computing, distributed processing, parallel computer architectures and algorithms, computer networks, computer arithmetic, computational geometry, and VLSI algorithms. She has published more than 100 technical papers in these areas. She has served as an Associate Editor of the Journal of Computing and Information and the Information Sciences Journal. She was the Stream Chair of the Seventh International Conference of Computing and Information, and the Conference Chair of the Fifth and Sixth International Conference on Computer Science and Informatics. She served on panels of the U.S. National Science Foundation and the 1992 IEEE Workshop on Imprecise and Approximate Computation and on many conference program committees. She is the chair of 60 research advisory committees for Ph.D. and Master's students. She is a registered professional engineer; and is recognized in Who's Who in the World, 2001, and Who's Who in America, 2002.

**Han Wang** is a PhD student in computer science at Texas A&M University. He received B.S. in Mathematics from Xi'an Jiaotong University, China. His research interests include recurrent neural network dynamics, natural language processing (NLP), and human computer interaction (HCI). In his spare time, he likes to participate in coding contests, open-source projects, and e-sports.

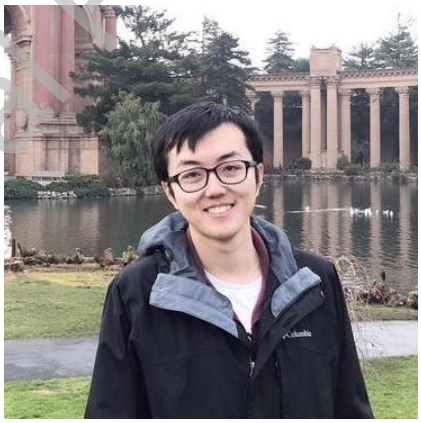
**Yoonsuck Choe (M'06-SM'14)** is currently a Professor and the Director of the Brain Networks Laboratory with the Department of Computer Science and Engineering, Texas A&M University, College Station, TX, USA. His current research interests include computational neuroscience, computational neuroanatomy, neuroinformatics, neural networks, and neuroevolution. His work ranges from visual cortical modeling, sensorimotor learning, temporal aspects of brain function (delay, memory, and prediction), whole brain physical sectioning imaging (knife-edge scanning microscopy), and web-based brain atlas frameworks

Ye Wang

Xinxiang Zhang

Mi Lu

Han Wang



Yoonsuck CHoe