Text Sentiment Classification Model Based on BiGRU-Attention Neural Network (Postprint)

Authors: Wang Wei, Sun Yuxia, Qi Qingjie, Meng Xiangfu

Date: 2018-10-11T00:00:00+00:00

Abstract

To address the issues of prolonged training time and inadequate learning of textual context information in the Bidirectional Long Short-Term Memory (BiLSTM) model, we propose a text sentiment classification model based on BiGRU-Attention. First, a Bidirectional Gated Recurrent Unit (BiGRU) neural network layer is employed to extract features from deep-level textual information. Second, an attention mechanism layer is utilized to assign appropriate weights to the extracted deep-level textual information. Finally, the text feature information with varying weights is fed into a softmax function layer for text sentiment polarity classification. Experimental results show that the proposed neural network model achieves an accuracy of 90.54% on the IMDB dataset, with a loss rate of 0.2430 and a time cost of 1100 s, thereby validating the effectiveness of the BiGRU-Attention model.

Full Text

Preamble

Title: Text Sentiment Classification Model Based on BiGRU-Attention Neural Network

Authors: Wang Wei^{a,b}, Sun Yuxia^{b}, Qi Qingjie^{c}, Meng Xiangfu^{b}

Affiliations: ^{a}College of Science & Technology, ^{b}College of Electronic & Information Engineering, ^{c}College of Mining, Liaoning Technical University, Huludao, Liaoning 125105, China

Abstract: The bidirectional long short-term memory (BiLSTM) neural network model suffers from long training times and cannot fully learn contextual information from text. To address these problems, this work proposes a text sentiment classification model based on a BiGRU-Attention neural network. First,

a bidirectional gated recurrent unit (BiGRU) neural network layer extracts features from deep text information. Second, an attention mechanism layer allocates corresponding weights to the extracted deep text information. Finally, text feature information with different weights is fed into a softmax function layer for text sentiment polarity classification. Experimental results demonstrate that the proposed neural network model achieves 90.54% accuracy on the IMDB dataset, with a loss rate of 0.2430 and a time cost of 1100 seconds, thereby verifying the effectiveness of the BiGRU-Attention model.

Keywords: text sentiment classification; attention mechanism; bidirectional gated recurrent unit (BiGRU)

0 Introduction

Text sentiment classification is an important task in natural language processing (NLP) that typically involves classifying texts containing subjective emotional color. It can extract hierarchical text features and mine users' emotional tendencies, with wide applications in policy opinion mining, public opinion surveys, product analysis, movie recommendations, search ranking, and other domains. In recent years, the rise of Web 2.0 has ushered in an era where users actively create information, enabling them to express their opinions and comments anytime through mobile terminals. Consequently, user review data has grown exponentially. Initial shallow machine learning algorithms such as Naive Bayes that researchers employed can no longer meet the needs of this ever-increasing data processing demand, making neural networks emerge as a research hotspot in recent years.

CNN was first widely applied in the image domain and rapidly expanded to other fields. LeCun et al. [**?**] applied CNN to text sentiment classification, improving classification accuracy. Although CNN achieved higher accuracy compared to traditional rule-based and machine learning algorithms [**?**], it still cannot fully learn contextual information from text. As research progressed, Mikolov et al. [**?**] proposed applying RNN to text classification tasks. Since the output value of a current node in RNN is jointly determined by the current input and the previous node' s output, RNN can fully learn information from both preceding and following contexts, making it more suitable for text classification than CNN. However, RNN suffers from gradient vanishing problems when learning long-term dependencies. To solve this issue, numerous variants such as LSTM and GRU have been proposed and widely applied in sentiment classification. Nevertheless, due to its structural complexity, LSTM involves exceptionally complicated computations and stores redundant intermediate variables, requiring substantial training time and memory space. Moreover, LSTM and GRU can only utilize historical information to make judgments about current information and cannot leverage future information, sometimes resulting in inaccurate judgments and insufficient text information extraction. Currently, widely used

models combining BiLSTM and attention mechanisms still cannot escape the problem of long training times caused by complex computations.

To address the aforementioned issues, this paper proposes a BiGRU-Attention model. By feeding text input vectors into a BiGRU layer instead of a BiLSTM layer, network training time can be further reduced. Feeding the output vectors from the BiGRU layer into an attention layer can further highlight key text information and improve the quality of text information extraction. The BiGRU-Attention model conducts comparative experiments on the IMDB dataset, using accuracy, loss rate, and time cost as evaluation metrics to demonstrate the effectiveness of the model in text classification.

1 Related Work

Research on text sentiment classification using deep learning methods has mushroomed in recent years. Liang et al. [**?**] employed a recursive neural network with an emotional transfer model to enhance the capture of text relevance. Bai et al. [**?**] proposed a hybrid neural network structure using BiLSTM-CNN-Attention for fusing two types of features. Liang et al. [**?**] adopted a hybrid model combining attention mechanisms and CNN to solve parallelization problems and reduce model training time. Zhao et al. [**?**] proposed an LSTM-Attention model that fully extracts semantic structure information. Yang et al. [**?**] presented a hybrid neural network model combining CNN, GRU, and attention mechanisms for text feature extraction. Si et al. [**?**] proposed a hybrid model of attention mechanism and LSTM for more effective Chinese part-of-speech tagging. Wang et al. [**?**] used LSTM neural networks for judgment result tendency analysis tasks, effectively improving the accuracy of key information extraction in judicial documents. Zhu et al. [**?**] proposed an improved attention-based LSTM feature selection model that solves the dimensionality curse problem and effectively highlights key text feature information. Li et al. [**?**] applied denoising autoencoder deep learning methods to sentiment analysis tasks, improving model robustness to original data and expressive power of information features. Liu [**?**] used GRU neural networks for time series prediction, improving prediction accuracy. Li et al. [**?**] employed BiGRU networks for rapid and accurate extraction of specific information from internet input sequences. Huang et al. [**?**] used a method combining GRU and attention mechanisms for distant supervision relationship extraction, improving accuracy. Zhang et al. [**?**] built text sentiment classification models using LSTM and GRU, enabling the model to achieve high accuracy in a short time. Tian [**?**] applied RSGRU hybrid neural networks to sentence-level sentiment analysis tasks, saving manpower and making maintenance easier. Zhou et al. [**?**] adopted minimal gated units (MGU) for processing sequential data, accelerating model training speed. Huang et al. [**?**] used a hybrid neural network model combining LSTM and GRU for extracting key text information, significantly improving recall rate.

In recent years, attention mechanisms have been widely applied in natural language processing. Attention mechanisms were first applied in computer vision

and image processing. In 2014, the Google Mind team [**?**] used a hybrid model of recurrent neural networks and attention mechanisms for image classification. The main idea is that when observing an image, people often concentrate their attention on a small region and can predict where attention should focus on the image next based on previous observations. The model determines which part of the image should be processed at the current state based on current input and previous state, processing fewer pixels and simplifying the task. Subsequently, Bahdanau et al. [**?**] applied attention mechanisms to machine translation tasks, using attention to connect the expression of words to be translated with the prediction of words to be translated, marking the first application of attention mechanisms in NLP. Luong et al. [**?**] introduced global and local attention mechanisms. Yin et al. [**?**] introduced three ways to combine convolution and attention mechanisms: adding attention before CNN input, adding attention after CNN feature extraction but before pooling, or combining both approaches, representing the first exploration of attention mechanisms in CNNs. Attention mechanisms became popular in machine translation between 2014-2015. In 2015, neural network models combining attention mechanisms and RNN were widely applied in NLP. Between 2015-2016, neural network models combining attention mechanisms and CNN became a research hotspot. Subsequently, researchers have applied attention mechanisms and neural network models to text sentiment classification in an endless stream, becoming a research hotspot in recent years.

After LSTM, CNN, attention mechanisms, and other hybrid neural network models were widely applied in text sentiment classification and achieved remarkable results in models combining LSTM and attention mechanisms, researchers gradually turned their attention to studies combining attention mechanisms with BiLSTM and BiGRU. Such research has emerged domestically and internationally in the past two to three years. Tian et al. [**?**] used a hybrid neural network combining BiLSTM and attention mechanisms to identify temporal relationships in Uyghur events. This hybrid approach first feeds text vectors into a BiLSTM layer to extract some text information, then into an attention mechanism layer for deep text feature extraction, and finally into a softmax layer for text sentiment classification. Si et al. [**?**] proposed an LSTM-BiLSTM-Attention network model for more effective Chinese part-of-speech tagging, first using an LSTM layer for text feature extraction, then using the bidirectional principle of BiLSTM to tag target words from both directions, and finally using an attention layer to further extract key information and increase part-of-speech tagging accuracy. Cheng [**?**] proposed a neural network model based on attention mechanisms and BiLSTM for sentiment analysis of Chinese product reviews, feeding segmented word vectors into BiLSTM for text feature extraction and into an attention layer to highlight key information for text classification. Rozental et al. [**?**] proposed a hybrid model using BiGRU neural networks and multiple convolutional max pooling operations to extract text feature information, finally classifying text in the softmax layer, achieving first and third place results when tested with English and Spanish. Chen et al. [**?**] used a hybrid neural

network combining BiLSTM and position attention mechanisms for text classification, also achieving good classification results. Kumar et al. [**?**] used a simple hybrid model of BiLSTM and two-layer attention mechanisms, building word-level and sentence-level attention mechanisms sequentially after the BiLSTM layer extracted text feature information. Evaluated on SemEval 2017 Task 5, this experimental model improved results by 1.7 and 3.7 percentage points over the current best system in sub-tracks 1 and 2.

Whether analyzing Uyghur event temporal relationships, Chinese part-of-speech tagging, product review analysis, or the latest international research on BiLSTM or BiGRU, these studies essentially use neural network models combined with attention mechanisms to process text and extract deep text information. Most employ hybrid models combining BiLSTM or BiGRU with attention mechanisms, and most models go through three stages: feeding text vectors into a BiLSTM layer for feature extraction, an attention mechanism layer to highlight key information, and a softmax layer for text classification. These studies have focused heavily on research combining BiLSTM or BiGRU neural networks with attention, achieving good results. Building on previous research, this paper proposes a simpler model that uses a more concise structure, simpler computation, and smaller storage space by combining BiGRU layers, attention layers, softmax layers, and fully connected layers in a simple hybrid neural structure to extract feature information for text classification.

2.1.1 Long Short-Term Memory Neural Network

RNN is a special neural network for processing sequential data, functioning similarly to certain human thinking habits. Imagine the scenario of completing English CET-4 or CET-6 cloze tests: to determine which word to fill in a blank, we must read not only the words before the blank but also the words before those. For example, in "The sea water in the deep sea is very _____," reading "very" and the preceding "in the deep sea" reveals that "deep" should be selected. Similarly, RNN was born to solve the relationship between current output and current input plus previous output in text. RNN is mainly applied in speech recognition, machine translation, Chinese word segmentation, part-of-speech tagging, and other sequential data domains, achieving good results in these areas. With ongoing research, RNN has been widely applied to text processing, and in recent years, models used in sentiment analysis have emerged in an endless stream.

The classic RNN structure consists of an input layer, hidden layer, and output layer. RNN can learn semantic information from text context, and the extracted feature information can serve as input to other neural networks or models, or be fed directly into a softmax function layer for sentiment polarity classification. The RNN structure unfolded at time t is shown in Figure 1 [Figure 1: see original paper].

The basic algorithmic idea of RNN is backpropagation through time. However,

during this process, gradients across time steps and long-term learning often cannot propagate from initial values to the earliest positions, making gradient vanishing problems likely to occur. To overcome this disadvantage, numerous RNN variants have been proposed, among which LSTM is a classic variant widely applied. The specific structure of a single LSTM neuron is shown in Figure 2 [Figure 2: see original paper].

The specific working principle of LSTM can be understood through the following formulas:

[1] f_t = sigmoid(W_f · [h_{t-1}, x_t] + b_f) [2] i_t = sigmoid(W_i · [h_{t-1}, x_t] + b_i) [3] C̃*t = tanh(W_C · [h_{t-1}, x_t] + b_C) [4] C_t = f_t * C*{t-1} + i_t * C̃_t [5] o_t = sigmoid(W_o · [h_{t-1}, x_t] + b_o) [6] h_t = o_t * tanh(C_t)

As shown in Figure 2 and the formulas, LSTM consists of four components: input gate, memory cell, output gate, and forget gate. Here, x_t represents the input vector at time t, h_{t-1} represents the output vector from the previous time step, W represents the weight coefficient matrices for each corresponding component, b represents the offset vectors for each corresponding component, and sigmoid denotes the activation function. Equation (1) calculates the forget gate value, determining how much information can be retained. From the form of equation (1), we can see that the forget gate value at time t is jointly determined by h_{t-1} and x_t. Equation (2) calculates the cell state value activated by the sigmoid function. Equation (3) calculates the candidate memory unit value determined by h_{t-1} and x_t. Equation (4) calculates the memory state unit value after regulation by f_t and i_t on C_{t-1} and C̃*t. Equations (5) and (6) calculate the final hidden state output of LSTM at time t, determined by o_t and h*{t-1}.

2.1.2 Gated Recurrent Unit

With the widespread application of LSTM in natural language processing, particularly in text classification tasks, people gradually discovered that LSTM has disadvantages including long training times, numerous parameters, and complex internal computations. In 2014, Cho et al. further proposed the GRU model, which is simpler, merges the cell state and hidden state of LSTM, and includes other modifications. The GRU model maintains LSTM's effectiveness while having a simpler structure, fewer parameters, and better convergence. The GRU model consists of two gates: an update gate and a reset gate. The update gate controls the degree to which the previous time step' s hidden output affects the current hidden layer—the larger the update gate value, the greater the influence. The reset gate controls the degree to which previous hidden layer information is ignored—the smaller the reset gate value, the more information is ignored.

The GRU structure is more streamlined, as shown in Figure 3 [Figure 3: see original paper]. The update method for the GRU model is as follows:

[7] r_t = σ(W_r ·[h_{t-1}, x_t]) [8] z_t = σ(W_z ·[h_{t-1}, x_t]) [9] ħ*t* = *tanh(W_h ·[r_t * h_{t-1}, x_t]) [10] h_t = (1 - z_t) * h*{t-1} + z_t * ħ_t

Where r_t represents the reset gate at time t, z_t represents the update gate at time t, ħ*t represents the candidate activation state at time t, h_t represents the activation state at time t, and h*{t-1} represents the hidden state at time (t-1). The update gate z is determined by the historical information that needs to be forgotten and the new information that needs to be accepted at the current state. The reset gate r is determined by the information obtained from historical information in the candidate state.

2.1.3 BiGRU

In unidirectional neural network structures, states are always output from front to back. However, in text sentiment classification, if the current time step' s output can connect with both previous and subsequent states—for example, when completing a cloze test for the sentence "The sea water in the deep sea is so _____ that the sun does not shine" —by reading both "The sea water in the deep sea" and "the sun does not shine," we can be more confident that "deep" should be filled in. This facilitates the extraction of deep text features and requires BiGRU to establish this connection. BiGRU is a neural network model composed of unidirectional, opposite-direction GRUs whose outputs are jointly determined by the states of these two GRUs. At each time step, input is simultaneously provided to two opposite-direction GRUs, and the output is jointly determined by these two unidirectional GRUs. The specific structure of BiGRU is shown in Figure 4 [Figure 4: see original paper].

As shown in Figure 4, the current hidden state of BiGRU is jointly determined by the current input x_t, the forward hidden state output h_{t-1} at time (t-1), and the backward hidden state output h_{t+1}. Since BiGRU can be viewed as two unidirectional GRUs, the hidden state of BiGRU at time t is obtained through weighted summation of the forward hidden state and backward hidden state:

[11] h_t = w_t * h_t + v_t * h_t + b_t

Where GRU() represents the nonlinear transformation of input word vectors, encoding them into corresponding GRU hidden states. w_t and v_t represent the weights corresponding to the forward hidden state and backward hidden state of the bidirectional GRU at time t, respectively, and b_t represents the bias corresponding to the hidden state at time t.

2.1.4 Attention Model

Attention mechanisms have demonstrated remarkable performance in sequential data such as speech recognition, machine translation, and part-of-speech tagging. Attention mechanisms can be used alone or as layers in other hybrid

models. They can be placed after the text vector input layer or after other network model training data. Through automatic weighted transformation of data, they connect two different parts, highlight key words, and enable the entire system to exhibit better performance. The attention mechanism resembles the human brain' s principle of observing certain things. For example, when describing a painting, people first observe the inscription, then purposefully observe the part of the image that expresses the theme. When describing the painting, they typically describe the most relevant content first before addressing other aspects. The attention mechanism is a mechanism that allocates sufficient attention to key information and highlights locally important information. Attention mechanisms can generally be divided into two categories: temporal attention mechanisms and spatial attention mechanisms. This paper primarily uses temporal attention.

The attention mechanism is similar to the human brain' s attention resource allocation mechanism. Through probability weight distribution, it calculates the probability weights of word vectors at different time steps, enabling some words to receive more attention and thereby improving the quality of hidden layer feature extraction. The basic structure of the attention model is shown in Figure 5 [Figure 5: see original paper].

In the attention model, the vector s from the initial hidden state to the new hidden state is the cumulative sum of the product of the weight coefficients α_i, which represent the proportion of each hidden state in the new hidden state, and the initial input hidden states h_i. The calculation formula is:

[12] s = Σ_{i=1}^{n} α_i h_i

[13] αi = *exp(e_i) / Σ*{j=1}^{n} exp(e_j)

[14] e_i = v_i^T tanh(W_w h_i + b_w)

Where e_i represents the energy value determined by the hidden state vector h_i at time i, W_w and v_i represent the weight coefficient matrices at time i, and b_w represents the corresponding offset at time i. Through equation (9), the transformation from the initial input state to the new attention state can be achieved.

2.2 BiGRU-Attention Model

The BiGRU-Attention model consists of three parts: a text vectorization input layer, a hidden layer, and an output layer. The hidden layer comprises three sublayers: a BiGRU layer, an attention layer, and a Dense layer. The structure of the BiGRU-Attention model is shown in Figure 6 [Figure 6: see original paper].

The functions of these three layers are described below:

1) **Input Layer**: The text vectorization input layer primarily preprocesses 25,000 IMDB movie review data items, transforming these reviews into sequence vector forms that can be directly received and processed by the

BiGRU layer. A text a consisting of m words forming l sentences is represented as a = {s_1, s_2, ···, s_l}, where the j-th sentence in the sample is represented as s_j = {w_{1j}, w_{2j}, ···, w_{mj}}. Text vectorization involves specific operations to make w_{ij} ^d. The specific steps are as follows:

a) Read and clean the data;
b) Vectorize the data to a fixed length of 1,000 (sentences shorter than the specified value are automatically padded with special symbols at the end; sentences longer than the specified value keep only the first 1,000 words, with the excess truncated);
c) Randomly initialize the data and split it into training and test sets at an 8:2 ratio;
d) After vectorization, each movie review becomes a uniform-length index vector, where each index corresponds to a word vector;
e) Concatenate word vectors to form word matrices.

After these four steps, the input IMDB data becomes a word matrix according to index-to-word-vector correspondence. Assuming the unified length of processed word vectors is 1,000, using the 100-dimensional vector form of glove.6B.100d, word vectors not found in glove.6B.100d are randomly initialized. Let c_{ij} be the word vector of the i-th word in the j-th sentence. Then an IMDB review of length 1,000 can be represented as:

[15] c_{1:1000}^j = c_1^j c_2^j ··· c_{1000}^j

Where represents the concatenation operator for word vectors, and c_{1:1000}^j is the word vector matrix for the j-th sentence. Each word in every IMDB review is mapped to its corresponding word vector in glove.6B.100d according to its index, generating a word vector matrix.

2) **Hidden Layer**: The hidden layer calculations are completed in two main steps:

a) Calculate the word vectors output by the BiGRU layer. The text word vectors serve as input to the BiGRU layer. The purpose of the BiGRU layer is to extract deep hierarchical features from the input text vectors. According to the BiGRU neural network model diagram, the BiGRU model can be viewed as consisting of forward GRU and backward GRU components, simplified here as equation (11). At time i, the word vector of the t-th word in the j-th sentence is c_{ijt}. After feature extraction through the BiGRU layer, the model can more fully learn relationships between contexts for semantic encoding. The specific calculation formula is shown in equation (11):

[16] h_{ijt} = BiGRU(c_{ijt}) ^m

b) Calculate the probability weights to be allocated to each word vector. This step primarily assigns corresponding probability weights to different word

vectors to further extract text features and highlight key text information. In text, different words play different roles in sentiment classification. Locative and temporal adverbials have minimal importance for text sentiment classification, while emotionally colored adjectives are crucial. To highlight the importance of different words for overall text sentiment classification, the BiGRU-Attention model introduces an attention mechanism layer. The input to the attention mechanism layer is the output vector processed by the previous BiGRU neural network layer. The weight coefficients of the attention mechanism layer are calculated through the following formulas:

[17] u_{ijt} = tanh(W_w h_{ijt} + b_w) [18] $\alpha\{ijt\} = exp(u\{ijt\}\hat{}T\ u\_w) / \Sigma\_t\ exp(u\_\{ijt\}\hat{}T\ u\_w)$ [19] $s\_\{ij\} = \Sigma\_t\ \alpha\{ijt\}\ h\{ijt\}$

Where h_{ijt} is the output vector from the previous BiGRU neural network layer, W_w represents the weight coefficient, b_w represents the bias coefficient, and u_w represents the randomly initialized attention matrix. The attention mechanism matrix is the cumulative sum of the product of different probability weights allocated by the attention mechanism and each hidden state, using the softmax function for normalization.

3) **Output Layer**: The input to the output layer is the output from the previous attention mechanism layer. The output layer uses the softmax function to calculate classification results from its input. The specific formula is:

[20] y_j = softmax(W_1 s_{ij} + b_1)

Where W_1 represents the weight coefficient matrix to be trained from the attention mechanism layer to the output layer, b_1 represents the corresponding bias to be trained, and y_j represents the output predicted label.

2.3 BiGRU-Attention Model Training Method

This BiGRU-Attention model takes the IMDB dataset, preset parameters, and iteration number N as inputs, processes the IMDB dataset into word vector form through the text vectorization input layer, and classifies the IMDB dataset using the BiGRU-Attention model.

Algorithm: BiGRU-Attention Neural Network Text Sentiment Classification Algorithm

Input: IMDB dataset, preset parameters, iteration number N.

Output: Sentiment classification of the IMDB dataset.

1. The text vectorization input layer cleans the data, splits it into fixed lengths, randomly initializes it, divides it into training and test sets, and converts it into corresponding word vectors.
2. For each movie review in the IMDB dataset:

- For hop = 1 to h:
  a) Use equation (11) to extract text hierarchical features;
  b) Use equations (17)-(19) to assign corresponding weights to text vectors.

3. Use equation (20) to calculate the classification result probabilities with the softmax function.
4. Compare with the original labels. The objective function of this paper is: [21] loss = -Σ_j ŷ_j log(y_j)
5. End for.

From equation (14), through the above training steps, features are extracted for words from 1 to h, corresponding weights are assigned and summed, the Dense layer further extracts features, and finally classification is performed in the softmax output layer. The product of each classified movie review label value and ŷ_j is accumulated (the sum is negative, so the negative is taken to minimize loss), errors are calculated, and rmsprop is used as the optimizer to make model training and convergence faster. During backpropagation through time, weights and biases are continuously adjusted and updated according to errors until the iteration number is reached or a fixed precision is achieved.

3.1 Experimental Setup

To verify the effectiveness of the BiGRU-Attention model, the public IMDB dataset was selected. The training and test sets were split at an 8:2 ratio for model training and testing, as detailed in Table 1 .

Table 1: Dataset

Dataset	Number of Samples
Training Set	20,000
Test Set	5,000

This experiment uses accuracy, loss rate, and iteration time as evaluation criteria. Data preprocessing is the most important step before experimental data is fed into neural network training. The text vector layer has already provided detailed discussion of data processing, so it will not be restated here.

This experiment uses the Keras deep learning framework with TensorFlow as the backend, implemented in Python. The experimental environment includes JetBrains PyCharm software, Windows 10 system, and 8 GB RAM. The experimental model has a three-layer network structure: text vector input layer, hidden layer, and text classification layer. Many hyperparameters need to be set and adjusted during the experiment, with adjustments made after each iteration based on experimental accuracy and loss rate. After multiple iterations, the hyperparameters were set as shown in Table 2 .

Table 2: Model Parameter Settings

Parameter	Value
BiGRU Hidden Layer Nodes	128
Loss Function	Categorical_{crossentropy}
Optimizer	rmsprop
Batch_{size}	128
Word Vector Dimension	100

3.2 Experimental Evaluation Standards

This paper adopts accuracy, loss rate, and iteration time as experimental evaluation standards. Assuming the total number of samples is M and the number of correctly classified samples is m, the accuracy is:

[22] Accuracy = m / M

The loss function is obtained during each random batch training process as the negative of the cumulative product of batch samples according to equation (14). Iteration time is the average of the sum of each iteration' s time during 10 iterations.

3.3 Experimental Steps

The specific experimental steps for the BiGRU-Attention model are as follows:

a) Perform data cleaning;
b) Uniformize into fixed-length index vector form;
c) Split into training and test sets;
d) Map each index vector to word vectors;
e) Concatenate word vectors to form word matrices as input to the BiGRU-Attention model;
f) Feed the text input matrix into the BiGRU layer, using equation (11) for model training to extract text hierarchical features;
g) Feed into the attention model to assign corresponding weights to text vectors;
h) Evaluate BiGRU-Attention model performance using the IMDB test set.

3.4 Comparison Experiments

The BiGRU-Attention model is compared with seven common models. The structures of these seven models are as follows:

a) **ASC**: Rozental et al. [**?**] proposed a network model using BiLSTM for feature extraction and dimensionality reduction, max pooling layers to highlight key information, and fully connected layers for dimensionality reduction. Four differently processed vectors are concatenated and fed into a fully connected layer, followed by softmax classification.

b) **DBLSTM-Attention**: Chen et al. [**?**] proposed a hybrid experimental model combining BiLSTM and position attention mechanisms.

c) **BiLSTM-Attentions**: Kumar et al. [**?**] proposed a hybrid experimental model combining BiLSTM and two-layer attention mechanisms.

d) **HDBN Model**: Yan et al. [**?**] proposed a hybrid neural network model (HDBN) that primarily uses DBM for noise reduction and dimensionality reduction, DBN for hierarchical feature extraction, and a softmax layer for text classification.

e) **S-LSTM**: Zhu et al. [**?**] proposed a hybrid model using tree-structured LSTM and memory cells to remember historical information.

f) **BiLSTM-Attention Model**: The BiLSTM-Attention model is the same type of hybrid experimental model as the BiGRU-Attention model. It simply replaces the BiGRU layer in the BiGRU-Attention model with a BiLSTM layer, with all other experimental settings remaining identical.

g) **LSTM-Attention Model**: The LSTM-Attention model, BiLSTM-Attention model, and BiGRU-Attention model are all the same type of hybrid experimental model. The LSTM-Attention model replaces the BiGRU layer in the BiGRU-Attention model with an LSTM layer, with all other experimental settings remaining identical.

All the above experimental models basically have a three-layer network structure: text vector input layer, hidden layer (text information extraction function layer), and text classification layer. Hyperparameters are identical across models, with 10 iterations. From this analysis, except for the experimental variable layer (the model function layer), all other conditions are the same across the four models, ensuring the uniqueness of the experimental variable layer and the targeted comparability of experimental results.

3.5 Experimental Results Analysis

3.5.1 Comparison with HDBN, S-LSTM, ASC, DBLSTM-Attention, and BiLSTM-Attentions Models

This experiment selects the highest accuracy on the test set during 10 iterations as the model' s accuracy. The loss rate corresponding to the test set' s highest accuracy is taken as the model' s loss rate, and the corresponding iteration time is recorded as the model' s time cost, as detailed in Table 3 .

Table 3: Classification Results of Different Models

Model	Accuracy	Loss Rate	Time Cost
BiGRU-Attention	90.54%	0.2430	1100s
ASC	89.96%	0.2445	1115s
BiLSTM-Attentions	90.34%	0.2456	1345s

Model	Accuracy	Loss Rate	Time Cost
DBLSTM-Attention	88.85%	0.2567	1205s
S-LSTM	88.74%	0.2612	1417s
HDBN	86.94%	0.2712	1221s

As shown in Table 3, the proposed BiGRU-Attention model outperforms the other five experimental models in terms of accuracy and loss rate. The time cost is higher than the ASC model but lower than the other four models. Overall, the accuracy, loss rate, and time cost of these six models are relatively close, with these metrics varying and fluctuating according to model complexity. Comparing from top to bottom in the table, we can see that the BiGRU-Attention model' s accuracy and loss rate performance is superior to the ASC model. Both models utilize BiGRU neural network layers to extract key text information. The difference lies in that one adds an attention layer on top of the BiGRU layer to allocate corresponding weights to highlight key text features, while the other adds a max pooling layer on top of the BiGRU layer to highlight important information and forget secondary information through max pooling' s characteristics. The comparison shows that the attention layer' s performance in highlighting important information is superior to convolutional max pooling, as evidenced by higher accuracy and lower loss rate in experimental evaluation metrics. However, because the attention layer' s information highlighting function is achieved through continuous weighted calculations, it increases time cost compared to simple max pooling. The BiGRU-Attention model' s experimental evaluation metrics are superior to DBLSTM-Attention. These two models both have attention layers with basically the same main function and similar model structure, but differ in that the first layer is BiGRU in one and BiLSTM in the other. Through observation and analysis of these two models' evaluation metrics, we can see that BiGRU neural networks outperform BiLSTM neural networks in extracting key text information and reducing computation. Comparing BiLSTM-Attentions with DBLSTM-Attention, the difference is that BiLSTM-Attentions has one more attention layer than DBLSTM-Attention. Analysis of experimental evaluation metrics clearly shows the important role of the attention layer in highlighting text information.

In summary, through comparative analysis of the above models, we can draw the conclusion that on the IMDB dataset, BiGRU outperforms BiLSTM because BiGRU converges faster, has simpler computation, and fewer parameters, reducing model training time while improving accuracy and reducing loss rate. The attention layer outperforms convolutional pooling layers in highlighting key information. Therefore, the BiGRU-Attention model demonstrates excellent performance across experimental metrics.

3.5.2 Iteration Count

This experiment uses 10 iterations as the basis for evaluation metric analysis. Accuracy, loss rate, and time cost are not static during these 10 iterations but change dynamically. To deeply reflect this dynamic change, this paper selects four most typical models that are most similar to and most recently proposed (BiGRU-Attention, ASC, BiLSTM-Attentions, and DBLSTM-Attention) for comparison.

a) This paper trains the BiGRU-Attention model, ASC model, BiLSTM-Attentions model, and DBLSTM-Attention model on the training set and conducts comparative experiments on the test set to obtain the relationship between accuracy and iteration count on the test set, as shown in Figure 7 [Figure 7: see original paper].

Figure 7 shows the accuracy changes of the four models with iteration count. Viewed from bottom to top, the accuracy of each model continuously improves overall. The BiGRU-Attention model' s accuracy is consistently higher than the other three models. The BiGRU-Attention model' s accuracy is higher than the ASC model, highlighting that the attention layer can more quickly highlight important information and extract deep text features compared to max pooling, achieving rapid convergence and quickly improving accuracy—it reaches the highest accuracy by the third iteration, with initial training accuracy higher than the other three models, indicating good training effectiveness. The BiGRU-Attention model' s accuracy changes relatively smoothly overall, though it may show lower accuracy at certain iterations, but always remains higher than the other three models. Overall, the curves of BiGRU-Attention, ASC, and BiLSTM-Attentions are relatively close, while the DBLSTM curve shows relatively large fluctuations. This shows that the BiGRU model performs better and more stably in extracting deep text features. From the iteration count perspective, more iterations do not necessarily mean higher accuracy—each model has an optimal iteration count at which it achieves highest accuracy. For example, BiGRU-Attention reaches highest accuracy at the fourth iteration, while ASC reaches it at the sixth iteration. This analysis shows that the BiGRU-Attention model can effectively improve training data accuracy with the fewest iterations.

b) The time required to complete one iteration is the time cost of the experiment. Figure 8 [Figure 8: see original paper] shows the trend curves of the time required for one iteration under the same experimental conditions for the four models.

As shown in Figure 8, the iteration times of each model generally fluctuate little, with overall time remaining stable. Typically, after the minimum iteration time is reached, training time will not have major fluctuations in subsequent training. The ASC model has the shortest single iteration training time, which is inseparable from the convergence speed of convolutional max pooling layers. The BiGRU-Attention model' s iteration time curve lies between

the DBLSTM-Attention iteration time curve and the ASC curve, with overall time performance in the middle. The difference between BiGRU-Attention and DBLSTM-Attention is basically caused by the difference between BiLSTM and BiGRU. Their different iteration times indicate different iteration speeds, showing that BiGRU has faster computation and fewer parameters than BiLSTM. The BiLSTM-Attentions model' s iteration time curve is relatively the highest because BiLSTM neural networks are relatively computationally complex, increasing computation time, and the attention layer also increases weighted computation time while highlighting key information.

c) The loss rate on the test dataset is also an important standard for measuring model performance—the smaller the loss rate, the better the model performance. As shown in Figure 9 [Figure 9: see original paper], some models show relatively large loss rate fluctuations during iterations, such as BiGRU-Attention, DBLSTM-Attention, and BiLSTM-Attentions. However, after the ASC model reaches its minimum loss rate, the curve changes little, indicating better stability for models without attention layers. Different models reach their minimum loss rates at different iteration counts. For example, BiGRU-Attention reaches its minimum loss rate of 0.2430 at the fourth iteration, while ASC reaches its minimum loss rate of 0.2445 at the sixth iteration. The BiGRU-Attention model's loss rate is significantly lower than other models in initial iterations. After reaching the minimum loss rate, its loss rate shows a gradual upward trend with increasing iteration count, while the ASC model' s curve changes relatively smoothly overall, showing a downward trend.

Combining Figures 7-9, although the BiGRU-Attention model requires slightly higher time cost than the ASC model, it achieves higher accuracy, lower loss rate, requires fewer iterations to reach optimal accuracy and loss rate, and generally needs less training time. The BiGRU-Attention model demonstrates incomparable advantages over DBLSTM-Attention and BiLSTM-Attentions models in terms of accuracy, loss rate, and iteration count on the training set. In summary, these experiments prove the effectiveness of the BiGRU-Attention model.

3.5.3 Comparison with BiLSTM-Attention and LSTM-Attention Models

Although comparative experiments between BiGRU-Attention and HDBN, S-LSTM, ASC, DBLSTM-Attention, and BiLSTM-Attentions models have sufficiently demonstrated that BiGRU-Attention reduces model training time while achieving high accuracy and low loss rate, these are not comparisons with the same type of hybrid experimental model, lacking experimental persuasiveness. Therefore, we selected BiGRU-Attention for comparison with BiLSTM-Attention and LSTM-Attention models, as detailed in Table 4 .

Table 4: Comparison of Three Models

Model	Accuracy	Loss Rate	Time Cost
BiGRU-Attention	90.54%	0.2430	1100s
BiLSTM-Attention	89.23%	0.2543	1352s
LSTM-Attention	87.63%	0.2612	1221s

Through comparison of the three models' accuracy, loss rate, and iteration time, we can see that the BiGRU-Attention model performs better on the training and test sets than the other two experimental models.

4 Conclusion

This paper proposes a novel neural network model based on BiGRU-Attention. Compared with the currently most widely used hybrid models of BiLSTM neural networks and attention, it can improve accuracy while reducing loss rate and appropriately decreasing model training time. On one hand, this demonstrates that BiGRU is simpler than BiLSTM and trains faster; on the other hand, it demonstrates the effectiveness of combining BiGRU with attention models. Compared with recently proposed models, the BiGRU-Attention neural network model is generally superior to other network models. Although the BiGRU-Attention neural network model achieves high accuracy and low loss rate on the IMDB dataset, its accuracy will decrease as data volume increases. Models incorporating attention mechanisms require further automatic weighting and processing of all objects and must store corresponding weight information, increasing system computation and overhead.

Future work will seek neural network models that can achieve high accuracy, low loss rate, small computation, and low system overhead with short training times when data volume is massive, making them more suitable for text sentiment classification.

References

[1] Lecun Yann, Bottou Lenon, Bengio Yoshua, et al. Gradient-based learning applied to document recognition [J]. Proceedings of the IEEE, 1998, 86 (11): 2278-2324.

[2] Chen Ke, Liang Bin, Ke Wende, et al. Chinese micro-blog emotional analysis based on multi-channel convolutional neural network [J]. Computer Research and Development, 2018, 55 (5): 945-957.

[3] Mikolov T, Sutskever I, Chen Kai, et al. Distributed representations of words and phrases and their compositionality [J]. Neural Information Processing Systems, 2013: 3111-3119.

[4] Liang Jun, Chai Yumei, Yuan Huibin, et al. Micro-blog emotional analysis based on deep learning [J]. Chinese Journal of Information, 2014, 28 (5): 155-161.

[5] Bai Jing, Li Fei, Ji Donghong. Attention based BiLSTM-CNN Chinese microblog stance detection model [J]. Computer Application and Software, 2018, 35 (3): 266-274.

[6] Liang Bin, Liu Quan, Xu Jin, et al. Specific target emotion analysis based on multi-attentions convolution neural network [J]. Computer Research and Development, 2017, 54 (8): 1724-1735.

[7] Zhao Qinlu, Cai Xiaodong, Li Bo, et al. Text feature extraction method based on LSTM-Attention neural network [J]. Modern Electronic Technology, 2018, 41 (8): 167-170.

[8] Yang Dong, Wang Yizhi. Text classification based on Attention-based C-GRU neural network [J]. Computer and Modernization, 2018 (2): 96-100.

[9] Si Nianwen, Wang Hengjun, Li Wei, et al. Chinese POS tagging model based on attention time memory network [J]. Computer Science, 2018, 45 (4): 66-70+82.

[10] Wang Yepei, Song Meijiao, Wang Pu, et al. Analysis of decision results propensity based on deep learning [J/OL]. Computer Application Research, 2019, 36 (2). [2018-06-23]. http://kns.cnki.net/kcms/detail/51.1196.TP.20180208.1753.154.html.

[11] Zhu Xingjia, Li Honglian, Lu Xueqiang, et al. An improved Attention-Based LSTM feature selection model [J]. Journal of Beijing University of Information Technology: Natural Science Edition, 2018, 33 (2): 54-59.

[12] Li Yanghui, Xie Ming, Yi Yang. Fine-grained Emotional Analysis of Social Networking Platform Based on Deep Learning [J]. Computer Applications, 2017, 34 (3): 743-747.

[13] Liu Yang. Time series prediction based on GRU neural network [D]. Chengdu: Chengdu University of Technology, 2017.

[14] Li Xiao, Huang Zheng. Internet Information Mining Based on GRU network [J]. Information Technology, 2018 (3): 1-5, 9.

[15] Huang Zhaowei, Chang Liang, Bin Chenzhong, et al. Remote supervisory relationship extraction based on GRU and Attention mechanism [J/OL]. Computer Application Research, 2019, 36 (10). [2018-07-05]. http://kns.cnki.net/kcms/detail/51.1196.TP.20180619.1516.012.html.

[16] Zhang Yuhuan, Qian Jiang. Text sentiment analysis based on two kinds of LSTM structure [J]. Software, 2018, 39 (1): 116-120.

[17] Tian Zhu. Text sentiment polarity classifications based on depth feature extraction [D]. Jinan: Shandong University, 2017.

[18] Zhou Guobing, Wu Jianxin, Zhang Chenlin, et al. Minimal gated unit for recurrent neural networks [J]. International Journal of Automation and Computing, 2016, 13 (3): 226-234.

[19] Huang Lei, Du Changshun. Text classification based on recurrent neural network [J]. Journal of Beijing University of Chemical Technology: Natural Science Edition, 2017, 44 (1): 98-104.

[20] Mnih Volodymyr, Heess Nicolas, Graves Alex, et al. Recurrent models of visual attention [J]. Neural Information Processing Systems, 2014: 2204-2212.

[21] Bahdanau D, Cho K, Bengio Y. Neural machine translation by jointly learning to align and translate [C]// Proc of International Conference on Learning Representations. 2015: 1-15.

[22] Luong T, Pham H, Manning C D. Effective approaches to attention-based neural machine translation [J]. Empirical Methods in Natural Language Processing, 2015: 1412-1421.

[23] Yin Wenpeng, Schutze H, Xiang Bing, et al. Attention-based convolutional netural network for modeling sentence pairs [J]. Transactions of the Association for Computional Linguistics, 2016, 4: 259-272.

[24] Tian Shengwei, Hu Wei, Yu Long, et al. Bi-LSTM Uygur event sequence recognition combined with attention mechanism [J]. Journal of Southeast University: Natural Science Edition, 2018, 48 (3): 393-399.

[25] Cheng Lu. A bidirectional LSTM model based on attention mechanism for emotional classification of Chinese comments [J]. Software Engineering, 2017, 20 (11): 4-6, 3.

[26] Rozental Alon, Fleischer Daniel. Amobee at SemEval-2018 Task 1: GRU neural network with a CNN attention mechanism for sentiment classification [C]// North American Chapter of the Association for Computational Linguistics. 2018: 218-225.

[27] Chen Peng, Sun Zhongqian, Bing Lidong, et al. Recurrent Attention Network on Memory for Aspect Sentient Analysis [C]// Proc of Conference on Empirical Methods in Natural Language Processing. 2017: 452-461.

[28] Kumar A, Kawahara D, Kurohashi S, et al. Knowledge-enriched two-layered attention for sentiment analysis [C]// North American Chapter of the Association for Computational Linguistics. 2018: 253-258.

[29] Yan Yan, Yin Xucheng, Li Sujiang, et al. Hybrid deep belief network [J]. Computational Intelligence and Neuroscience, 2015 (5): 650527: 1-650527: 11.

[30] Zhu Xiaodan, Sobhani P, Guo Hongyu. Learning document semantic representation with long short-term memory over recursive structures [C]// Proc of the 32nd International Conference on International Conference on Machince Learning. 2015: 1604-1612.

Note: Figure translations are in progress. See original paper for figures.