# Cached Long Short-Term Memory Neural Networks for Document-Level Sentiment Classification

**Jiacheng Xu**[†] **Danlu Chen**[‡] **Xipeng Qiu**[*‡] **Xuanjing Huang**[‡]
Software School, Fudan University [†]
School of Computer Science, Fudan University [‡]
Shanghai Key Laboratory of Intelligent Information Processing, Fudan University[†‡]
825 Zhangheng Road, Shanghai, China[†‡]
{jcxu13,dlchen13,xpqiu,xjhuang}@fudan.edu.cn

## Abstract

Recently, neural networks have achieved great success on sentiment classification due to their ability to alleviate feature engineering. However, one of the remaining challenges is to model long texts in document-level sentiment classification under a recurrent architecture because of the deficiency of the memory unit. To address this problem, we present a Cached Long Short-Term Memory neural networks (CLSTM) to capture the overall semantic information in long texts. CLSTM introduces a cache mechanism, which divides memory into several groups with different forgetting rates and thus enables the network to keep sentiment information better within a recurrent unit. The proposed CLSTM outperforms the state-of-the-art models on three publicly available document-level sentiment analysis datasets.

## 1 Introduction

Sentiment classification is one of the most widely used natural language processing techniques in many areas, such as E-commerce websites, online social networks, political orientation analyses (Wilson et al., 2009; O'Connor et al., 2010), etc.

Recently, deep learning approaches (Socher et al., 2013; Kim, 2014; Chen et al., 2015; Liu et al., 2016) have gained encouraging results on sentiment classification, which frees researchers from handcrafted feature engineering. Among these methods, Recurrent Neural Networks (RNNs) are one of the most prevalent architectures because of the ability to handle variable-length texts.

Sentence- or paragraph-level sentiment analysis expects the model to extract features from limited source of information, while document-level sentiment analysis demands more on selecting and storing global sentiment message from long texts with noises and redundant local pattern. Simple RNNs are not powerful enough to handle the overflow and to pick up key sentiment messages from relatively far time-steps .

Efforts have been made to solve such a scalability problem on long texts by extracting semantic information hierarchically (Tang et al., 2015a; Tai et al., 2015), which first obtain sentence representations and then combine them to generate high-level document embeddings. However, some of these solutions either rely on explicit *a priori* structural assumptions or discard the order information within a sentence, which are vulnerable to sudden change or twists in texts especially a long-range one (McDonald et al., 2007; Mikolov et al., 2013). Recurrent models match people's intuition of reading word by word and are capable to model the intrinsic relations between sentences. By keeping the word order, RNNs could extract the sentence representation implicitly and meanwhile analyze the semantic meaning of a whole document without any explicit boundary.

Partially inspired by neural structure of human brain and computer system architecture, we present the Cached Long Short-Term Memory neural networks (CLSTM) to capture the long-range sentiment information. In the dual store memory model

* Corresponding author.



*Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1660–1669,
Austin, Texas, November 1-5, 2016. 

proposed by Atkinson and Shiffrin (1968), memories can reside in the short-term "buffer" for a limited time while they are simultaneously strengthening their associations in long-term memory. Accordingly, CLSTM equips a standard LSTM with a similar cache mechanism, whose internal memory is divided into several groups with different forgetting rates. A group with high forgetting rate plays a role as a cache in our model, bridging and transiting the information to groups with relatively lower forgetting rates. With different forgetting rates, CLSTM learns to capture, remember and forget semantics information through a very long distance.

Our main contributions are as follows:

- We introduce a cache mechanism to diversify the internal memory into several distinct groups with different memory cycles by squashing their forgetting rates. As a result, our model can capture the local and global emotional information, thereby better summarizing and analyzing sentiment on long texts in an RNN fashion.
- Benefiting from long-term memory unit with a low forgetting rate, we could keep the gradient stable in the long back-propagation process. Hence, our model could converge faster than a standard LSTM.
- Our model outperforms state-of-the-art methods by a large margin on three document-level datasets (Yelp 2013, Yelp 2014 and IMDB). It worth noticing that some of the previous methods have utilized extra user and product information.

## 2 Related Work

In this section, we briefly introduce related work in two areas: First, we discuss the existing document-level sentiment classification approaches; Second, we discuss some variants of LSTM which address the problem on storing the long-term information.

### 2.1 Document-level Sentiment Classification

Document-level sentiment classification is a sticky task in sentiment analysis (Pang and Lee, 2008), which is to infer the sentiment polarity or intensity of a whole document. The most challenging part is that not every part of the document is equally informative for inferring the sentiment of the whole document (Pang and Lee, 2004; Yessenalina et al., 2010). Various methods have been investigated and explored over years (Wilson et al., 2005; Pang and Lee, 2008; Pak and Paroubek, 2010; Yessenalina et al., 2010; Moraes et al., 2013). Most of these methods depend on traditional machine learning algorithms, and are in need of effective handcrafted features.

Recently, neural network based methods are prevalent due to their ability of learning discriminative features from data (Socher et al., 2013; Le and Mikolov, 2014; Tang et al., 2015a). Zhu et al. (2015) and Tai et al. (2015) integrate a tree-structured model into LSTM for better semantic composition; Bhatia et al. (2015) enhances document-level sentiment analysis by using extra discourse paring results. Most of these models work well on sentence-level or paragraph-level sentiment classification. When it comes to the document-level sentiment classification, a bottom-up hierarchical strategy is often adopted to alleviate the model complexity (Denil et al., 2014; Tang et al., 2015b; Li et al., 2015).

### 2.2 Memory Augmented Recurrent Models

Although it is widely accepted that LSTM has more long-lasting memory units than RNNs, it still suffers from "forgetting" information which is too far away from the current point (Le et al., 2015; Karpathy et al., 2015). Such a scalability problem of LSTMs is crucial to extend some previous sentence-level work to document-level sentiment analysis.

Various models have been proposed to increase the ability of LSTMs to store long-range information (Le et al., 2015; Salehinejad, 2016) and two kinds of approaches gain attraction. One is to augment LSTM with an external memory (Sukhbaatar et al., 2015; Monz, 2016), but they are of poor performance on time because of the huge external memory matrix. Unlike these methods, we fully exploit the potential of internal memory of LSTM by adjusting its forgetting rates.

The other one tries to use multiple time-scales to distinguish different states (El Hihi and Bengio, 1995; Koutnik et al., 2014; Liu et al., 2015). They partition the hidden states into several groups and each group is activated and updated at different frequencies (e.g. one group updates every 2 time-step,

the other updates every 4 time-step). In these methods, different memory groups are not fully interconnected, and the information is transmitted from faster groups to slower ones, or vice versa.

However, the memory of slower groups are not updated at every step, which may lead to sentiment information loss and semantic inconsistency. In our proposed CLSTM, we assign different forgetting rates to memory groups. This novel strategy enable each memory group to be updated at every time-step, and every bit of the long-term and short-term memories in previous time-step to be taken into account when updating.

## 3 Long Short-Term Memory Networks

Long short-term memory network (LSTM) (Hochreiter and Schmidhuber, 1997) is a typical recurrent neural network, which alleviates the problem of gradient diffusion and explosion. LSTM can capture the long dependencies in a sequence by introducing a memory unit and a gate mechanism which aims to decide how to utilize and update the information kept in memory cell.

Formally, the update of each LSTM component can be formalized as:

$$\mathbf{i}^{(t)} = \sigma(\mathbf{W}_i\mathbf{x}^{(t)} + \mathbf{U}_i\mathbf{h}^{(t-1)}), \quad (1)$$

$$\mathbf{f}^{(t)} = \sigma(\mathbf{W}_f\mathbf{x}^{(t)} + \mathbf{U}_f\mathbf{h}^{(t-1)}), \quad (2)$$

$$\mathbf{o}^{(t)} = \sigma(\mathbf{W}_o\mathbf{x}^{(t)} + \mathbf{U}_o\mathbf{h}^{(t-1)}), \quad (3)$$

$$\tilde{\mathbf{c}}^{(t)} = \tanh(\mathbf{W}_c\mathbf{x}^{(t)} + \mathbf{U}_c\mathbf{h}^{(t-1)}), \quad (4)$$

$$\mathbf{c}^{(t)} = \mathbf{f}^{(t)} \odot \mathbf{c}^{(t-1)} + \mathbf{i}^{(t)} \odot \tilde{\mathbf{c}}^{(t)}, \quad (5)$$

$$\mathbf{h}^{(t)} = \mathbf{o}^{(t)} \odot \tanh(\mathbf{c}^{(t)}), \quad (6)$$

where $\sigma$ is the logistic sigmoid function. Operator $\odot$ is the element-wise multiplication operation. $\mathbf{i}^{(t)}$, $\mathbf{f}^{(t)}$, $\mathbf{o}^{(t)}$ and $\mathbf{c}^{(t)}$ are the input gate, forget gate, output gate, and memory cell activation vector at time-step $t$ respectively, all of which have the same size as the hidden vector $\mathbf{h}^{(t)} \in \mathbb{R}^H$. $\mathbf{W}_i$, $\mathbf{W}_f$, $\mathbf{W}_o \in \mathbb{R}^{H\times d}$ and $\mathbf{U}_i$, $\mathbf{U}_f$, $\mathbf{U}_o \in \mathbb{R}^{H\times H}$ are trainable parameters. Here, $H$ and $d$ are the dimensionality of hidden layer and input respectively.

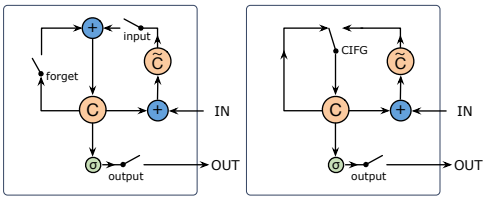



Figure 1: (a) A standard LSTM unit and (b) a CIFG-LSTM unit. There are three gates in (a), the input gate, forget gate and output gates, while in (b), there are only two gates, the CIFG gate and output gate.

## 4 Cached Long Short-Term Memory Neural Network

LSTM is supposed to capture the long-term and short-term dependencies simultaneously, but when dealing with considerably long texts, LSTM also fails on capturing and understanding significant sentiment message (Le et al., 2015). Specifically, the error signal would nevertheless suffer from gradient vanishing in modeling long texts with hundreds of words and thus the network is difficult to train.

Since the standard LSTM inevitably loses valuable features, we propose a cached long short-term memory neural networks (CLSTM) to capture information in a longer steps by introducing a cache mechanism. Moreover, in order to better control and balance the historical message and the incoming information, we adopt one particular variant of LSTM proposed by Greff et al. (2015), the Coupled Input and Forget Gate LSTM (CIFG-LSTM).

**Coupled Input and Forget Gate LSTM** Previous studies show that the merged version gives performance comparable to a standard LSTM on language modeling and classification tasks because using the input gate and forget gate simultaneously incurs redundant information (Chung et al., 2014; Greff et al., 2015).

In the CIFG-LSTM, the input gate and forget gate are coupled as one uniform gate, that is, let $\mathbf{i}^{(t)} = \mathbf{1} - \mathbf{f}^{(t)}$. We use $\mathbf{f}^{(t)}$ to denote the coupled gate. Formally, we will replace Eq. 5 as below:

$$\mathbf{c}^{(t)} = \mathbf{f}^{(t)} \odot \mathbf{c}^{(t-1)} + (\mathbf{1} - \mathbf{f}^{(t)}) \odot \tilde{\mathbf{c}}^{(t)} \quad (7)$$

Figure 1 gives an illustrative comparison of a standard LSTM and the CIFG-LSTM.

**Cached LSTM** Cached long short-term memory neural networks (CLSTM) aims at capturing the long-range information by a cache mechanism, which divides memory into several groups, and different forgetting rates, regarded as filters, are assigned to different groups.

Different groups capture different-scale dependencies by squashing the scales of forgetting rates. The groups with high forgetting rates are short-term memories, while the groups with low forgetting rates are long-term memories.

Specially, we divide the memory cells into $K$ groups $\{G_1, \cdots, G_K\}$. Each group includes a internal memory $\mathbf{c}_k$, output gate $\mathbf{o}_k$ and forgetting rate $\mathbf{r}_k$. The forgetting rate of different groups are squashed in distinct ranges.

We modify the update of a LSTM as follows.

$$\mathbf{r}_k^{(t)} = \psi_k \left( \sigma(\mathbf{W}_r^k \mathbf{x}^{(t)} + \sum_{j=1}^{K} \mathbf{U}_f^{j \to k} \mathbf{h}_j^{(t-1)}) \right), \tag{8}$$

$$\mathbf{o}_k^{(t)} = \sigma(\mathbf{W}_o^k \mathbf{x}^{(t)} + \sum_{j=1}^{K} \mathbf{U}_o^{j \to k} \mathbf{h}_j^{(t-1)}), \tag{9}$$

$$\tilde{\mathbf{c}}_k^{(t)} = \tanh(\mathbf{W}_c^k \mathbf{x}^{(t)} + \sum_{j=1}^{K} \mathbf{U}_c^{j \to k} \mathbf{h}_j^{(t-1)}), \tag{10}$$

$$\mathbf{c}_k^{(t)} = (1 - \mathbf{r}_k^{(t)}) \odot \mathbf{c}_k^{(t-1)} + (\mathbf{r}_k^{(t)}) \odot \tilde{\mathbf{c}}_k^{(t)}, \tag{11}$$

$$\mathbf{h}_k^{(t)} = \mathbf{o}_k^{(t)} \odot \tanh(\mathbf{c}_k^{(t)}), \tag{12}$$

where $\mathbf{r}_k^{(t)}$ represents forgetting rate of the $k$-th memory group at step $t$; $\psi_k$ is a squash function, which constrains the value of forgetting rate $\mathbf{r}_k$ within a range. To better distinguish the different role of each group, its forgetting rate is squashed into a distinct area. The squash function $\psi_k(\mathbf{z})$ could be formalized as:

$$\mathbf{r}_k = \psi_k(\mathbf{z}) = \frac{1}{K} \cdot \mathbf{z} + \frac{k-1}{K}, \tag{13}$$

where $\mathbf{z} \in (0, 1)$ is computed by logistic sigmoid function. Therefore, $\mathbf{r}_k$ can constrain the forgetting rate in the range of $(\frac{k-1}{K}, \frac{k}{K})$.

Intuitively, if a forgetting rate $\mathbf{r}_k$ approaches to $\mathbf{0}$, the group $k$ tends to be the long-term memory; if a $\mathbf{r}_k$ approaches to $\mathbf{1}$, the group $k$ tends to be the short-term memory. Therefore, group $G_1$ is the slowest, while group $G_K$ is the fastest one. The faster groups are supposed to play a role as a cache, transiting information from faster groups to slower groups.

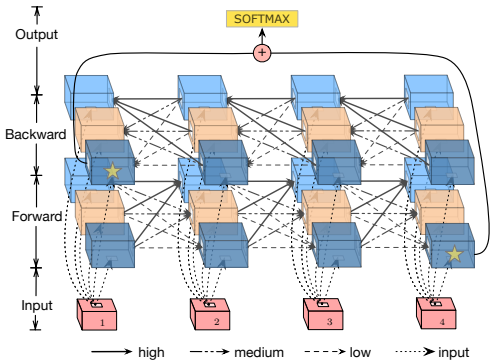



Figure 2: An overview of the proposed architecture. Different styles of arrows indicate different forgetting rates. Groups with stars are fed to a fully connected layers for softmax classification. Here is an instance of B-CLSTM with text length equal to 4 and the number of memory groups is 3.

**Bidirectional CLSTM** Graves and Schmidhuber (2005) proposed a Bidirectional LSTM (B-LSTM) model, which utilizes additional backward information and thus enhances the memory capability.

We also employ the bi-directional mechanism on CLSTM and words in a text will receive information from both sides of the context. Formally, the outputs of forward LSTM for the $k$-th group is $[\overrightarrow{\mathbf{h}}_k^{(1)}, \overrightarrow{\mathbf{h}}_k^{(2)}, \ldots, \overrightarrow{\mathbf{h}}_k^{(T)}]$. The outputs of backward LSTM for the $k$-th group is $[\overleftarrow{\mathbf{h}}_k^{(1)}, \overleftarrow{\mathbf{h}}_k^{(2)}, \ldots, \overleftarrow{\mathbf{h}}_k^{(T)}]$.

Hence, we encode each word $w_t$ in a given text $w_{1:T}$ as $\mathbf{h}_k^{(t)}$:

$$\mathbf{h}_k^{(t)} = \overrightarrow{\mathbf{h}}_k^{(t)} \oplus \overleftarrow{\mathbf{h}}_k^{(t)}, \tag{14}$$

where the $\oplus$ indicates concatenation operation.

**Task-specific Output Layer for Document-level Sentiment Classification** With the capability of modeling long text, we can use our proposed model to analyze sentiment in a document. Figure 2 gives an overview of the architecture.

Since the first group, the slowest group, is supposed to keep the long-term information and can better represent a whole document, we only utilize the

| Dataset | Type | Train Size | Dev. Size | Test Size | Class | Words/Doc | Sents/Doc |
|---|---|---|---|---|---|---|---|
| IMDB | Document | 67426 | 8381 | 9112 | 10 | 394.6 | 16.08 |
| Yelp 2013 | Document | 62522 | 7773 | 8671 | 5 | 189.3 | 10.89 |
| Yelp 2014 | Document | 183019 | 22745 | 25399 | 5 | 196.9 | 11.41 |

Table 1: Statistics of the three datasets used in this paper. The rating scale (Class) of Yelp2013 and Yelp2014 range from 1 to 5 and that of IMDB ranges from 1 to 10. Words/Doc is the average length of a sample and Sents/Doc is the average number of sentences in a document.

final state of this group to represent a document. As for the B-CLSTM, we concatenate the state of the first group in the forward LSTM at $T$-th time-step and the first group in the backward LSTM at first time-step.

Then, a fully connected layer followed by a softmax function is used to predict the probability distribution over classes for a given input. Formally, the probability distribution $\mathbf{p}$ is:

$$\mathbf{p} = \text{softmax}(\mathbf{W}_p \times \mathbf{z} + \mathbf{b}_p), \tag{15}$$

where $\mathbf{W}_p$ and $\mathbf{b}_p$ are model's parameters. Here $\mathbf{z}$ is $\overrightarrow{\mathbf{h}}_1^{(T)}$ in CLSTM, and $\mathbf{z}$ is $[\overrightarrow{\mathbf{h}}_1^{(T)} \oplus \overleftarrow{\mathbf{h}}_1^{(1)}]$ in B-CLSTM.

## 5 Training

The objective of our model is to minimize the cross-entropy error of the predicted and true distributions. Besides, the objective includes an $L_2$ regularization term over all parameters. Formally, suppose we have $m$ train sentence and label pairs $(w_{1:T_i}^{(i)}, y^{(i)})_{i=1}^m$, the object is to minimize the objective function $J(\theta)$:

$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m} \log \mathbf{p}_{y^{(i)}}^{(i)} + \frac{\lambda}{2}||\theta||^2, \tag{16}$$

where $\theta$ denote all the trainable parameters of our model.

## 6 Experiment

In this section, we study the empirical result of our model on three datasets for document-level sentiment classification. Results show that the proposed model outperforms competitor models from several aspects when modelling long texts.

### 6.1 Datasets

Most existing datasets for sentiment classification such as Stanford Sentiment Treebank (Socher et al., 2013) are composed of short paragraphs with several sentences, which cannot evaluate the effectiveness of the model under the circumstance of encoding long texts. We evaluate our model on three popular real-world datasets, Yelp 2013, Yelp 2014 and IMDB. Table 1 shows the statistical information of the three datasets. All these datasets can be publicly accessed[1]. We pre-process and split the datasets in the same way as Tang et al. (2015b) did.

- **Yelp 2013** and **Yelp 2014** are review datasets derived from Yelp Dataset Challenge[2] of year 2013 and 2014 respectively. The sentiment polarity of each review is 1 star to 5 stars, which reveals the consumers' attitude and opinion towards the restaurants.
- **IMDB** is a popular movie review dataset consists of 84919 movie reviews ranging from 1 to 10 (Diao et al., 2014). Average length of each review is 394.6 words, which is much larger than the length of two Yelp review datasets.

### 6.2 Evaluation Metrics

We use Accuracy (Acc.) and MSE as evaluation metrics for sentiment classification. Accuracy is a standard metric to measure the overall classification result and Mean Squared Error (MSE) is used to figure out the divergences between predicted sentiment labels and the ground truth ones.

### 6.3 Baseline Models

We compare our model, CLSTM and B-CLSTM with the following baseline methods.

- **CBOW** sums the word vectors and applies a non-linearity followed by a softmax classification layer.

[1] http://ir.hit.edu.cn/~dytang/paper/acl2015/dataset.7z

[2] http://www.yelp.com/dataset_challenge

| Model | IMDB | | Yelp 2014 | | Yelp 2013 | |
|---|---|---|---|---|---|---|
| | Acc. (%) | MSE | Acc. (%) | MSE | Acc. (%) | MSE |
| CBOW | 34.8 | 2.867 | 56.8 | 0.620 | 54.5 | 0.706 |
| PV (Tang et al., 2015b) | 34.1 | 3.291 | 56.4 | 0.643 | 55.4 | 0.692 |
| RNTN+Recurrent (Tang et al., 2015b) | 40.0 | 3.112 | 58.2 | 0.674 | 57.4 | 0.646 |
| UPNN (CNN) (Tang et al., 2015b) | 40.5 | 2.654 | 58.5 | 0.653 | 57.7 | 0.659 |
| JMARS* (Diao et al., 2014) | - | 3.143 | - | 0.998 | - | 0.970 |
| UPNN (CNN)* (Tang et al., 2015b) | **43.5** | **2.566** | **60.8** | **0.584** | **59.6** | **0.615** |
| RNN | 20.5 | 6.163 | 41.0 | 1.203 | 42.8 | 1.144 |
| LSTM | 37.8 | 2.597 | 56.3 | 0.592 | 53.9 | 0.656 |
| CIFG-LSTM | 39.1 | 2.467 | 55.2 | 0.598 | 57.3 | 0.558 |
| CLSTM | **42.1** | **2.399** | **59.2** | **0.539** | **59.4** | **0.587** |
| BLSTM | 43.3 | 2.231 | 59.2 | 0.538 | 58.4 | 0.583 |
| CIFG-BLSTM | 44.5 | 2.283 | 60.1 | 0.527 | 59.2 | 0.554 |
| B-CLSTM | **46.2** | **2.112** | **61.9** | **0.496** | **59.8** | **0.549** |

Table 2: Sentiment classification results of our model against competitor models on IMDB, Yelp 2014 and Yelp 2013. Evaluation metrics are classification accuracy (Acc.) and MSE. Models with * use user and product information as additional features. Best results in each group are in bold.

| Dataset | IMDB | Yelp13 | Yelp14 |
|---|---|---|---|
| Hidden layer units | 120 | 120 | 120 |
| Number of groups | 3 | 4 | 4 |
| Weight Decay | $1e{-4}$ | $1e{-4}$ | $5e{-4}$ |
| Batch size | 128 | 64 | 64 |

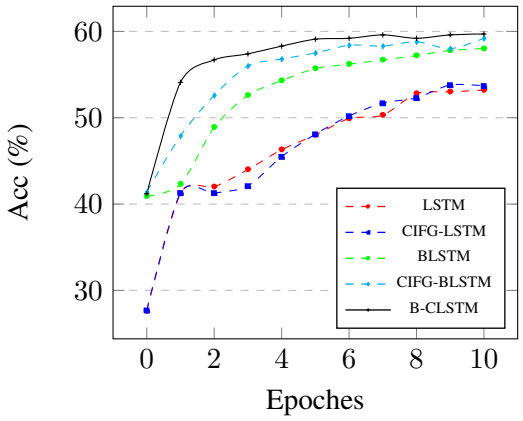
Table 3: Optimal hyper-parameter configuration for three datasets.

- **JMARS** is one of the state-of-the-art recommendation algorithm (Diao et al., 2014), which leverages user and aspects of a review with collaborative filtering and topic modeling.
- **CNN** UPNN (CNN) (Tang et al., 2015b) can be regarded as a CNN (Kim, 2014). Multiple filters are sensitive to capture different semantic features during generating a representation in a bottom-up fashion.
- **RNN** is a basic sequential model to model texts (Elman, 1991).
- **LSTM** is a recurrent neural network with memory cells and gating mechanism (Hochreiter and Schmidhuber, 1997).
- **BLSTM** is the bidirectional version of LSTM, and can capture more structural information and longer distance during looking forward and back (Graves et al., 2013).
- **CIFG-LSTM & CIFG-BLSTM** are Coupled Input Forget Gate LSTM and BLSTM, denoted as CIFG-LSTM and CIFG-BLSTM respectively (Greff et al., 2015). They combine the input and forget gate of LSTM and require smaller number of parameters in comparison with the standard LSTM.

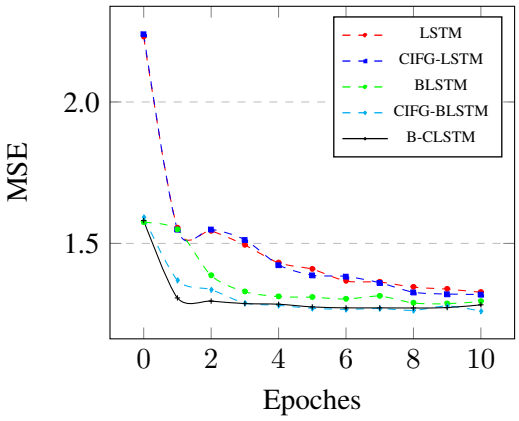
### 6.4 Hyper-parameters and Initialization

For parameter configuration, we choose parameters on validation set mainly according to classification accuracy for convenience because MSE always has strong correlation with accuracy. The dimension of pre-trained word vectors is 50. We use 120 as the dimension of hidden units, and choose weight decay among $\{ 5e{-4}, 1e{-4}, 1e{-5} \}$. We use Adagrad (Duchi et al., 2011) as optimizer and its initial learning rate is 0.01. Batch size is chosen among $\{$ 32, 64, 128 $\}$ for efficiency. For CLSTM, the number of memory groups is chosen upon each dataset, which will be discussed later. We remain the total number of the hidden units unchanged. Given 120 neurons in all for instance, there are four memory groups and each of them has 30 neurons. This makes model comparable to (B)LSTM. Table 3 shows the optimal hyper-parameter configurations for each dataset.

For model initialization, we initialize all recurrent matrices with randomly sampling from uniform distribution in [-0.1, 0.1]. Besides, we use GloVe(Pennington et al., 2014) as pre-trained word vectors. The word embeddings are fine-tuned during training. Hyper-parameters achieving best results on




(a) Accuracy on Yelp 2013




(b) MSE on Yelp 2013

Figure 3: Convergence speed experiment on Yelp 2013. X-axis is the iteration epoches and Y-axis is the classifcication accuracy(%) achieved.

the validation set are chosen for final evaluation on test set.

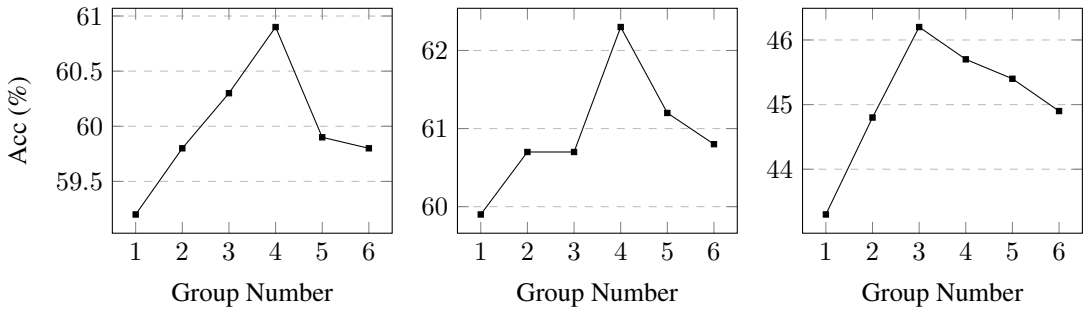
### 6.5 Results

The classification accuracy and mean square error (MSE) of our models compared with other competitive models are shown in Table 2. When comparing our models to other neural network models, we have several meaningful findings.

1. Among all unidirectional sequential models, RNN fails to capture and store semantic features while vanilla LSTM preserves sentimental messages much longer than RNN. It shows that internal memory plays a key role in text modeling. CIFG-LSTM gives performance comparable to vanilla LSTM.
2. With the help of bidirectional architecture, models could look backward and forward to capture features in long-range from global perspective. In sentiment analysis, if users show their opinion at the beginning of their review, single directional models will possibly forget these hints.
3. The proposed CLSTM beats the CIFG-LSTM and vanilla LSTM and even surpasses the bidirectional models. In Yelp 2013, CLSTM achieves 59.4% in accuracy, which is only 0.4 percent worse than B-CLSTM, which reveals that the cache mechanism has successfully and effectively stored valuable information without the support from bidirectional structure.
4. Compared with existing best methods, our model has achieved new state-of-the-art results by a large margin on all document-level datasets in terms of classification accuracy. Moreover, B-CLSTM even has surpassed JMARS and CNN (UPNN) methods which utilized extra user and product information.
5. In terms of time complexity and numbers of parameters, our model keeps almost the same as its counterpart models while models of hierarchically composition may require more computational resources and time.

### 6.6 Rate of Convergence

We compare the convergence rates of our models, including CIFG-LSTM, CIFG-BLSTM and B-CLSTM, and the baseline models (LSTM and BLSTM). We configure the hyper-parameter to make sure every competing model has approximately the same numbers of parameters, and various models have shown different convergence rates in Figure 3. In terms of convergence rate, B-CLSTM beats other competing models. The reason why B-CLSTM converges faster is that the splitting memory groups can be seen as a better initialization and constraints during the training process.




(a) Acc on Yelp 2013 (b) Acc on Yelp 2014 (c) Acc on IMDB

Figure 4: Classification accuracy on different number of memory group on three datasets. X-axis is the number of memory group(s).

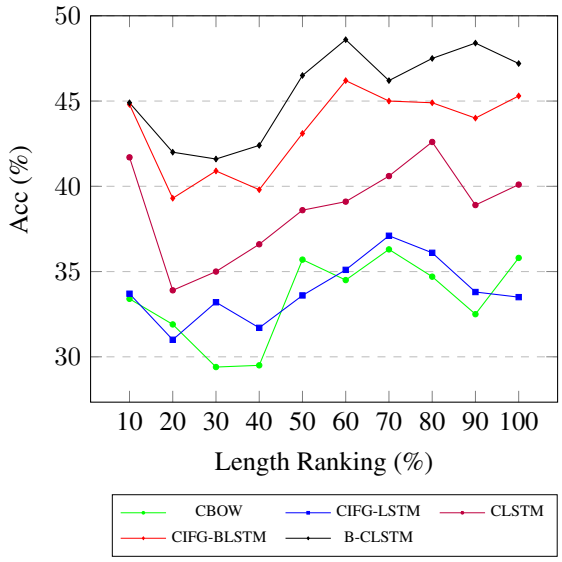



Figure 5: Study of model sensitivity on document length on IMDB. All test samples are sorted by their length and divided into 10 parts. Left most dot means classification accuracy on the shortest 10% samples. X-axis is length ranking from 0% to 100%.

### 6.7 Effectiveness on Grouping Memory

For the proposed model, the number of memory groups is a highlight. In Figure 4, we plot the best prediction accuracy (Y-axis) achieved in validation set with different number of memory groups on all datasets. From the diagram, we can find that our model outperforms the baseline method. In Yelp 2013, when we split the memory into 4 groups, it achieves the best result among all tested memory group numbers. We can observe the dropping trends when we choose more than 5 groups.

For fair comparisons, we set the total amount of neurons in our model to be same with vanilla LSTM. Therefore, the more groups we split, the less the neurons belongs to each group, which leads to a worse capacity than those who have sufficient neurons for each group.

### 6.8 Sensitivity on Document Length

We also investigate the performance of our model on IMDB when it encodes documents of different lengths. Test samples are divided into 10 groups with regard to the length. From Figure 5, we can draw several thoughtful conclusions.

1. Bidirectional models have much better performance than the counterpart models.
2. The overall performance of B-CLSTM is better than CIFG-BLSTM. This means that our model is adaptive to both short texts and long documents. Besides, our model shows power in dealing with very long texts in comparison with CIFG-BLSTM.
3. CBOW is slightly better than CIFG-LSTM due to LSTM forgets a large amount of information during the unidirectional propagation.

## 7 Conclusion

In this paper, we address the problem of effectively analyzing the sentiment of document-level texts in an RNN architecture. Similar to the memory structure of human, memory with low forgetting rate captures the global semantic features while memory with high forgetting rate captures the local semantic features. Empirical results on three real-world

document-level review datasets show that our model outperforms state-of-the-art models by a large margin.

For future work, we are going to design a strategy to dynamically adjust the forgetting rates for fine-grained document-level sentiment analysis.

## Acknowledgments

We appreciate the constructive work from Xinchi Chen. Besides, we would like to thank the anonymous reviewers for their valuable comments. This work was partially funded by National Natural Science Foundation of China (No. 61532011 and 61672162), the National High Technology Research and Development Program of China (No. 2015AA015408).

## References

Richard C Atkinson and Richard M Shiffrin. 1968. Human memory: A proposed system and its control processes. *The psychology of learning and motivation*, 2:89–195.

Parminder Bhatia, Yangfeng Ji, and Jacob Eisenstein. 2015. Better document-level sentiment analysis from rst discourse parsing. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing,(EMNLP)*.

Xinchi Chen, Xipeng Qiu, Chenxi Zhu, Shiyu Wu, and Xuanjing Huang. 2015. Sentence modeling with gated recursive neural network. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*.

Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. *NIPS Deep Learning Workshop*.

Misha Denil, Alban Demiraj, Nal Kalchbrenner, Phil Blunsom, and Nando de Freitas. 2014. Modelling, visualising and summarising documents with a single convolutional neural network. *arXiv preprint arXiv:1406.3830*.

Qiming Diao, Minghui Qiu, Chao-Yuan Wu, Alexander J. Smola, Jing Jiang, and Chong Wang. 2014. Jointly modeling aspects, ratings and sentiments for movie recommendation (JMARS). In *The 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '14, New York, NY, USA - August 24 - 27, 2014*, pages 193–202.

John C Duchi, Elad Hazan, and Yoram Singer. 2011. Adaptive Subgradient Methods for Online Learning and Stochastic Optimization. *Journal of Machine Learning Research*, 12:2121–2159.

Salah El Hihi and Yoshua Bengio. 1995. Hierarchical recurrent neural networks for long-term dependencies. In *NIPS*, pages 493–499.

Jeffrey L Elman. 1991. Distributed representations, simple recurrent networks, and grammatical structure. *Machine Learning*, 7(2-3):195–225.

Alex Graves and Jürgen Schmidhuber. 2005. Framewise phoneme classification with bidirectional lstm and other neural network architectures. *Neural Networks*, 18(5):602–610.

Alan Graves, Navdeep Jaitly, and Abdel-rahman Mohamed. 2013. Hybrid speech recognition with deep bidirectional lstm. In *Automatic Speech Recognition and Understanding (ASRU), 2013 IEEE Workshop on*, pages 273–278. IEEE.

Klaus Greff, Rupesh Kumar Srivastava, Jan Koutník, Bas R Steunebrink, and Jürgen Schmidhuber. 2015. LSTM: A Search Space Odyssey. *arXiv.org*, March.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Andrej Karpathy, Justin Johnson, and Fei-Fei Li. 2015. Visualizing and understanding recurrent networks. *International Conference on Learning Representations (ICLR), Workshop Track*.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 1746-1751*.

Jan Koutnik, Klaus Greff, Faustino Gomez, and Juergen Schmidhuber. 2014. A clockwork rnn. pages 1863–1871.

Quoc V Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In *ICML*, volume 14, pages 1188–1196.

Quoc V Le, Navdeep Jaitly, and Geoffrey E Hinton. 2015. A simple way to initialize recurrent networks of rectified linear units. *arXiv preprint arXiv:1504.00941*.

Jiwei Li, Thang Luong, Dan Jurafsky, and Eduard H. Hovy. 2015. When are tree structures necessary for deep learning of representations? In Llus Mrquez, Chris Callison-Burch, Jian Su, Daniele Pighin, and Yuval Marton, editors, *EMNLP*, pages 2304–2314. The Association for Computational Linguistics.

PengFei Liu, Xipeng Qiu, Xinchi Chen, Shiyu Wu, and Xuanjing Huang. 2015. Multi-timescale long short-term memory neural network for modelling sentences and documents. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*.

Pengfei Liu, Xipeng Qiu, and Xuanjing Huang. 2016. Recurrent neural network for text classification with multi-task learning. In *Proceedings of International Joint Conference on Artificial Intelligence*.

Ryan McDonald, Kerry Hannan, Tyler Neylon, Mike Wells, and Jeff Reynar. 2007. Structured models for fine-to-coarse sentiment analysis. In *Annual Meeting-Association For Computational Linguistics*, volume 45, page 432. Citeseer.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient Estimation of Word Representations in Vector Space. *arXiv.org*.

Ke Tran Arianna Bisazza Christof Monz. 2016. Recurrent memory networks for language modeling. In *Proceedings of NAACL-HLT*, pages 321–331.

Rodrigo Moraes, Joao Francisco Valiati, and Wilson P GaviãO Neto. 2013. Document-level sentiment classification: An empirical comparison between svm and ann. *Expert Systems with Applications*, 40(2):621–633.

Brendan O'Connor, Ramnath Balasubramanyan, Bryan R Routledge, and Noah A Smith. 2010. From Tweets to Polls: Linking Text Sentiment to Public Opinion Time Series. *ICWSM 2010*.

Alexander Pak and Patrick Paroubek. 2010. Twitter as a corpus for sentiment analysis and opinion mining. In *LREc*, volume 10, pages 1320–1326.

Bo Pang and Lillian Lee. 2004. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the 42Nd Annual Meeting on Association for Computational Linguistics*, ACL '04, Stroudsburg, PA, USA. Association for Computational Linguistics.

Bo Pang and Lillian Lee. 2008. Opinion mining and sentiment analysis. *Foundations and trends in information retrieval*, 2(1-2):1–135.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global Vectors for Word Representation. *EMNLP*, pages 1532–1543.

Hojjat Salehinejad. 2016. Learning over long time lags. *arXiv preprint arXiv:1602.04335*.

Richard Socher, Alex Perelygin, Jean Y Wu, Jason Chuang, Christopher D Manning, Andrew Y Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the conference on empirical methods in natural language processing (EMNLP)*, volume 1631, page 1642. Citeseer.

Sainbayar Sukhbaatar, Jason Weston, Rob Fergus, et al. 2015. End-to-end memory networks. In *Advances in Neural Information Processing Systems*, pages 2431–2439.

Kai Sheng Tai, Richard Socher, and Christopher D Manning. 2015. Improved Semantic Representations From Tree-Structured Long Short-Term Memory Networks. *ACL*, pages 1556–1566.

Duyu Tang, Bing Qin, and Ting Liu. 2015a. Document Modeling with Gated Recurrent Neural Network for Sentiment Classification. *EMNLP*, pages 1422–1432.

Duyu Tang, Bing Qin, and Ting Liu. 2015b. Learning Semantic Representations of Users and Products for Document Level Sentiment Classification. *ACL*, pages 1014–1023.

Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. 2005. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of the conference on human language technology and empirical methods in natural language processing*, pages 347–354. Association for Computational Linguistics.

Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. 2009. Recognizing contextual polarity: An exploration of features for phrase-level sentiment analysis. *Computational linguistics*, 35(3):399–433.

Ainur Yessenalina, Yisong Yue, and Claire Cardie. 2010. Multi-level structured models for document-level sentiment classification. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 1046–1056. Association for Computational Linguistics.

Xiaodan Zhu, Parinaz Sobhani, and Hongyu Guo. 2015. Long short-term memory over recursive structures. In *Proceedings of the 32nd International Conference on Machine Learning*, pages 1604–1612.