

# Aspect Based Sentiment Analysis with Gated Convolutional Networks

Wei Xue and Tao Li

School of Computing and Information Sciences  
Florida International University, Miami, FL, USA  
{wxue004, taoli}@cs.fiu.edu

## Abstract

Aspect based sentiment analysis (ABSA) can provide more detailed information than general sentiment analysis, because it aims to predict the sentiment polarities of the given aspects or entities in text. We summarize previous approaches into two subtasks: aspect-category sentiment analysis (ACSA) and aspect-term sentiment analysis (ATSA). Most previous approaches employ long short-term memory and attention mechanisms to predict the sentiment polarity of the concerned targets, which are often complicated and need more training time. We propose a model based on convolutional neural networks and gating mechanisms, which is more accurate and efficient. First, the novel Gated Tanh-ReLU Units can selectively output the sentiment features according to the given aspect or entity. The architecture is much simpler than attention layer used in the existing models. Second, the computations of our model could be easily parallelized during training, because convolutional layers do not have time dependency as in LSTM layers, and gating units also work independently. The experiments on SemEval datasets demonstrate the efficiency and effectiveness of our models.<sup>1</sup>

## 1 Introduction

Opinion mining and sentiment analysis (Pang and Lee, 2008) on user-generated reviews can provide valuable information for providers and consumers. Instead of predicting the overall sen-

timent polarity, fine-grained aspect based sentiment analysis (ABSA) (Liu and Zhang, 2012) is proposed to better understand reviews than traditional sentiment analysis. Specifically, we are interested in the sentiment polarity of aspect categories or target entities in the text. Sometimes, it is coupled with aspect term extractions (Xue et al., 2017). A number of models have been developed for ABSA, but there are two different subtasks, namely aspect-category sentiment analysis (ACSA) and aspect-term sentiment analysis (ATSA). The goal of ACSA is to predict the sentiment polarity with regard to the given aspect, which is one of a few predefined categories. On the other hand, the goal of ATSA is to identify the sentiment polarity concerning the target entities that appear in the text instead, which could be a multi-word phrase or a single word. The number of distinct words contributing to aspect terms could be more than a thousand. For example, in the sentence “Average to good Thai food, but terrible delivery.”, ATSA would ask the sentiment polarity towards the entity *Thai food*; while ACSA would ask the sentiment polarity toward the aspect *service*, even though the word *service* does not appear in the sentence.

Many existing models use LSTM layers (Hochreiter and Schmidhuber, 1997) to distill sentiment information from embedding vectors, and apply attention mechanisms (Bahdanau et al., 2014) to enforce models to focus on the text spans related to the given aspect/entity. Such models include Attention-based LSTM with Aspect Embedding (ATAE-LSTM) (Wang et al., 2016b) for ACSA; Target-Dependent Sentiment Classification (TD-LSTM) (Tang et al., 2016a), Gated Neural Networks (Zhang et al., 2016) and Recurrent Attention Memory Network (RAM) (Chen et al., 2017) for ATSA. Attention mechanisms has been successfully used in many

<sup>1</sup>The code and data is available at <https://github.com/wxue004cs/GCAE>

NLP tasks. It first computes the alignment scores between context vectors and target vector; then carry out a weighted sum with the scores and the context vectors. However, the context vectors have to encode both the aspect and sentiment information, and the alignment scores are applied across all feature dimensions regardless of the differences between these two types of information. Both LSTM and attention layer are very time-consuming during training. LSTM processes one token in a step. Attention layer involves exponential operation and normalization of all alignment scores of all the words in the sentence (Wang et al., 2016b). Moreover, some models needs the positional information between words and targets to produce weighted LSTM (Chen et al., 2017), which can be unreliable in noisy review text. Certainly, it is possible to achieve higher accuracy by building more and more complicated LSTM cells and sophisticated attention mechanisms; but one has to hold more parameters in memory, get more hyper-parameters to tune and spend more time in training. In this paper, we propose a fast and effective neural network for ACSA and ATSA based on convolutions and gating mechanisms, which has much less training time than LSTM based networks, but with better accuracy.

For ACSA task, our model has two separate convolutional layers on the top of the embedding layer, whose outputs are combined by novel gating units. Convolutional layers with multiple filters can efficiently extract n-gram features at many granularities on each receptive field. The proposed gating units have two nonlinear gates, each of which is connected to one convolutional layer. With the given aspect information, they can selectively extract aspect-specific sentiment information for sentiment prediction. For example, in the sentence “Average to good Thai food, but terrible delivery.”, when the aspect *food* is provided, the gating units automatically ignore the negative sentiment of aspect *delivery* from the second clause, and only output the positive sentiment from the first clause. Since each component of the proposed model could be easily parallelized, it has much less training time than the models based on LSTM and attention mechanisms. For ATSA task, where the aspect terms consist of multiple words, we extend our model to include another convolutional layer for the target expressions. We evaluate our models on the SemEval datasets, which contains

restaurants and laptops reviews with labels on aspect level. To the best of our knowledge, no CNN-based model has been proposed for aspect based sentiment analysis so far.

## 2 Related Work

We present the relevant studies into following two categories.

### 2.1 Neural Networks

Recently, neural networks have gained much popularity on sentiment analysis or sentence classification task. Tree-based recursive neural networks such as Recursive Neural Tensor Network (Socher et al., 2013) and Tree-LSTM (Tai et al., 2015), make use of syntactic interpretation of the sentence structure, but these methods suffer from time inefficiency and parsing errors on review text. Recurrent Neural Networks (RNNs) such as LSTM (Hochreiter and Schmidhuber, 1997) and GRU (Chung et al., 2014) have been used for sentiment analysis on data instances having variable length (Tang et al., 2015; Xu et al., 2016; Lai et al., 2015). There are also many models that use convolutional neural networks (CNNs) (Collobert et al., 2011; Kalchbrenner et al., 2014; Kim, 2014; Conneau et al., 2016) in NLP, which also prove that convolution operations can capture compositional structure of texts with rich semantic information without laborious feature engineering.

### 2.2 Aspect based Sentiment Analysis

There is abundant research work on aspect based sentiment analysis. Actually, the name ABSA is used to describe two different subtasks in the literature. We classify the existing work into two main categories based on the descriptions of sentiment analysis tasks in SemEval 2014 Task 4 (Pontiki et al., 2014): Aspect-Term Sentiment Analysis and Aspect-Category Sentiment Analysis.

**Aspect-Term Sentiment Analysis.** In the first category, sentiment analysis is performed toward the aspect terms that are labeled in the given sentence. A large body of literature tries to utilize the relation or position between the target words and the surrounding context words either by using the tree structure of dependency or by simply counting the number of words between them as a relevance information (Chen et al., 2017).

Recursive neural networks (Lakkaraju et al., 2014; Dong et al., 2014; Wang et al., 2016a) rely

on external syntactic parsers, which can be very inaccurate and slow on noisy texts like tweets and reviews, which may result in inferior performance.

Recurrent neural networks are commonly used in many NLP tasks as well as in ABSA problem. TD-LSTM (Tang et al., 2016a) and gated neural networks (Zhang et al., 2016) use two or three LSTM networks to model the left and right contexts of the given target individually. A fully-connected layer with gating units predicts the sentiment polarity with the outputs of LSTM layers.

Memory network (Weston et al., 2014) coupled with multiple-hop attention attempts to explicitly focus only on the most informative context area to infer the sentiment polarity towards the target word (Tang et al., 2016b; Chen et al., 2017). Nonetheless, memory network simply bases its knowledge bank on the embedding vectors of individual words (Tang et al., 2016b), which makes itself hard to learn the opinion word enclosed in more complicated contexts. The performance is improved by using LSTM, attention layer and feature engineering with word distance between surrounding words and target words to produce target-specific memory (Chen et al., 2017).

**Aspect-Category Sentiment Analysis.** In this category, the model is asked to predict the sentiment polarity toward a predefined aspect category. Attention-based LSTM with Aspect Embedding (Wang et al., 2016b) uses the embedding vectors of aspect words to selectively attend the regions of the representations generated by LSTMs.

### 3 Gated Convolutional Network with Aspect Embedding

In this section, we present a new model for ACSA and ATSA, namely Gated Convolutional network with Aspect Embedding (GCAE), which is more efficient and simpler than recurrent network based models (Wang et al., 2016b; Tang et al., 2016a; Ma et al., 2017; Chen et al., 2017). Recurrent neural networks sequentially compose hidden vectors  $\mathbf{h}_i = f(\mathbf{h}_{i-1})$ , which does not enable parallelization over inputs. In the attention layer, softmax normalization also has to wait for all the alignment scores computed by a similarity function. Hence, they cannot take advantage of highly-parallelized modern hardware and libraries. Our model is built on convolutional layers and gating units. Each convolutional filter computes n-gram features at different granularities from the embedding vectors

at each position individually. The gating units on top of the convolutional layers at each position are also independent from each other. Therefore, our model is more suitable to parallel computing. Moreover, our model is equipped with two kinds of effective filtering mechanisms: the gating units on top of the convolutional layers and the max pooling layer, both of which can accurately generate and select aspect-related sentiment features.

We first briefly review the vanilla CNN for text classification (Kim, 2014). The model achieves state-of-the-art performance on many standard sentiment classification datasets (Le et al., 2017).

The CNN model consists of an embedding layer, a one-dimension convolutional layer and a max-pooling layer. The embedding layer takes the indices  $w_i \in \{1, 2, \dots, V\}$  of the input words and outputs the corresponding embedding vectors  $\mathbf{v}_i \in \mathbb{R}^D$ .  $D$  denotes the dimension size of the embedding vectors.  $V$  is the size of the word vocabulary. The embedding layer is usually initialized with pre-trained embeddings such as GloVe (Pennington et al., 2014), then they are fine-tuned during the training stage. The one-dimension convolutional layer convolves the inputs with multiple convolutional kernels of different widths. Each kernel corresponds a linguistic feature detector which extracts a specific pattern of n-gram at various granularities (Kalchbrenner et al., 2014). Specifically, the input sentence is represented by a matrix through the embedding layer,  $\mathbf{X} = [\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_L]$ , where  $L$  is the length of the sentence with padding. A convolutional filter  $\mathbf{W}_c \in \mathbb{R}^{D \times k}$  maps  $k$  words in the receptive field to a single feature  $c$ . As we slide the filter across the whole sentence, we obtain a sequence of new features  $\mathbf{c} = [c_1, c_2, \dots, c_L]$ .

$$c_i = f(\mathbf{X}_{i:i+K} * \mathbf{W}_c + b_c) \quad , \quad (1)$$

where  $b_c \in \mathbb{R}$  is the bias,  $f$  is a non-linear activation function such as tanh function,  $*$  denotes convolution operation. If there are  $n_k$  filters of the same width  $k$ , the output features form a matrix  $\mathbf{C} \in \mathbb{R}^{n_k \times L_k}$ . For each convolutional filter, the max-over-time pooling layer takes the maximal value among the generated convolutional features, resulting in a fixed-size vector whose size is equal to the number of filters  $n_k$ . Finally, a softmax layer uses the vector to predict the sentiment polarity of the input sentence.

Figure 1 illustrates our model architecture. The

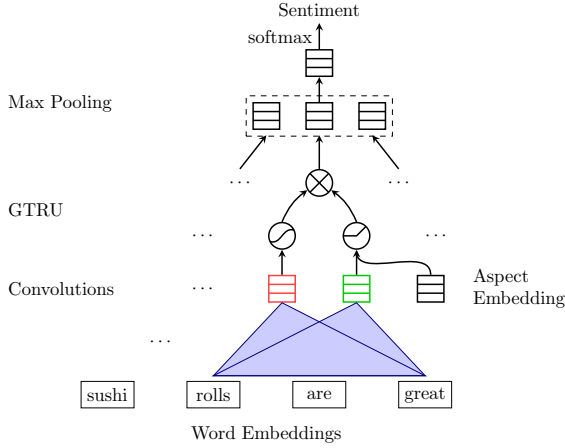


Figure 1: Illustration of our model GCAE for ACSA task. A pair of convolutional neuron computes features for a pair of gates: tanh gate and ReLU gate. The ReLU gate receives the given aspect information to control the propagation of sentiment features. The outputs of two gates are element-wisely multiplied for the max pooling layer.

Gated Tanh-ReLU Units (GTRU) with aspect embedding are connected to two convolutional neurons at each position  $t$ . Specifically, we compute the features  $c_i$  as

$$a_i = \text{relu}(\mathbf{X}_{i:i+k} * \mathbf{W}_a + \mathbf{V}_a v_a + b_a) \quad (2)$$

$$s_i = \text{tanh}(\mathbf{X}_{i:i+k} * \mathbf{W}_s + b_s) \quad (3)$$

$$c_i = s_i \times a_i, \quad (4)$$

where  $v_a$  is the embedding vector of the given aspect category in ACSA or computed by another CNN over aspect terms in ATSA. The two convolutions in Equation 2 and 3 are the same as the convolution in the vanilla CNN, but the convolutional features  $a_i$  receives additional aspect information  $v_a$  with ReLU activation function. In other words,  $s_i$  and  $a_i$  are responsible for generating sentiment features and aspect features respectively. The above max-over-time pooling layer generates a fixed-size vector  $e \in \mathbb{R}^{d_k}$ , which keeps the most salient sentiment features of the whole sentence. The final fully-connected layer with softmax function uses the vector  $e$  to predict the sentiment polarity  $\hat{y}$ . The model is trained by minimizing the cross-entropy loss between the ground-truth  $y$  and the predicted value  $\hat{y}$  for all data samples.

$$\mathcal{L} = - \sum_i \sum_j y_i^j \log \hat{y}_i^j, \quad (5)$$

where  $i$  is the index of a data sample,  $j$  is the index of a sentiment class.

## 4 Gating Mechanisms

The proposed Gated Tanh-ReLU Units control the path through which the sentiment information flows towards the pooling layer. The gating mechanisms have proven to be effective in LSTM. In aspect based sentiment analysis, it is very common that different aspects with different sentiments appear in one sentence. The ReLU gate in Equation 2 does not have upper bound on positive inputs but strictly zero on negative inputs. Therefore, it can output a similarity score according to the relevance between the given aspect information  $v_a$  and the aspect feature  $a_i$  at position  $t$ . If this score is zero, the sentiment features  $s_i$  would be blocked at the gate; otherwise, its magnitude would be amplified accordingly. The max-over-time pooling further removes the sentiment features which are not significant over the whole sentence.

In language modeling (Dauphin et al., 2017; Kalchbrenner et al., 2016; van den Oord et al., 2016; Gehring et al., 2017), Gated Tanh Units (GTU) and Gated Linear Units (GLU) have shown effectiveness of gating mechanisms. GTU is represented by  $\text{tanh}(\mathbf{X} * \mathbf{W} + b) \times \sigma(\mathbf{X} * \mathbf{V} + c)$ , in which the sigmoid gates control features for predicting the next word in a stacked convolutional block. To overcome the gradient vanishing problem of GTU, GLU uses  $(\mathbf{X} * \mathbf{W} + b) \times \sigma(\mathbf{X} * \mathbf{V} + c)$  instead, so that the gradients would not be down-scaled to propagate through many stacked convolutional layers. However, a neural network that has only one convolutional layer would not suffer from gradient vanish problem during training. We show that on text classification problem, our GTRU is more effective than these two gating units.

## 5 GCAE on ATSA

ATSA task is defined to predict the sentiment polarity of the aspect terms in the given sentence. We simply extend GCAE by adding a small convolutional layer on aspect terms, as shown in Figure 2. In ACSA, the aspect information controlling the flow of sentiment features in GTRU is from one aspect word; while in ATSA, such information is provided by a small CNN on aspect terms  $[w_i, w_{i+1}, \dots, w_{i+k}]$ . The additional CNN extracts the important features from multiple words

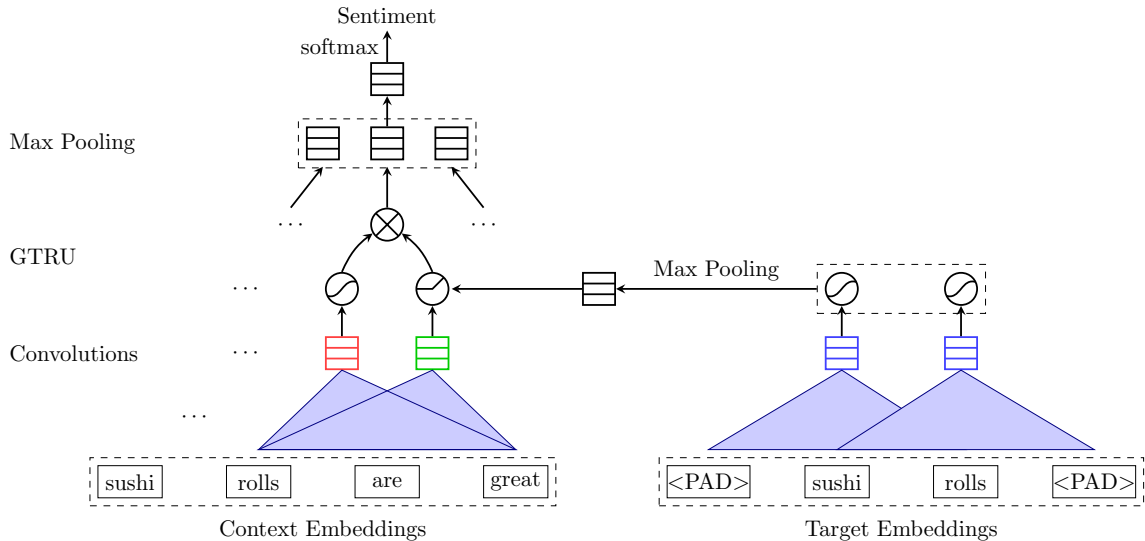


Figure 2: Illustration of model GCAE for ATSA task. It has an additional convolutional layer on aspect terms.

while retains the ability of parallel computing.

## 6 Experiments

### 6.1 Datasets and Experiment Preparation

We conduct experiments on public datasets from SemEval workshops (Pontiki et al., 2014), which consist of customer reviews about restaurants and laptops. Some existing work (Wang et al., 2016b; Ma et al., 2017; Chen et al., 2017) removed “conflict” labels from four sentiment labels, which makes their results incomparable to those from the workshop report (Kiritchenko et al., 2014). We reimplemented the compared methods, and used hyper-parameter settings described in these references.

The sentences which have different sentiment labels for different aspects or targets in the sentence are more common in review data than in standard sentiment classification benchmark. The sentence in Table 1 shows the reviewer’s different attitude towards two aspects: food and delivery. Therefore, to assess how the models perform on review sentences more accurately, we create small but difficult datasets, which are made up of the sentences having opposite or different sentiments on different aspects/targets. In Table 1, the two identical sentences but with different sentiment labels are both included in the dataset. If a sentence has 4 aspect targets, this sentence would have 4 copies in the data set, each of which is associated with different target and sentiment label.

For ACSA task, we conduct experiments on

restaurant review data of SemEval 2014 Task 4. There are 5 aspects: food, price, service, ambience, and misc; 4 sentiment polarities: positive, negative, neutral, and conflict. By merging restaurant reviews of three years 2014 - 2016, we obtain a larger dataset called “Restaurant-Large”. Incompatibilities of data are fixed during merging. We replace conflict labels with neutral labels in the 2014 dataset. In the 2015 and 2016 datasets, there could be multiple pairs of “aspect terms” and “aspect category” in one sentence. For each sentence, let  $p$  denote the number of positive labels minus the number of negative labels. We assign a sentence a positive label if  $p > 0$ , a negative label if  $p < 0$ , or a neutral label if  $p = 0$ . After removing duplicates, the statistics are shown in Table 2. The resulting dataset has 8 aspects: restaurant, food, drinks, ambience, service, price, misc and location.

For ATSA task, we use restaurant reviews and laptop reviews from SemEval 2014 Task 4. On each dataset, we duplicate each sentence  $n_a$  times, which is equal to the number of associated aspect categories (ACSA) or aspect terms (ATSA) (Ruder et al., 2016b,a). The statistics of the datasets are shown in Table 2.

The sizes of hard data sets are also shown in Table 2. The test set is designed to measure whether a model can detect multiple different sentiment polarities in one sentence toward different entities. Without such sentences, a classifier for overall sentiment classification might be good enough

Sentence	aspect category/term	sentiment label
Average to good Thai food, but terrible delivery.	food	positive
Average to good Thai food, but terrible delivery.	delivery	negative

Table 1: Two example sentences in one hard test set of restaurant review dataset of SemEval 2014.

for the sentences associated with only one sentiment label.

In our experiments, word embedding vectors are initialized with 300-dimension GloVe vectors which are pre-trained on unlabeled data of 840 billion tokens (Pennington et al., 2014). Words out of the vocabulary of GloVe are randomly initialized with a uniform distribution  $U(-0.25, 0.25)$ . We use Adagrad (Duchi et al., 2011) with a batch size of 32 instances, default learning rate of  $1e-2$ , and maximal epochs of 30. We only fine tune early stopping with 5-fold cross validation on training datasets. All neural models are implemented in PyTorch.

## 6.2 Compared Methods

To comprehensively evaluate the performance of GCAE, we compare our model against the following models.

**NRC-Canada** (Kiritchenko et al., 2014) is the top method in SemEval 2014 Task 4 for ACSA and ATSA task. SVM is trained with extensive feature engineering: various types of n-grams, POS tags, and lexicon features. The sentiment lexicons improve the performance significantly, but it requires large scale labeled data: 183 thousand Yelp reviews, 124 thousand Amazon laptop reviews, 56 million tweets, and 3 sentiment lexicons labeled manually.

**CNN** (Kim, 2014) is widely used on text classification task. It cannot directly capture aspect-specific sentiment information on ACSA task, but it provides a very strong baseline for sentiment classification. We set the widths of filters to 3, 4, 5 with 100 features each.

**TD-LSTM** (Tang et al., 2016a) uses two LSTM networks to model the preceding and following contexts of the target to generate target-dependent representation for sentiment prediction.

**ATAE-LSTM** (Wang et al., 2016b) is an attention-based LSTM for ACSA task. It appends the given aspect embedding with each word embedding as the input of LSTM, and has an attention layer above the LSTM layer.

**IAN** (Ma et al., 2017) stands for interactive

attention network for ATSA task, which is also based on LSTM and attention mechanisms.

**RAM** (Chen et al., 2017) is a recurrent attention network for ATSA task, which uses LSTM and multiple attention mechanisms.

**GCN** stands for gated convolutional neural network, in which GTRU does not have the aspect embedding as an additional input.

## 6.3 Results and Analysis

### 6.3.1 ACSA

Following the SemEval workshop, we report the overall accuracy of all competing models over the test datasets of restaurant reviews as well as the hard test datasets. Every experiment is repeated five times. The mean and the standard deviation are reported in Table 4.

LSTM based model ATAE-LSTM has the worst performance of all neural networks. Aspect-based sentiment analysis is to extract the sentiment information closely related to the given aspect. It is important to separate aspect information and sentiment information from the extracted information of sentences. The context vectors generated by LSTM have to convey the two kinds of information at the same time. Moreover, the attention scores generated by the similarity scoring function are for the entire context vector.

GCAE improves the performance by 1.1% to 2.5% compared with ATAE-LSTM. First, our model incorporates GTRU to control the sentiment information flow according to the given aspect information at each dimension of the context vectors. The element-wise gating mechanism works at fine granularity instead of exerting an alignment score to all the dimensions of the context vectors in the attention layer of other models. Second, GCAE does not generate a single context vector, but two vectors for aspect and sentiment features respectively, so that aspect and sentiment information is unraveled. By comparing the performance on the hard test datasets against CNN, it is easy to see the convolutional layer of GCAE is able to differentiate the sentiments of multiple entities.

Convolutional neural networks CNN and GCN

	Positive		Negative		Neutral		Conflict	
	Train	Test	Train	Test	Train	Test	Train	Test
Restaurant-Large	2710	1505	1198	680	757	241	-	-
Restaurant-Large-Hard	182	92	178	81	107	61	-	-
Restaurant-2014	2179	657	839	222	500	94	195	52
Restaurant-2014-Hard	139	32	136	26	50	12	40	19

Table 2: Statistics of the datasets for ACSA task. The hard dataset is only made up of sentences having multiple aspect labels associated with multiple sentiments.

	Positive		Negative		Neutral		Conflict	
	Train	Test	Train	Test	Train	Test	Train	Test
Restaurant	2164	728	805	196	633	196	91	14
Restaurant-Hard	379	92	323	62	293	83	43	8
Laptop	987	341	866	128	460	169	45	16
Laptop-Hard	159	31	147	25	173	49	17	3

Table 3: Statistics of the datasets for ATSA task.

are not designed for aspect based sentiment analysis, but their performance exceeds that of ATAE-LSTM.

The performance of SVM (Kiritchenko et al., 2014) depends on the availability of the features it can use. Without the large amount of sentiment lexicons, SVM perform worse than neural methods. With multiple sentiment lexicons, the performance is increased by 7.6%. This inspires us to work on leveraging sentiment lexicons in neural networks in the future.

The hard test datasets consist of replicated sentences with different sentiments towards different aspects. The models which cannot utilize the given aspect information such as CNN and GCN perform poorly as expected, but GCAE has higher accuracy than other neural network models. GCAE achieves 4% higher accuracy than ATAE-LSTM on Restaurant-Large and 5% higher on SemEval-2014 on ACSA task. However, GCN, which does not have aspect modeling part, has higher score than GCAE on the original restaurant dataset. It suggests that GCN performs better than GCAE when there is only one sentiment label in the given sentence, but not on the hard test dataset.

### 6.3.2 ATSA

We apply the extended version of GCAE on ATSA task. On this task, the aspect terms are marked in the sentences and usually consist of multiple words. We compare IAN (Ma et al., 2017), RAM (Chen et al., 2017), TD-LSTM (Tang et al., 2016a), ATAE-LSTM (Wang et al., 2016b), and

our GCAE model in Table 5. The models other than GCAE is based on LSTM and attention mechanisms. IAN has better performance than TD-LSTM and ATAE-LSTM, because two attention layers guides the representation learning of the context and the entity interactively. RAM also achieves good accuracy by combining multiple attentions with a recurrent neural network, but it needs more training time as shown in the following section. On the hard test dataset, GCAE has 1% higher accuracy than RAM on restaurant data and 1.7% higher on laptop data. GCAE uses the outputs of the small CNN over aspect terms to guide the composition of the sentiment features through the ReLU gate. Because of the gating mechanisms and the convolutional layer over aspect terms, GCAE outperforms other neural models and basic SVM. Again, large scale sentiment lexicons bring significant improvement to SVM.

### 6.4 Training Time

We record the training time of all models until convergence on a validation set on a desktop machine with a 1080 Ti GPU, as shown in Table 6. LSTM based models take more training time than convolutional models. On ATSA task, because of multiple attention layers in IAN and RAM, they need even more time to finish the training. GCAE is much faster than other neural models, because neither convolutional operation nor GTRU has time dependency compared with LSTM and attention layer. Therefore, it is easier for hardware and libraries to parallel the comput-

Models	Restaurant-Large		Restaurant 2014	
	Test	Hard Test	Test	Hard Test
SVM*	-	-	75.32	-
SVM + lexicons*	-	-	<b>82.93</b>	-
ATAE-LSTM	83.91±0.49	66.32±2.28	78.29±0.68	45.62±0.90
CNN	84.28±0.15	50.43±0.38	79.47±0.32	44.94±0.01
GCN	84.48±0.06	50.08±0.31	<b>79.67±0.35</b>	44.49±1.52
GCAE	<b>85.92±0.27</b>	<b>70.75±1.19</b>	79.35±0.34	<b>50.55±1.83</b>

Table 4: The accuracy of all models on test sets and on the subsets made up of test sentences that have multiple sentiments and multiple aspect terms. Restaurant-Large dataset is created by merging all the restaurant reviews of SemEval workshops within three years. ‘\*’: the results with SVM are retrieved from NRC-Canada (Kiritchenko et al., 2014).

Models	Restaurant		Laptop	
	Test	Hard Test	Test	Hard Test
SVM*	77.13	-	63.61	-
SVM + lexicons*	<b>80.16</b>	-	<b>70.49</b>	-
TD-LSTM	73.44±1.17	56.48±2.46	62.23±0.92	46.11±1.89
ATAE-LSTM	73.74±3.01	50.98±2.27	64.38±4.52	40.39±1.30
IAN	76.34±0.27	55.16±1.97	68.49±0.57	44.51±0.48
RAM	76.97±0.64	55.85±1.60	68.48±0.85	45.37±2.03
GCAE	<b>77.28±0.32</b>	<b>56.73±0.56</b>	<b>69.14±0.32</b>	<b>47.06±2.45</b>

Table 5: The accuracy of ATSA subtask on SemEval 2014 Task 4. ‘\*’: the results with SVM are retrieved from NRC-Canada (Kiritchenko et al., 2014)

Model	ATSA
ATAE	25.28
IAN	82.87
RAM	64.16
TD-LSTM	19.39
GCAE	3.33

Table 6: The time to converge in seconds on ATSA task.

Gates	Restaurant-Large		Restaurant 2014	
	Test	Hard Test	Test	Hard Test
GTU	84.62	60.25	79.31	<b>51.93</b>
GLU	84.74	59.82	79.12	50.80
GTRU	<b>85.92</b>	<b>70.75</b>	<b>79.35</b>	50.55

Table 7: The accuracy of different gating units on restaurant reviews on ACSA task.

ing process. Since the performance of SVM is retrieved from the original paper, we are not able to compare the training time of SVM.

### 6.5 Gating Mechanisms

In this section, we compare GLU ( $\mathbf{X} * \mathbf{W} + b$ )  $\times$   $\sigma(\mathbf{X} * \mathbf{W}_a + \mathbf{V}v_a + b_a)$  (Dauphin et al., 2017),

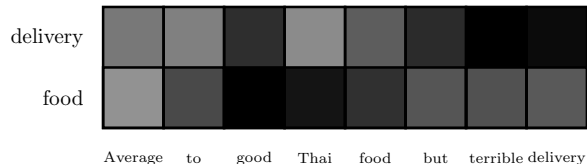


Figure 3: The outputs of the ReLU gates in GTRU.

GTU  $\tanh(\mathbf{X} * \mathbf{W} + b) \times \sigma(\mathbf{X} * \mathbf{W}_a + \mathbf{V}v_a + b_a)$  (van den Oord et al., 2016), and GTRU used in GCAE. Table 7 shows that all of three gating units achieve relatively high accuracy on restaurant datasets. GTRU outperforms the other gates. It has a convolutional layer generating aspect features via ReLU activation function, which controls the magnitude of the sentiment signals according to the given aspect information. On the other hand, the sigmoid function in GTU and GLU has the upper bound +1, which may not be able to distill sentiment features effectively.

## 7 Visualization

In this section, we take a concrete review sentence as an example to illustrate how the proposed gate GTRU works. It is more difficult to visualize



the weights generated by the gates than the attention weights in other neural networks. The attention weight score is a global score over the words and the vector dimensions; whereas in our model, there are  $N_{\text{word}} \times N_{\text{filter}} \times N_{\text{dimension}}$  gate outputs. Therefore, we train a small model with only one filter which is only three word wide. Then, for each word, we sum the  $N_{\text{dimension}}$  outputs of the ReLU gates. After normalization, we plot the values on each word in Figure 3. Given different aspect targets, the ReLU gates would control the magnitude of the outputs of the tanh gates.

## 8 Conclusions and Future Work

In this paper, we proposed an efficient convolutional neural network with gating mechanisms for ACSA and ATSA tasks. GTRU can effectively control the sentiment flow according to the given aspect information, and two convolutional layers model the aspect and sentiment information separately. We prove the performance improvement compared with other neural models by extensive experiments on SemEval datasets. How to leverage large-scale sentiment lexicons in neural networks would be our future work.

## References

- Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural Machine Translation by Jointly Learning to Align and Translate. In *ICLR*, pages CoRR abs-1409.0473.
- Peng Chen, Zhongqian Sun, Lidong Bing, and Wei Yang. 2017. Recurrent Attention Network on Memory for Aspect Sentiment Analysis. In *EMNLP*, pages 463–472.
- Junyoung Chung, Caglar Gulcehre, Kyunghyun Cho, and Yoshua Bengio. 2014. Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling. In *NIPS*.
- Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel P. Kuksa. 2011. Natural Language Processing (Almost) from Scratch. *Journal of Machine Learning Research*, 12:2493–2537.
- Alexis Conneau, Holger Schwenk, Loïc Barrault, and Yann LeCun. 2016. Very Deep Convolutional Networks for Text Classification. In *EACL*, pages 1107–1116.
- Yann N Dauphin, Angela Fan, Michael Auli, and David Grangier. 2017. Language Modeling with Gated Convolutional Networks. In *ICML*, pages 933–941.
- Li Dong, Furu Wei, Chuanqi Tan, Duyu Tang, Ming Zhou, and Ke Xu. 2014. Adaptive Recursive Neural Network for Target-dependent Twitter Sentiment Classification. In *ACL*, pages 49–54.
- John C Duchi, Elad Hazan, and Yoram Singer. 2011. Adaptive Subgradient Methods for Online Learning and Stochastic Optimization. *Journal of Machine Learning Research*, pages 2121–2159.
- Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N Dauphin. 2017. Convolutional Sequence to Sequence Learning. In *ICML*, pages 1243–1252.
- Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long Short-Term Memory. *Neural computation*, 9(8):1735–1780.
- Nal Kalchbrenner, Lasse Espeholt, Karen Simonyan, Aaron van den Oord, Alex Graves, and Koray Kavukcuoglu. 2016. Neural Machine Translation in Linear Time. *CoRR*, abs/1610.10099.
- Nal Kalchbrenner, Edward Grefenstette, and Phil Blunsom. 2014. A convolutional neural network for modelling sentences. In *ACL*, pages 655–665.
- Yoon Kim. 2014. Convolutional Neural Networks for Sentence Classification. In *EMNLP*, pages 1746–1751.
- Svetlana Kiritchenko, Xiaodan Zhu, Colin Cherry, and Saif M. Mohammad. 2014. NRC-Canada-2014: Detecting aspects and sentiment in customer reviews. In *SemEval@COLING*, pages 437–442, Stroudsburg, PA, USA. Association for Computational Linguistics.
- Siwei Lai, Liheng Xu, Kang Liu, and Jun Zhao. 2015. Recurrent Convolutional Neural Networks for Text Classification. *AAAI*, pages 2267–2273.
- Himabindu Lakkaraju, Richard Socher, and Chris Manning. 2014. Aspect specific sentiment analysis using hierarchical deep learning. In *NIPS Workshop on Deep Learning and Representation Learning*.
- Hoa T Le, Christophe Cerisara, and Alexandre Denis. 2017. Do Convolutional Networks need to be Deep for Text Classification? *CoRR*, abs/1707.04108.
- Bing Liu and Lei Zhang. 2012. A Survey of Opinion Mining and Sentiment Analysis. *Mining Text Data*, (Chapter 13):415–463.
- Dehong Ma, Sujian Li, Xiaodong Zhang, and Houfeng Wang. 2017. Interactive Attention Networks for Aspect-Level Sentiment Classification. In *IJCAI*, pages 4068–4074. International Joint Conferences on Artificial Intelligence Organization.
- Aaron van den Oord, Nal Kalchbrenner, Lasse Espeholt, Koray Kavukcuoglu, Oriol Vinyals, and Alex Graves. 2016. Conditional Image Generation with PixelCNN Decoders. In *NIPS*, pages 4790–4798.

- Bo Pang and Lillian Lee. 2008. Opinion mining and sentiment analysis. *Foundations and Trends® in Information Retrieval*, 2:1–135.
- Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global Vectors for Word Representation. In *EMNLP*, pages 1532–1543.
- Maria Pontiki, Dimitrios Galanis, John Pavlopoulos, Haris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. 2014. Semeval-2014 task 4: Aspect based sentiment analysis. In *SemEval@COLING*, pages 27–35, Stroudsburg, PA, USA. Association for Computational Linguistics.
- Sebastian Ruder, Parsa Ghaffari, and John G Breslin. 2016a. A Hierarchical Model of Reviews for Aspect-based Sentiment Analysis. In *EMNLP*, pages 999–1005.
- Sebastian Ruder, Parsa Ghaffari, and John G Breslin. 2016b. INSIGHT-1 at SemEval-2016 Task 5 - Deep Learning for Multilingual Aspect-based Sentiment Analysis. In *SemEval@NAACL-HLT*, pages 330–336.
- Richard Socher, Alex Perelygin, Jean Y Wu, and Jason Chuang. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *EMNLP*, pages 1631–1642.
- Kai Sheng Tai, Richard Socher, and Christopher D Manning. 2015. Improved Semantic Representations From Tree-Structured Long Short-Term Memory Networks. In *ACL*, pages 1556–1566.
- Duyu Tang, Bing Qin, Xiaocheng Feng, and Ting Liu. 2016a. Effective LSTMs for Target-Dependent Sentiment Classification. In *COLING*, pages 3298–3307.
- Duyu Tang, Bing Qin, and Ting Liu. 2015. Document Modeling with Gated Recurrent Neural Network for Sentiment Classification. In *EMNLP*, pages 1422–1432.
- Duyu Tang, Bing Qin, and Ting Liu. 2016b. Aspect Level Sentiment Classification with Deep Memory Network. In *EMNLP*, pages 214–224.
- Wenya Wang, Sinno Jialin Pan, Daniel Dahlmeier, and Xiaokui Xiao. 2016a. Recursive Neural Conditional Random Fields for Aspect-based Sentiment Analysis. In *EMNLP*, pages 616–626.
- Yequan Wang, Minlie Huang, Xiaoyan Zhu, and Li Zhao. 2016b. Attention-based LSTM for Aspect-level Sentiment Classification. In *EMNLP*, pages 606–615.
- Jason Weston, Sumit Chopra, and Antoine Bordes. 2014. Memory Networks. In *ICLR*, pages CoRR abs–1410.3916.
- Jiacheng Xu, Danlu Chen, Xipeng Qiu, and Xuanjing Huang. 2016. Cached Long Short-Term Memory Neural Networks for Document-Level Sentiment Classification. In *EMNLP*, pages 1660–1669.
- Wei Xue, Wubai Zhou, Tao Li, and Qing Wang. 2017. Mtna: A neural multi-task model for aspect category classification and aspect term extraction on restaurant reviews. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, volume 2, pages 151–156.
- Meishan Zhang, Yue Zhang, and Duy-Tin Vo. 2016. Gated Neural Networks for Targeted Sentiment Analysis. In *AAAI*, pages 3087–3093.