CITS4012 Natural Language Processing Project

你的名字(你的学号)

Abstract

Aspect-based sentiment analysis (ABSA) is a fine-grained task aimed to classify the sentiment polarities of a part of sentence with a given aspect. This paper investigates the impact of different methods of integrating aspect information on the final performance when performing ABSA tasks. By using the attention mechanism to detect sentence parts related to a given aspect, three GRU-based models are proposed, including changing the location of the aspect in the input data and whether to include the aspect embedding in the attention calculation. An Accuracy of 66.4% has been reached on MAMS-ACSA test dataset [1]. Detailed ablation study and a visualise qualitative case study are introduced.

1 Introduction

With the development and prosperity of e-commerce, analyzing user feedback has become an important task. By understanding the reviews written by users, e-commerce platforms can better understand how their products or services appear in the minds of users, and can better improve them. However, a single piece of user review often contains different aspects of information, and the sentiments they reflect may be completely different. In this case, analyzing user reviews from the sentence level is not a reasonable choice because it is difficult to say what the sentence as a whole wants to express. Classifying the sentiments represented by different components of a sentence is a task called Aspect-based sentiment analysis (ABSA), as Figure 1 explains. As a fine-grained task in sentiment analysis, ABSA requires algorithms to find the location of a part of sentence that represents a given aspect. With Figure 1, given the aspect "food", a ABSA solution should notice the first half of the sentence "The dumplings here are really good", thereby getting a positive sentiment; given the "environment", the second half of the sentence "but the table was not wiped clean" elicits a negative sentiment.

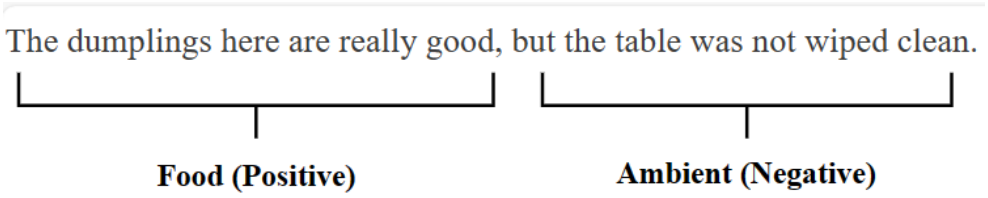

Figure 1: An ABSA Example, showing a sentence with two different aspects which are having two different polarities.

Datasets related to this task have also been proposed, including SemEval-2014 Restaurant Review dataset and Laptop Review dataset [2].

After 2016, neutral networks is taking a place in this area. In 2016, Attention-based LSTM [3] combined aspect's embedding and input embedding, sent the concated embedding to a LSTM sequence, then combined aspect embedding again with LSTM's hidden states to calculate attention, thus made concentration on different aspects of input sentence possible. In 2019, pre-trained language model (PLM) BERT is introduced into this task [4]. In 2020, Sentence Constituent-Aware Network (SCAN) [5] was proposed graph attention modules into this task.

Other forms of networks for this task also exist. In 2020, Hierarchical Graph Convolutional Network (Hier-GCN) [6] is proposed. It first detects the relationship between different aspects in the sentence through a low-level GCN, and then use a high-level GCN to detect the polarity to which each aspect belongs.

In 2019, a new data set called MAMS was proposed [1], in order to make up for the lack of aspect and polar diversity of most sentences in the above two data sets [2], which easily degenerates into the task of sentence-level sentiment analysis. MAMS-ACSA is a sub-dataset that focuses on the Aspect Category sentiment analysis (ACSA) task. Each sentence has at least two aspects of different polarities. Some methods that perform well in the SemEval-2014 data set cannot achieve good results in MAMS. The article also proposed CapsNet based on Capsule and CapsNet-BERT, a version that uses BERT to replace the embedding and encoding parts. It defeated the models that performed well on other data sets at the time on the MAMS data set.

This paper studies the impact of combining aspect information in different ways on the final performance when performing ABSA tasks. The attention mechanism is used to detect the parts of the sentence that is associated with given aspect. We proposed three GRU-based models, each using different aspects information integration methods: including changing the location of the aspect in the input data, and whether to include aspect embedding into the calculation of attention. The main contributions of this article are as follows:

(a) Proposed two seq2seq models with GRU as the basic unit, and studied the different roles of aspects at the end of input and at the beginning of Decoder.
(b) Implemented the GRU version of ATAE-RNN, inspired by ATAE-LSTM [3]
(c) Visual analysis of the Attention used by the above three models

In the next section Methods, we will elaborate on the structure of our models and their parameters within. In section Experiments, we will describe the process of training the model, including a brief overview of the data set, preprocessing methods and hyperparameter settings. In Results, we will conduct quantitative and qualitative analysis based on the results obtained in Experiments. Finally, we will summarize our findings and the limitations of this study in the Conclusion section.

2 Methods

2.1 Attn-GRU (aspect in sentence)

To capture information from sequential data like sentences, we adopt RNN components in our model. We first came up with a thought that regrad ABSA as a generation task on the seq2seq model [7]. seq2seq is a Recurrent neutral Network (RNN) architecture,

including the encoder part that gradually encodes the input sequence to form a hidden state sequence, and the decoder part will decode the hidden state of the encoder and finally obtains the output sequence. We choose GRU because compared with vanilla RNN, GRU has gating mechanism preventing it from losing gradients in long sequences. Compared with LSTM, it is relatively simple and saves us more time from training.

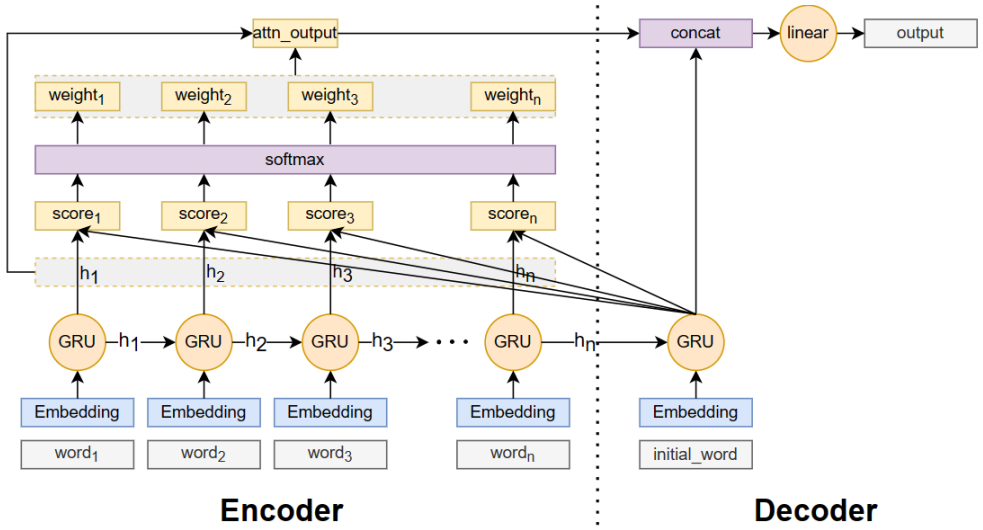

Figure 2: Model Structure of Attn-GRU (aspect in sentence) and Attn-GRU (aspect in decoder input)

Figure 2 describes the structure of this model. Input data first goes to embedding layer to become an embedding vector that GRU could handle. Dropout layer is used after Decoder's embedding for better generalization performance. The GRU component accepts embedding and one hidden state, and also outputs hidden state, output hidden state will be seen as the represetation of the input data. Decoder will accept the last hidden state of encoder and decode it. In order to give the model the ability to distinguish different components in a sentence, attention mechanism is used, more specifically, in our first and second model, dot-product attention is used, as Equation 1 shows, where s_t is all encoder hidden states, and h_0 is the decoder hidden state, as we use a genarative model architecture to do a classification task, the decoder as output part only need one hidden state.

$$\text{score}(s_t, h_0) = s_t^\top h_0 \tag{1}$$

Combining the hidden state of decoder with the hidden states of encoder at each time step by matrix multiplication and then do softmax to get the probality distribution of each word to appear in the final attention output, called attention weights. Using attention weights to multiply the hidden states of encoders, we get a final fusion of all hiddens in encoder, this fusion hidden state represents the most important part of the sentence in terms of one aspect.

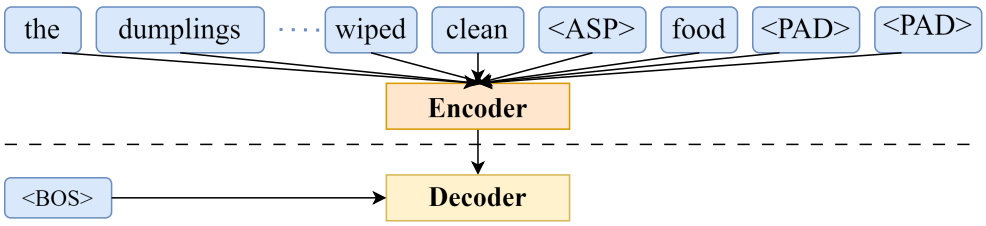

Figure 3: Attn-GRU (aspect in sentence)'s input, noted that aspect "food" is along with input sentence.

In terms of integrating aspect information, this model does not explicitly add aspect information during the training or attention calculation process. Instead, the aspect will be added to the end of the input, like shown in Figure 3, and Decoder will use a predefined word "<BOS>" to initialize decoder hidden state. .

2.2 Attn-GRU (aspect in decoder input)

This model has no structural differences from the previous one. The only difference is that aspect information will be input as the initial word of the Decoder, replacing the initial word "<BOS>" in the previous model. That is, compared to the previous model, this model calculates attention with a hidden state that not been propagated through the second GRU component. Its function is equivalent to the previous model calculating self-attention for the last word of the input. Input structure is showed in Figure 4.

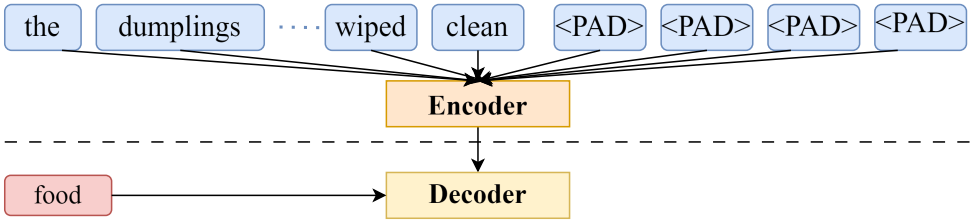

Figure 4: Attn-GRU (aspect in decoder input)'s input, noted that aspect "food" is passed to Decoder.

2.3 Attention-based GRU with Aspect Embedding (ATAE-GRU)

Inspired by ATAE-LSTM [3], we also tried to directly involve aspect information in the attention calculation instead of using the hidden layer that changes through timesteps; and aspect embedding is also used for the input of each GRU. Aspect will enter EMbedding alone, and then merge with inpu sentence embedding to enter the GRU sequence.

3 Experiments

3.1 Datasets

Dataset used in our study is MAMS-ACSA, a sub-dataset of MAMS [1] designed for Aspect-Category Sentiment Analysis task. The Topic of this MAMS-ACSA is Restaurant reviews, including 8 aspects: food, service, staff, price, ambience, menu, place and miscellaneous. Each sentence has at least two different aspects with different sentiment polarities. Detailed Stataics of this dataset can be found in Table 1, including polarities counting of train, validation and test dataset.

	negative	neutral	positive	total
train	2084	3077	1929	7090
validation	259	388	241	888
test	263	393	245	901

Table 1: Stataics of MAMS-ACSA dataset

3.2 Experiment Setup

We first preprocessed the input sentences, including expanding contraction words, removing punctuation, and convert all words in sentence to lower case. To make full use of GPU computing power, we use batch training and padding sentences to align with the longest sentence using symbol "<PAD>". These padding symbol will have an embedding of all zeros, during training, these word embedding will be packed to a shorter length to avoid the effect of these meaningless symbols. After GRU component, we will restore the length to the max length sentence in the batch. Decoder will also mark zero vectors to "-inf" to prevent these effect to attention. All model embedding layers will use GloVE's pre-trained weights [8]. We noticed that the GloVE's dictionary didn't contain every word in our data set, less than 1000 words were missing, most of which are typos and numbers. So we will not freeze the gradient to make the embeddings of these words can also be learned in the later training process.

Loss function we used is Negative Log-Likelihood Loss (NLLLoss), the target is to minimize this funciton, as Equation 2, where y being target distribution and $\hat{y}$ being predicted distribution. All loss we mentioned is epoch loss, given n in Equation 2 refers to the counts of whole dataset. Adam optimizer is used to decrease our target loss.

$$\text{loss} = -\sum_{n=1}^{n}\big(y_i \log \hat{y}_{\theta,i} + (1-y_i)\log\big(1-\hat{y}_{\theta,i}\big)\big) \tag{2}$$

Grid Search is employed to search for a optimal pair of batch size and learning rate. The batch size range is 32 and 64, and the learning rate is from 3e-5 to 9e-5, stepping at a rate of 1e-5, for a total of 12 combinations of parameters. Hidden size is set to 50, dropout is set to 0.1. To avoid missing the convergence, the learning rate decays to the 0.975 times of its original every 10 epochs of training. Each search will train a new model in 60 epochs. After each epoch training, a validation test will be performed on the validation set, the accuracy and the loss on the validation set will be recorded and be used as a metric to choose optimal parameters. After all training tests, the one parameter sets gave us the best accuracy performance with a steady declining loss curve on the validation set will be selected. After The best parameter is selected, we will apply them to the models we have and evaluate them on test set 5 times to report the average accuracy.

The best parameters we found for Attn-GRU (aspect in sentence) are batch size 32 and learning rate 6e-5. For Attn-GRU (aspect in decoder), we used batch 32 and a higher learning rate 8e-5. For our last model ATAE-GRU, we found that within 50 epochs, it is hard to converge, so we just obtain a set of best parameters from 50 epochs experiments, then use this setting but increase training epochs to 80 to get a final model, with batch size 64 and learning rate 8e-05.

4 Results

4.1 Quantitative Results

The accuracy of all our models on the test set is shown in Table 2. It can be seen that except for the first model, the other two models perform similarly. Attn-GRU (aspect in decoder) surpasses ATAE-GRU with a slight advantage of 0.2%. This shows that the location where the aspect is added plays an important role. When the aspect is after the input sentence, the attention mechanism will focus on the relationship between "<BOS>" and the various words in the input sentence, which is meaningless. A GRU that accepts hidden state with aspect information cannot complete the ABSA task well.

Models	**Accuracy**(%)
Attn-GRU(aspect in sentence)	45.13
Attn-GRU(aspect in decoder)	66.48
ATAE-GRU	66.26

Table 2: Testing results of three model proposed

4.2 Ablation Studies

We use Attn-GRU (aspect in decoder) to carry out ablation study. Note that the difference between Attn-GRU (aspect in sentence) and this model is only the lcoation of the aspect, so it can also be used as a comparison object. The analysis of the two has been Described in last section.

We also tried not using GloVE pretrained word vectors, instead we train our own embeddings from scratch. The results showed that the accuracy dropped to 61.11%, indicating that the pre-training vector expresses the meaning of words more accurately than our own trained embedding. We also tried to use GloVE but did not turn on gradient propagation at all, thus ignoring the existence of the approximately 1,000 unknown words mentioned above. Experiments showed that the accuracy rate was 66.59% with almost no change. This shows that our data is not enough to learn the meaning of these words, and whether fixed embedding is used or not has no impact on the final performance. Finally, a replacement of dot product attention is carried out by using scaled dot product. The accuracy dropped to 65.08%, which was not very obvious. Since there are no issues found such as gradient explosion or disappearance that lead to unstable training when using dot product, the addition of scaling factors is not suitable for this scenario.

4.3 Qualitative Results

In order to more intuitively demonstrate the differences between each model and how the attention mechanism works, we selected a case in the test set(index 604 and 605). The sentence of this case contains two different aspects: service and food, and its corresponding polarity is negative and positive. We noticed that neutral accounts for the majority in the data set, so when selecting use cases, we will avoid cases that are originally neutral to prevent the model from outputting neutral directly to obtain a smaller loss.

Figure 5 ~ Figure 7 showed the attention weight of each word, the deeper background color is, the more important the word is. Color has been adjusted by dividing the max attention weight in this sentence to enhance visualisation.

From Figure 5, we can see that although Attn-GRU (aspect in sentence) has an attention mechanism, because the aspect information is not in place, its attention weights do not make a difference when the input aspects are different, and the maximum value is 0.1 , so no significance differences between it and other words. And all prediction errors in the final result are neutral.

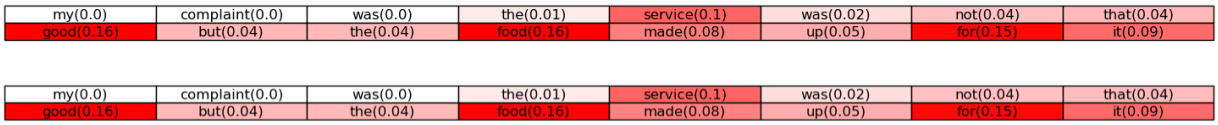

Figure 5: up to down: service_negative, predicted as neutral; food_positive, predicted as neutral;

From Figure 6, we can see that Attn-GRU (aspect in decoder) captures “not” and “but”, two words that clearly refer to sentiment polarity, but these two words should belong to different aspects in this sentence. More, there is only a very slight difference in the attention weights on both sides. Although it correctly predicts polarity in the service case, it lacks reasonable interpretability.

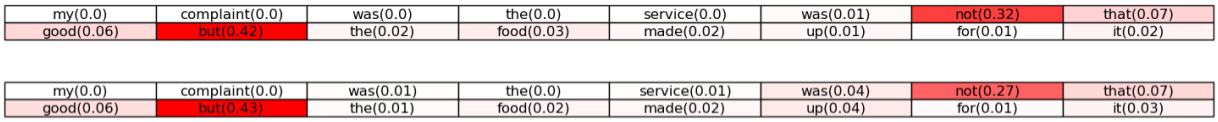

Figure 6: up to down: service_negative, predicted as negative; food_positive, predicted as neutral;

From Figure 7, we can see that ATAE-GRU shows the importance of different sentence components under different aspects. When the aspect is service, the words related to service have some weight, and the positive words related to food are also captured, the word “but” turns the situation around. When the aspect is food, only the second half of the sentence about food maintains a higher weight, lead the prediction to correct.

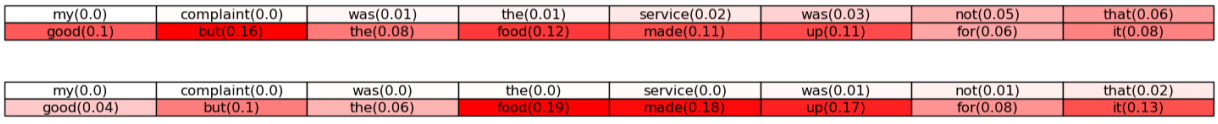

Figure 7: up to down: service_negative, predicted as negative; food_positive, predicted as positive;

5 Conclusion

By comparing models that integrating aspect information in three different ways, we discovered the importance of the attention mechanism in the ABSA task. Using GloVe pre-trained word embeddings, the RNN model combined with the Attention mechanism can achieve an accuracy of 66.4% on the MAMS-ACSA test data set. At a hidden size of 50, the well-trained model and ATAE-GRU have similar accuracy, but ATAE-GRU has better interpretability in terms of attention weights. Due to time limitations and the predefined topic of this project, we did not explore graph models and use pre-trained language models, which is popular nowadays. We will focus on combining multiple tasks

together to solve this task like employing an addition module to detect aspect category first.

Bibliography

[1] Q. Jiang, L. Chen, R. Xu, X. Ao, and M. Yang, "A Challenge Dataset and Effective Models for Aspect-Based Sentiment Analysis," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, K. Inui, J. Jiang, V. Ng, and X. Wan, Eds., Hong Kong, China: Association for Computational Linguistics, Nov. 2019, pp. 6280–6285. doi: 10.18653/v1/D19-1654.

[2] D. Kirange, R. R. Deshmukh, and M. Kirange, "Aspect Based Sentiment Analysis Semeval-2014 Task 4," *Asian Journal of Computer Science and Information Technology (AJCSIT) Vol*, vol. 4, 2014.

[3] Y. Wang, M. Huang, X. Zhu, and L. Zhao, "Attention-Based LSTM for Aspect-level Sentiment Classification," in *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, J. Su, K. Duh, and X. Carreras, Eds., Austin, Texas: Association for Computational Linguistics, Nov. 2016, pp. 606–615. doi: 10.18653/v1/D16-1058.

[4] M. Hoang, O. A. Bihorac, and J. Rouces, "Aspect-Based Sentiment Analysis Using Bert," in *Proceedings of the 22nd Nordic Conference on Computational Linguistics*, 2019, pp. 187–196.

[5] Y. Li, C. Yin, and S.-h. Zhong, "Sentence Constituent-Aware Aspect-Category Sentiment Analysis with Graph Attention Networks," in *Natural Language Processing and Chinese Computing: 9th CCF International Conference, NLPCC 2020, Zhengzhou, China, October 14–18, 2020, Proceedings, Part I 9*, Springer, 2020, pp. 815–827.

[6] H. Cai, Y. Tu, X. Zhou, J. Yu, and R. Xia, "Aspect-Category Based Sentiment Analysis with Hierarchical Graph Convolutional Network," in *Proceedings of the 28th International Conference on Computational Linguistics*, 2020, pp. 833–843.

[7] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to Sequence Learning with Neural Networks," *Advances in neural information processing systems*, vol. 27, 2014.

[8] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global Vectors for Word Representation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014, pp. 1532–1543.