Modeling Aspect Correlation for Aspect-based Sentiment Analysis via Recurrent Inverse Learning Guidance

Longfeng Li[1], **Haifeng Sun**[1], **Qi Qi**[1], **Jingyu Wang**[1], **Jing Wang**[1]*and **Jianxin Liao**[1]
[1]State Key Laboratory of Networking and Switching Technology,
Beijing University of Posts and Telecommunications, Beijing 100876, China
{lilongfeng,hfsun,qiqi8266,wangjingyu,wangjing,liaojx}@bupt.edu.cn

Abstract

Aspect-based sentiment analysis (ABSA) aims to distinguish sentiment polarity of every specific aspect in a given sentence. Previous researches have realized the importance of interactive learning with context and aspects. However, these methods are ill-studied to learn complex sentence with multiple aspects due to overlapped polarity feature. And they do not consider the correlation between aspects to distinguish overlapped feature. In order to solve this problem, we propose a new method called Recurrent Inverse Learning Guided Network (RILGNet). Our RILGNet has two points to improve the modeling of aspect correlation and the selecting of aspect feature. First, we use Recurrent Mechanism to improve the joint representation of aspects, which enhances the aspect correlation modeling iteratively. Second, we propose Inverse Learning Guidance to improve the selection of aspect feature by considering aspect correlation, which provides more useful information to determine polarity. Experimental results on SemEval 2014 Datasets demonstrate the effectiveness of RILGNet, and we further prove that RILGNet is state-of-the-art method in multi-aspect scenarios.

1 Introduction

Aspect-based sentiment analysis (ABSA) aims to distinguish the orientation of sentiment existed on every aspect (Liu, 2012). For example, in the sentence: *"Food is usually very good, though I wonder about freshness of raw vegetables"*, aspect *"raw vegetables"* has negative polarity and *"food"* has positive polarity. ABSA task consists of two subtasks called aspect extraction and aspect sentiment classification (Zhang et al., 2018). In this paper, we focus on aspect sentiment classification task and assume that aspects are known (Majumder et al., 2018; Jiang et al., 2020). As shown in above case, multiple sentiment polarities could exist in one sentence, which leads to the overlapping use of multiple aspects when predicting the polarity of current aspect (Du et al., 2019; Jiang et al., 2020). We can see that sentiment polarity of *"food"* comes from word *"good"* which implies positive. *"raw vegetables"* does not have any words linked in above case. However, conjunction *"though"* determines sentiment polarity of *"raw vegetables"* which implies negative. Feature about *"good"* is overlapped by these two aspects and conjunction *"though"* leads to different sentiment polarity between *"food"* and *"raw vegetables"*. In this case, two aspects exhibit high correlation. Therefore, modeling the correlation between multiple aspects is a suitable way to distinguish overlapped feature. From the above case, it is not difficult to see that aspect correlation comes from these sources: aspects, their sentiment polarities and conjunctions. Conjunction establishes correlation between aspects. If model learns correlation between aspects, it can make right predition of *"raw vegetables"* from the feature of *"food"*.

Previous works have been devoted to introduce attention mechanism into neural network and great progress has been made (Wang et al., 2016; Ma et al., 2017). Self-attention-based models such as BERT (Devlin et al., 2019) have been applied to ABSA by using BERT as an encoder (Yu and Jiang, 2019; Lin et al., 2019). To handle the overlapped aspect feature, some works aim at decoupling overlapped feature for every aspect. Du et al. (2019) uses a Capsule Network and Interactive Attention to decouple overlapped feature for predicted aspect. In another way, Majumder et al. (2018) tries to model inter-aspect relation by learning every aspect feature respectively. However, these methods only use aspect (e.g., *"food"*) and not use sentiment polarity (e.g., *"positive"*). Aspect and sentiment polarity both reflect aspect correlation. Considering aspect correlation is an ef-

* Corresponding author.

Proceedings of the 29th International Conference on Computational Linguistics, pages 6887–6896
October 12–17, 2022.

fective way to determine the polarity of aspect that is not easy to find linked words. We find that inverse learning (Zhou and Small, 2021; Putzky and Welling, 2017) is a suitable way to excavate the correlation from aspect and sentiment polarity because it learns from input (aspect and sentiment polarity) to output (generated sentence). And generated sentence is same with aspect and sentiment polarity (Radford et al., 2018; Luong et al., 2015) in correlation.

In this paper, we design a Recurrent Inverse Learning Guided Network (RILGNet) to improve the performance of ABSA task through modeling aspect correlation. Our RILGNet has two point to improve aspect correlation modeling and aspect feature selecting. First, we use Recurrent Mechanism (Kolluru et al., 2020) to learn aspect joint representation and model aspect correlation. And we use the correlation and representation to make current prediction. Recurrent Mechanism can find clear stated aspect polarities (*"food"* and positive in above sentence), and then use them to re-find indirect aspect polarities (*"raw vegetables"* and negative). For this purpose, Recurrent Mechanism is a suitable way. Second, we use Inverse Learning Guidance (ILG) to excavate the correlation from aspects and sentiment polarities. We use inverse learning to generate a sentence by using aspect-polarity combination. We consider the difference between the given sentence and the generated sentence embedding by calculating mean squared error (MSE) and its gradient. This gradient expresses difference in correlation between the prediction and the ground-truth. We fuse this gradient to aspect joint representation, which makes the selection of aspect feature focus more on high correlation aspects. We show that RILGNet achieves 83.70% and outperforms state-of-the-art method by 1.34% on average in SemEval 2014 two distinct domains: laptop and restaurant. And we also prove that RILGNet is state-of-the-art method in multi-aspect scenarios.

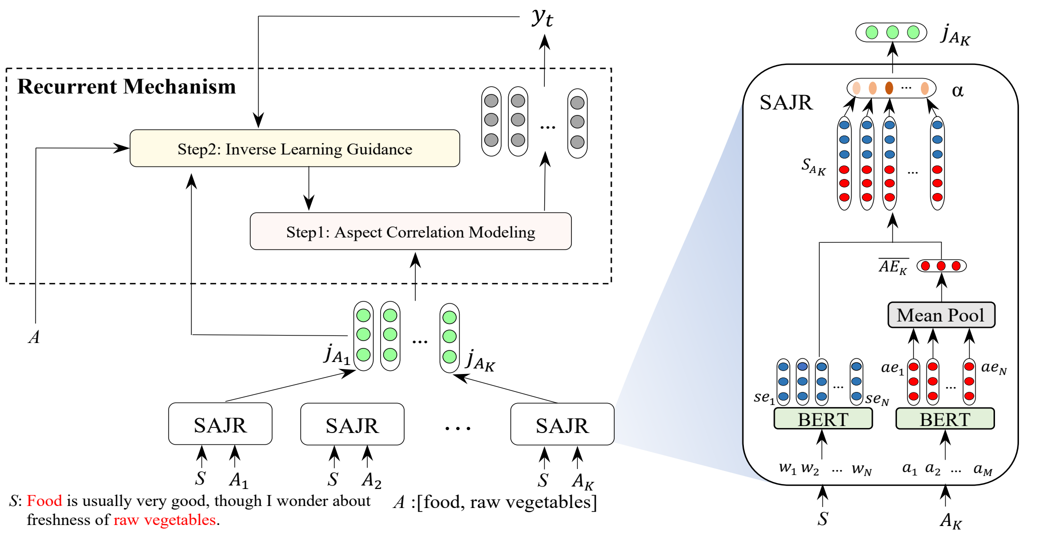
The main contributions are as follows: (1) RILGNet improves the performance of ABSA task by modeling aspect correlation from aspect and sentiment polarity. (2) We design an ILG to get guidance about how aspect joint representations are modified and how features are selected from current prediction. (3) Experimental results show that our method outperforms state-of-the-art method on SemEval 2014 Datasets, especially in multi-aspect scenarios.

2 Related Works

The traditional methods of ABSA find optional words about aspect in sentence, and then get the corresponding sentiment through special designed aspect features such as sentiment lexicon, n-grams, and dependency information, which is labor-intensive (Boiy and Moens, 2009; Kiritchenko et al., 2014). Meanwhile, the quality of feature directly affects classification accuracy.

Motivated by attention mechanism in the field of deep learning, some ABSA's works focus on integrating attention into neural network, such as RNNs (Wang et al., 2016; Chen et al., 2017) and memory networks (Ma et al., 2017; Majumder et al., 2018), to learn attention distribution of different aspects and extract aspect-aware sentence representation. Most recently, self-attention-based models such as BERT (Devlin et al., 2019) have been applied to ABSA by using BERT as the embedding layer (Yu and Jiang, 2019; Lin et al., 2019). These works effort to focus on understanding the context based on aspect and have achieved great performance improvement.

Besides the above works, Chen et al. (2020); Zhang and Qian (2020); Tay et al. (2018); Li et al. (2019); Mao et al. (2019) use syntactic information and improved semantic understanding to improve end-to-end performance. Graph-based models are applied in ABSA task by utilizing word relations through dependency parses to facilitate the task with better semantic guidance for analyzing context and aspect words (Tian et al., 2021; Wang et al., 2020; Li et al., 2021; Hou et al., 2021). They realize that the overlapped feature about multi-aspect is ignored when the model predicts current aspect. Majumder et al. (2018) calculates the aspect-aware sentence representation for every aspect in a sentence, which models the inter-aspect relation in a sentence for handling the overlapped feature from aspect information. Du et al. (2019) uses a Capsule Network and Interactive Attention to decouple overlapped feature to predicted aspect. Jiang et al. (2020) uses mutual enhanced transformation to obtain more effective representation of aspect and context for handling overlapped feature. However, unlike our proposed method, they only use aspect information without sentiment polarity information. Aspect and sentiment polarity both reflect aspect correlation.



(a) Overall Framework

(b) Sentence-Aspect Joint Representation

Figure 1: The overall framework of RILGNet and Sentence-Aspect Joint Representation. When the recurrent round is 1, the output of Inverse Learning Guidance is 0.

3 Proposed Method

In this section, we formalize the task and describe proposed Recurrent Inverse Learning Guided Network (RILGNet) in detail. The input and output of ABSA task are as follows:

Input : We give a sentence $S = [w_1, w_2, ..., w_N]$, known aspect set $A = [A_1, A_2, ..., A_K]$ and every aspect $A_i = [a_1, a_2, ..., a_M]$, where N and M are the maximum number of sentence and aspect word respectively, K is the maximum number of aspect.

Output : Sentiment Classification Result $y \in \{positive, negative, neutral\}$ for each predicted aspect A_i.

The framework of RILGNet is shown in Figure 1 (a). It can be roughly divided into two parts, namely Sentence-Aspect Joint Representation (SAJR) and Recurrent Mechanism. Recurrent Mechanism can be further divided into two steps: Aspect Correlation Modeling and Inverse Learning Guidance. The core idea is how to model aspect correlation in A about A_i and how to use correlation to discriminate overlapped feature and determine sentiment polarity y. In the t_{th} recurrent round, RILGNet uses sentence S and aspect set A to obtain the SAJR $J_A = [j_{A_1}, j_{A_2}, ..., j_{A_K}]$. Then RILGNet sends J_A and A to Recurrent Mechanism and get current prediction y_t. We choose y_T as final prediction.

3.1 Sentence-Aspect Joint Representation

Different words in a sentence don't contribute equally to sentimental information for a current aspect (e.g., adjectives often contribute more than verbs). This requires a sentence representation to reflect the sentiment of different aspect. As shown in Figure 1 (b), we design Sentence-Aspect Joint Representation (SAJR) to achieve this. We get sentence representation $SE = [se_1, se_2, ..., se_N] \in \mathbb{R}^{N \times D}$ from sentence S and aspect representation $AE_i = [ae_{1,i}, ae_{2,i}, ..., ae_{M,i}]$ from aspect A_i by BERT (Devlin et al., 2019) firstly, where D is the size of hidden state in BERT. For multi-word aspect, $\overline{AE_i}$ is averaged by AE_i. Then we concatenate SE and $\overline{AE_i}$ in all words for $S_{A_i} \in \mathbb{R}^{N \times 2D}$:

$$S_{A_i} = [se_1; \overline{AE_i}, se_2; \overline{AE_i}, ...se_N; \overline{AE_i}] \quad (1)$$

In order to differentiate which word contributes more to the sentiment, we utilize an attention layer to obtain sentence-aspect joint representation $j_{A_i} \in \mathbb{R}^D$, indicating which word is important for A_i:

$$z = S_{A_i} W_s + b_s \quad (2)$$

$$\alpha = softmax(z) \quad (3)$$

$$j_{A_i} = Dense(\alpha^T S_{A_i}) \quad (4)$$

where $z = [z_1, z_2, ..., z_N] \in \mathbb{R}^N$, $\alpha \in \mathbb{R}^N$, W_s and b_s are trainable matrices. $j_{A_i} \in \mathbb{R}^D$ and $J_A = [j_{A_1}, j_{A_2}, ..., j_{A_K}]$.

3.2 Aspect Correlation Modeling

We design Aspect Correlation Modeling (ACM) to learn aspect joint representation, then use it to

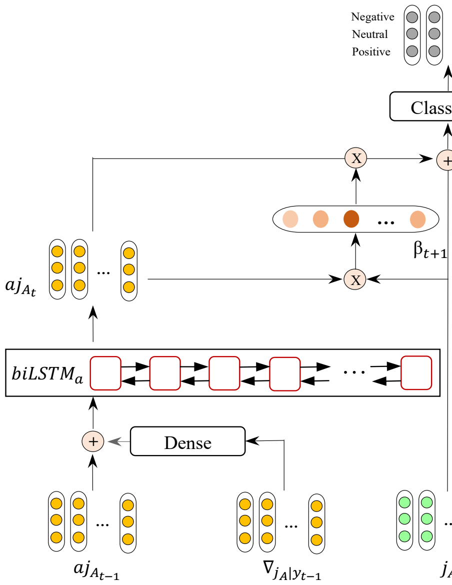


Figure 2: The overall framework of Aspect Correlation Modeling. When recurrent round is 1, $\bigtriangledown_{J_A|y_0}$ is 0.

model aspect correlation and get current prediction y_t. Specially, as shown in Figure 2, we use a bi-LSTM to get current aspect joint representation $AJ_{A,t} = [aj_{A_{1,t}}, aj_{A_{2,t}}, ..., aj_{A_{K,t}}] \in \mathbb{R}^{N \times D}$ at t_{th} recurrent round following Majumder et al. (2018):

$$IJR_{t-1} = tanh(\bigtriangledown_{J_A|y_{t-1}} W_y + b_y) \quad (5)$$

$$AJ_{A,t} = biLSTM_a(AJ_{A,t-1} + IJR_{t-1}) \quad (6)$$

where $AJ_{A,0}$ is J_A from Equation (4) and $\bigtriangledown_{J_A|y_0}$ is 0, W_y and b_y are trainable matrices, $\bigtriangledown_{J_A|y_{t-1}}$ is inverse learning guided gradient from (12). We will introduce this gradient in the next section.

After this, in order to get aspect correlation between A_i and other aspects, we employ an attention layer between every aspect query representation j_{A_i} and $AJ_{A,t}$ as:

$$z_c = j_{A_i}(AJ_{A,t})^T \quad (7)$$

$$\beta_t = softmax(z_c) \quad (8)$$

where $z_c = [z_{c,1}, z_{c,2}, ..., z_{c,K}] \in \mathbb{R}^{K \times 1}$, $\beta_t = [\beta_{t,1}, \beta_{t,2}, ..., \beta_{t,K}] \in \mathbb{R}^{K \times 1}$ and j_{A_i} comes from (4). Here, β_t measures the aspect correlation, stating the contribution of A to predicting A_i. It is worth noting that correlation score β_t does not indicate positive or negative correlation between different aspects, but only indicates how siginificant correlation is achieved.

Then, we obtain the aspect joint representative output vector $output_t \in \mathbb{R}^{1 \times D}$ by:

$$output_t = (\beta_t)^T AJ_{A,t} \quad (9)$$

Finally, aspect query representation j_{A_i} is added with $output_t$ to generate refined aspect representation, and is passed to a softmax classifier as:

$$oj_{i,t} = softmax((j_{A_i} + output_t)W_p + b_p) \quad (10)$$

$$y_{i,t} = argmax(oj_{i,t}) \quad (11)$$

where W_p and b_p are trainable matrices. $y_{i,t}$ is current prediction for A_i, and we concatenate all $y_{i,t}$ for every aspect A_i to get current prediction y_t.

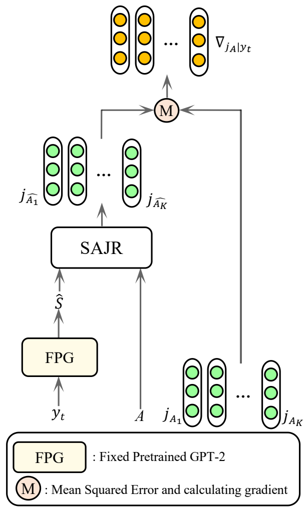


Figure 3: The overall framework of Inverse Learning Guidance.

3.3 Inverse Learning Guidance

The motivation of Inverse Learning Guidance (ILG) is to provide prompt to model aspect correlation by excavating aspect and sentiment polarity. This prompt can give feedback on how aspect correlation is disturbed from current prediction y_t. We use the idea of inverse learning (Zhou and Small, 2021; Putzky and Welling, 2017) to correct the disturbed aspect correlation. It can set aspect-polarity combination as input to generate sentence $\hat{S}$ which is same with y_t about aspect correlation. The difference between $\hat{S}$ and given sentence S is an embodiment for the difference between y_t and the ground-truth $\hat{y}$ in aspect correlation.

The inverse learning model is implemented by GPT-2, which has shown effective at generating the textual language based on specific information(Radford et al., 2018). If GPT-2 can generate a sentence that is more similar to the given sentence in semantics, prediction y_t will be closer to the ground-truth $\hat{y}$ and y_t will have more similar aspect correlation with $\hat{y}$. Conversely, when the generated sentence is quite different from the given sentence, despite some noise, ILG can reflect the difference by high or low gradient because of the inconsistent aspect-polarity combinations. Gradient guides the selection of features to focus more on high correlation aspect. GPT-2 is finetuned by inverting the input and output of training data. The parameters of GPT-2 will be fixed during training process. The details of the structure and training for GPT-2 is in Section 4.2.

The structure of ILG is shown in Figure 3. Specifically, we use a template like *"[A_1: polarity, A_2: polarity, ..., A_K: polarity.]"* to generate a sentence $\hat{S}$, and polarities come from y_t. We then use the $\hat{S}$ and A to get $\hat{J_A} = [\hat{j_{A_1}}, \hat{j_{A_2}}, ..., \hat{j_{A_K}}] \in \mathbb{R}^{K \times D}$. We select the mean squared error (MSE) to reflect aspect-level difference between $\hat{J_A}$ and J_A about aspect correlation. And we calculate its gradient as output of this step, which is defined as $\bigtriangledown_{J_A|y_{t-1}} \in \mathbb{R}^{K \times D}$:

$$\bigtriangledown_{J_A|y_{t-1}} = \nabla(\frac{1}{D}\sum_{d=1}^{D}(\hat{J_{A,d}} - J_{A,d})^2) \quad (12)$$

where D is the dimension of sentence embedding. $\bigtriangledown_{J_A|y_{t-1}}$ expresses the difference about correlation between S and $\hat{S}$. We fuse $\bigtriangledown_{J_A|y_{t-1}}$ to ACM in (5) and (6), which makes the selection of aspect features focus more on high correlation aspects.

3.4 Training

We train RILGNet using Cross Entropy as loss function at every recurrent round and averaging them as:

$$L = -\frac{1}{NT}\sum_{t=1}^{T}\sum_{n=1}^{N}\hat{y_n} log y_{nt} \quad (13)$$

where $\hat{y_n}$ is the ground-truth, y_{nt} is the prediction in round t, N is the number of samples and T is the maximum recurrent round.

4 Experiment

4.1 Datasets

We evaluate our model with SemEval-2014 ABSA dataset [1] (Pontiki et al., 2014). It contains two domain datasets: Laptop and Restaurant. Table 2 shows the count of samples in total and multi-aspect sentences.

4.2 Implements

BERT: BERT Encoder employs a pre-trained BERT-based model ("BERT-base-uncased") for fine-tuning (Devlin et al., 2019). The hidden state size is 768 and the number of self-attention layers is 12. The total number of parameters for the pre-trained BERT model is 110M in our experiments. The length of word list is 30,522. The max sequence length is 100. We used the last hidden state of BERT for word representation and fine-tuned them on our method. The batch size is 128 in training and 64 in testing. RILGNet is trained by the Adam optimizer (Kingma and Ba, 2015) with the default configuration with 20 epochs. The learning rate is 5×10^{-5} in RILGNet.

GPT-2: In the ILG part, we use pre-trained GPT-2 (Radford et al., 2018) to generate a sentence. We finetune GPT-2 by using a sentence as *"[A_1: polarity, A_2: polarity, ..., A_K: polarity.]"* to get generated sentence. A_i and their polarities come from ground truth. Finally, we use the Corss Entropy as loss function by predicting the next word. We train our model in 20 epochs with batch size 4, and the learning rate is 1.5×10^{-4} with same Adam optimizer in other parts of RILGNet. The hidden state size is 768, the number of self-attention heads is 12. The max sequence length is 100, which is same with BERT. And we fix the parameter of GPT-2 when we train the other side. We use Top-5 Sampling to get next candidate word when generating sentence. We choose next word in their probabilities on these five words.

Aspect Correlation Modeling: In this step, the hidden state size of the bi-LSTM is 768 with one layer. The classifier is a Fully Connected Layer, and the output size is the known aspect set size $\times$ 3(*Positive, Negative, Neutral*).

4.3 Compared Methods

We compare our method with the following baseline methods:

[1] http://alt.qcri.org/semeval2014/task4

Model	Laptop		Restaurant	
	Accuracy	Macro-F1	Accuracy	Macro-F1
SVM (Kiritchenko et al., 2014)	70.49	-	80.16	-
AE-LSTM (Wang et al., 2016)	68.90	-	76.60	-
ATAE-LSTM (Wang et al., 2016)	68.81	63.11	77.68	64.89
IAN (Ma et al., 2017)	72.57	66.73	78.48	67.55
Cabasc (Liu et al., 2018)	75.07	-	80.89	-
IARM (Majumder et al., 2018)	73.80	-	80.00	-
TNet-LF (Li et al., 2018)	75.08	69.78	80.18	70.06
HGMN (Ran et al., 2019)	76.67	72.22	82.33	73.34
IACapsNet (Du et al., 2019)	76.80	73.29	81.79	73.40
METNet (Jiang et al., 2020)	78.37	74.93	82.50	73.92
DualGCN+BERT (Li et al., 2021)	81.80	78.10	87.13	81.16
GraphMerge (Hou et al., 2021)	81.35	78.65	87.32	81.95
RILGNet w/o Inverse Learning Guidance	81.50	79.02	86.79	80.51
RILGNet	**83.70**	**80.13**	**88.13**	**82.66**

Table 1: The performance of RILGNet on SemEval-2014 ABSA dataset with best performances bolded. "-" indicates not reported in the original paper.

Domain	Train		Test	
	Total	MA	Total	MA
Laptop	2328	1396	638	379
Restaurant	3608	2595	1120	835

Table 2: The number of samples in SemEval-2014 ABSA datatsets. MA means the multi-aspects sentence in datasets.

SVM (Kiritchenko et al., 2014): It is a traditional support vector machine (SVM) based model to get the feature about aspect and sentiment.

AE-LSTM (Wang et al., 2016): AE-LSTM is an attention-based LSTM model, which uses attention mechanism to get the correlation between aspect and words in sentence.

ATAE-LSTM (Wang et al., 2016): ATAE-LSTM is an upgraded version about AE-LSTM. It adds the aspect embedding to the model in order to consider the importance of aspect term to sentiment.

IAN (Ma et al., 2017): IAN interactively learns the context and aspect and extract the feature about sentence and aspect separately.

Cabasc (Liu et al., 2018): Cabasc employs two attention enhancement layers to learn the word order information, aspect information and the correlation between aspect and word in sentence.

IARM (Majumder et al., 2018): IARM obtians aspect-aware sentence representations for all aspects in a sentence to predict the sentiment polarity of current aspect.

TNet-LF (Li et al., 2018): TNet-LF computes the importance of each aspect term based on sentence word rather than the whole sentence dynamically.

HGMN (Ran et al., 2019): HGMN gets the aspect-specific text spans in sentence instead of only the aggregated contextual representation based on attention layer.

IACapsNet (Du et al., 2019): IACapsNet uses the Capsule Network and Interactive Attention to decouple the overlapped feature about multi aspects in a sentence.

METNet (Jiang et al., 2020): METNet uses the mutual enhanced transformation to improve the learning of aspect and fuses the aspect and sentence information by gated convolutional network.

DualGCN+BERT (Li et al., 2021): It uses a dual graph convolutional network that considers the complementarity of syntax structures and semantic correlations simultaneously.

GraphMerge (Hou et al., 2021): It uses a graph ensemble technique, to make use of the predictions from different parsers, combining the dependency relations from different parses.

5 Results

We show the results about different experiments below:

Model	$Laptop^{\#}$		$Restaurant^{\#}$	
	Accuracy	Macro-F1	Accuracy	Macro-F1
TNet-LF (Li et al., 2018)	74.80	67.34	80.31	69.35
IARM (Majumder et al., 2018)	74.10	-	80.48	-
METNet (Jiang et al., 2020)	77.95	73.80	82.59	73.92
RILGNet w/o Inverse Learning Guidance	79.15	77.81	86.59	80.33
RILGNet	**81.27**	**78.72**	**88.02**	**82.61**

Table 3: The performance of RILGNet on $Laptop^{\#}$ and $Restaurant^{\#}$ with best performances bolded. "-" indicates not reported in the original paper.

5.1 Overall Comparison

We assess the overall performance by comparing RILGNet with state-of-the-art methods on SemEval-2014 ABSA dataset. It is evident from the results that RILGNet outperforms all the baselines in Table 1 in both of the domains. GraphMerge (Hou et al., 2021) achieves the best results on Laptop Marco-F1 and Restaurant all metrics due to its progressive graph ensemble technique, making use of the predictions from different parsers and combining the dependency relations from different parses. RILGNet gets bigger improvement in laptop domain (2.35% on Accuaracy and 1.48% on Marco-F1), compared to restaurant domain (0.81% on Accuracy and 0.71% on Marco-F1). This shows that the conbination of aspect correlation modeling and inverse learning guidance in Recurrent Mechanism has overall positive influence on sentiment classification.

In addition, in order to highlight the advantage of RILGNet in multi-aspect scenarios about handling overlapped feature, we conduct further experiments. We delete the single-aspect sentences in two domain test set and sign the new datasets as $Laptop^{\#}$ and $Restaurant^{\#}$ same with Jiang et al. (2020). We report the results of TNet-LF, IARM, METNet and our RILGNet in Table 3. RILGNet achieves significant improvements in two domains, which indicates the effectiveness of RILGNet in multi-aspect scenarios.

5.2 Ablation Study

We evaluate that Inverse Learning Guidance is effective for ABSA task by comparing RILGNet and RILGNet without(w/o) Inverse Learning Guidance. In order to prove above point, we conduct ablation experiments. As shown in Table 1, Compared with RILGNet w/o Inverse Learning Guidance, RILGNet achieves 2.2%, 1.34% in Accuracy, and 1.11%, 2.15% in Marco-F1 improvements on Laptop and Restaurant datasets respectively, which proves the effectiveness of the Inverse Learning Guidance.

The number of recurrent round	Laptop		Restaurant	
	Acc	F1	Acc	F1
1	82.75	78.83	85.53	78.76
2	81.81	76.62	85.89	79.29
3	**83.70**	**80.13**	86.79	80.55
4	81.97	77.43	87.77	82.06
5	81.97	77.43	**88.13**	**82.66**
6	82.44	78.41	86.52	80.00
7	81.50	76.51	85.98	79.91

Table 4: The performance of RILGNet about differnent recurrent round from 1 to 7 on Laptop and Restaurant datasets with best performances bolded.

We also conduct ablation experiments in multi-aspect scenarios. In Table 3, RILGNet achieves 2.12% and 1.43% in Accuracy and 0.91% and 2.28% in Marco-F1 improvements on $Laptop^{\#}$ and $Restaurant^{\#}$ datasets compared with RILGNet w/o Inverse Learning Guidance, which effectively proves the importance of Inverse Learning Guidance in multi-aspect scenarios.

Besides, in order to highlight the advantage of our Recurrent Mechanism, we evaluate the effect of recurrent round number T. Specifically, we conduct the experiments on Laptop and Restaurant datasets and T varies from 1 to 7. In Table 4, we can see that RILGNet achieves the best results when T is 3 in Laptop and T is 5 in Restaurant respectively. We find that irregular recurrent performance showns on RILGNet in restaurant. Certain recurrent round could yield higher quality representation than others, which is more stable in laptop.

(a) Partial attention Weight for aspect “*drinks*”.

(b) Partial attention Weight for aspect “*lychee martini*”.

(c) Partial attention Weight for aspect “*food*”.

Figure 4: The partial attention weight of all aspect for sentence *"I love the drinks, esp lychee martini, and the food is also very good."* because of too long sentence. The other part has no higher attenion score than this part.

5.3 Case Study

In order to evaluate how aspect correlation is learned by RILGNet, we select one sample from test set to visualize the correlation score β from Equation (8) and attention score α from (3) about best trained RILGNet. It is worth noting that correlation score β does not indicate positive or negative correlation between different aspects, but only indicates how siginificant correlation is achieved.

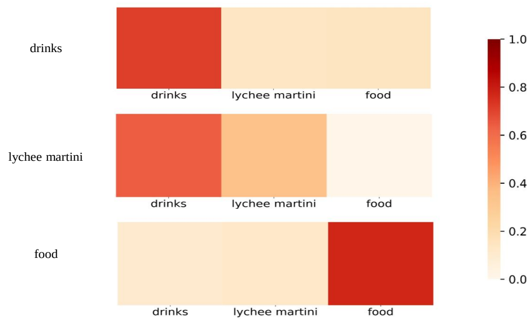


Figure 5: The correlation score between three aspects from the sentence *"I love the drinks, esp lychee martini, and the food is also very good."*

In the sentence *"I love the drinks, esp lychee martini, and the food is also very good."*, there are three aspect: *"drinks"*, *"lychee martini"* and *"food"* with positive polarities, and RILGNet makes all right prediction. *"lychee martini"* has high correlation with *"drinks"* because of conjunction *"esp"*.

As shown in Figure 4, the aspect joint representation of *"drinks"* from SAJR pays more attention to the word *"love"*, *"food"* pays more to the word *"good"* and *"lychee martini"* doesn't obviously pay high attention to optional words, which means that model doesn't know the sentiment of *"lychee martini"* only by SAJR. But model makes right prediction on *"lychee martini"*. We consider that other parts of RILGNet models aspect correlation between these three aspects by handling overlapped feature of word *"good"*.

To evaluate aspect correlation modeling, we show the correlation score β from (8). As shown in Figure 5, the top expresses correlation score between *"drinks"* and others, middle expresses *"lychee martini"* and low expresses *"food"*. Aspect *"lychee martini"* has high β with *"drinks"* in middle. From Equation (9), we can see that RILGNet knows that the feature of *"drinks"* is more important than others when model determines the polarity of *"lychee martini"*. β about *"food"* and *"drinks"* is high with itself. There is no doubt that our RILGNet can model correlation between these aspects: there is a siginificant correlation between *"drinks"* and *"lychee martini"*, and correlation between *"food"* and others is weak.

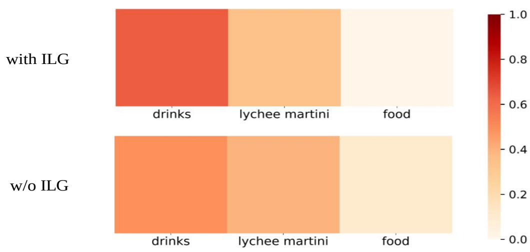


Figure 6: The correlation score of *"lychee martini"* on RILGNet and RILGNet w/o ILG.

Besides, we also show correlation scores between RILGNet and RILGNet w/o ILG of *"lychee martini"* in Figure 6. We can see that RILGNet w/o ILG gets highest score in *"drinks"*. But it also considers some information from *"lychee martini"* and *"food"*, which is not useful for sentiment classification. As a contrast, RILGNet has higher score than RILGNet w/o ILG in *"drinks"*, meaning that ILG could make the selection of feature focus more on high correlation aspects (eg,. *"drinks"*). To sum up, these results prove that RILGNet could model aspect correlation and ILG could make the selection of feature focus more on high correlation aspect with the guidance of aspect correlation.

6 Conclusion

In this paper, we propose a Recurrent Inverse Learning Guidance Network (RILGNet) to solve ABSA task by modeling aspect correlation. Firstly, in order to solve the problem of overlapped aspect feature, we design a Recurrent Meshanism to learn aspect joint representation and model aspect correlation. Secondly, RILGNet uses Inverse Learning Guidance to give feedback on how current predition reflect aspect correlation by gradient iteratively. Experimental results demonstrate the effectiveness of RILGNet for ABSA. And RILGNet also performs well in multi-aspect scenarios especially. Above all, modeling aspect correlation or label correlation will be useful for other NLP tasks.

Acknowledgement

This work was supported in part by the National Natural Science Foundation of China under Grants (62201072, 62071067, 62171057, 62101064, 62001054), in part by the Ministry of Education and China Mobile Joint Fund (MCM20200202), China Postdoctoral Science Foundation under Grant 2022M710468, Beijing University of Posts and Telecommunications-China Mobile Research Institute Joint Innovation Center.

References

Erik Boiy and Marie-Francine Moens. 2009. A machine learning approach to sentiment analysis in multilingual web texts. *Inf. Retr.*, 12(5):526–558.

Chenhua Chen, Zhiyang Teng, and Yue Zhang. 2020. Inducing target-specific latent structures for aspect sentiment classification. In *EMNLP (1)*, pages 5596–5607. Association for Computational Linguistics.

Peng Chen, Zhongqian Sun, Lidong Bing, and Wei Yang. 2017. Recurrent attention network on memory for aspect sentiment analysis. In *EMNLP*, pages 452–461. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT (1)*, pages 4171–4186. Association for Computational Linguistics.

Chunning Du, Haifeng Sun, Jingyu Wang, Qi Qi, Jianxin Liao, Tong Xu, and Ming Liu. 2019. Capsule network with interactive attention for aspect-level sentiment classification. In *EMNLP/IJCNLP (1)*, pages 5488–5497. Association for Computational Linguistics.

Xiaochen Hou, Peng Qi, Guangtao Wang, Rex Ying, Jing Huang, Xiaodong He, and Bowen Zhou. 2021. Graph ensemble learning over multiple dependency trees for aspect-level sentiment classification. In *NAACL-HLT*, pages 2884–2894. Association for Computational Linguistics.

Bin Jiang, Jing Hou, Wanyue Zhou, Chao Yang, Shihan Wang, and Liang Pang. 2020. Metnet: A mutual enhanced transformation network for aspect-based sentiment analysis. In *COLING*, pages 162–172. International Committee on Computational Linguistics.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *ICLR (Poster)*.

Svetlana Kiritchenko, Xiaodan Zhu, Colin Cherry, and Saif Mohammad. 2014. Nrc-canada-2014: Detecting aspects and sentiment in customer reviews. In *SemEval@COLING*, pages 437–442. The Association for Computer Linguistics.

Keshav Kolluru, Vaibhav Adlakha, Samarth Aggarwal, Mausam, and Soumen Chakrabarti. 2020. Openie6: Iterative grid labeling and coordination analysis for open information extraction. In *EMNLP (1)*, pages 3748–3761. Association for Computational Linguistics.

Ruifan Li, Hao Chen, Fangxiang Feng, Zhanyu Ma, Xiaojie Wang, and Eduard H. Hovy. 2021. Dual graph convolutional networks for aspect-based sentiment analysis. In *ACL/IJCNLP (1)*, pages 6319–6329. Association for Computational Linguistics.

Xin Li, Lidong Bing, Wai Lam, and Bei Shi. 2018. Transformation networks for target-oriented sentiment classification. In *ACL (1)*, pages 946–956. Association for Computational Linguistics.

Xin Li, Lidong Bing, Piji Li, and Wai Lam. 2019. A unified model for opinion target extraction and target sentiment prediction. In *AAAI*, pages 6714–6721. AAAI Press.

Peiqin Lin, Meng Yang, and Jianhuang Lai. 2019. Deep mask memory network with semantic dependency and context moment for aspect level sentiment classification. In *IJCAI*, pages 5088–5094. ijcai.org.

Bing Liu. 2012. *Sentiment Analysis and Opinion Mining*. Synthesis Lectures on Human Language Technologies. Morgan & Claypool Publishers.

Qiao Liu, Haibin Zhang, Yifu Zeng, Ziqi Huang, and Zufeng Wu. 2018. Content attention model for aspect based sentiment analysis. In *WWW*, pages 1023–1032. ACM.

Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective approaches to attention-based neural machine translation. In *EMNLP*, pages 1412–1421. The Association for Computational Linguistics.

Dehong Ma, Sujian Li, Xiaodong Zhang, and Houfeng Wang. 2017. Interactive attention networks for aspect-level sentiment classification. In *IJCAI*, pages 4068–4074. ijcai.org.

Navonil Majumder, Soujanya Poria, Alexander F. Gelbukh, Md. Shad Akhtar, Erik Cambria, and Asif Ekbal. 2018. IARM: inter-aspect relation modeling with memory networks in aspect-based sentiment analysis. In *EMNLP*, pages 3402–3411. Association for Computational Linguistics.

Qianren Mao, Jianxin Li, Senzhang Wang, Yuanning Zhang, Hao Peng, Min He, and Lihong Wang. 2019. Aspect-based sentiment classification with attentive neural turing machines. In *IJCAI*, pages 5139–5145. ijcai.org.

Maria Pontiki, Dimitris Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. 2014. Semeval-2014 task 4: Aspect based sentiment analysis. In *SemEval@COLING*, pages 27–35. The Association for Computer Linguistics.

Patrick Putzky and Max Welling. 2017. Recurrent inference machines for solving inverse problems. *CoRR*, abs/1706.04008.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training.

Xiangying Ran, Yuanyuan Pan, Wei Sun, and Chongjun Wang. 2019. Learn to select via hierarchical gate mechanism for aspect-based sentiment analysis. In *IJCAI*, pages 5160–5167. ijcai.org.

Zhan Shi, Xinchi Chen, Xipeng Qiu, and Xuanjing Huang. 2018. Toward diverse text generation with inverse reinforcement learning. In *IJCAI*, pages 4361–4367. ijcai.org.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *NIPS*, pages 3104–3112.

Yi Tay, Luu Anh Tuan, and Siu Cheung Hui. 2018. Learning to attend via word-aspect associative fusion for aspect-based sentiment analysis. In *AAAI*, pages 5956–5963. AAAI Press.

Yuanhe Tian, Guimin Chen, and Yan Song. 2021. Aspect-based sentiment analysis with type-aware graph convolutional networks and layer ensemble. In *NAACL-HLT*, pages 2910–2922. Association for Computational Linguistics.

Kai Wang, Weizhou Shen, Yunyi Yang, Xiaojun Quan, and Rui Wang. 2020. Relational graph attention network for aspect-based sentiment analysis. In *ACL*, pages 3229–3238. Association for Computational Linguistics.

Yequan Wang, Minlie Huang, Xiaoyan Zhu, and Li Zhao. 2016. Attention-based LSTM for aspect-level sentiment classification. In *EMNLP*, pages 606–615. The Association for Computational Linguistics.

Jianfei Yu and Jing Jiang. 2019. Adapting BERT for target-oriented multimodal sentiment classification. In *IJCAI*, pages 5408–5414. ijcai.org.

Lantao Yu, Jiaming Song, and Stefano Ermon. 2019. Multi-agent adversarial inverse reinforcement learning. In *ICML*, volume 97 of *Proceedings of Machine Learning Research*, pages 7194–7201. PMLR.

Lei Zhang, Shuai Wang, and Bing Liu. 2018. Deep learning for sentiment analysis: A survey. *WIREs Data Mining Knowl. Discov.*, 8(4).

Mi Zhang and Tieyun Qian. 2020. Convolution over hierarchical syntactic and lexical graphs for aspect level sentiment analysis. In *EMNLP (1)*, pages 3540–3549. Association for Computational Linguistics.

Li Zhou and Kevin Small. 2021. Inverse reinforcement learning with natural language goals. In *AAAI*, pages 11116–11124. AAAI Press.