Neurocomputing 597 (2024) 128064

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

A dual relation-encoder network for aspect sentiment triplet extraction

Tian Xia [a], Xia Sun [a,*], Yidong Yang [a], Yunfei Long [b], Richard Sutcliffe [b]

[a] School of Information Science and Technology, Northwest University, Xi'an, 710127, China
[b] School of Computer Science and Electrical Engineering, University of Essex, Colchester, CO43SQ, UK

ARTICLE INFO

Communicated by D. Cavaliere

Keywords:
Relationship emphasis
Multi-channel GCN
Criss-cross attention
Tag span shrinking
Aspect sentiment triplet extraction

ABSTRACT

Aspect sentiment triplet extraction (ASTE) combines several subtasks of aspect-based sentiment analysis, which aims to extract aspect terms, opinion terms, and their corresponding sentiment polarities in a sentence. The interaction relations between words have strong cueing information. However, previous ASTE approaches use them indiscriminately, ignoring the emphasis of relations on different subtasks. In order to fully exploit the interaction relations, we designed a multi-task learning method which uses two separate relation-encoder networks, each focusing on a different task. We call this proposed model the dual relation-encoder network (DRN). The two networks are the entity extraction relation-encoder (EER) and the entity matching relation-encoder (EMR), respectively. EER uses multi-channel graph convolutional networks to add semantic and syntactic information to the original embeddings. EMR first fuses different kinds of interaction relations, then employs criss-cross attention to obtain interaction information from other positions in the same row and column, which can provide a global view. Finally, we extract entities by sequence labeling and derive triplets with the help of span-shrunken tags. To validate the efficiency of DRN, we conducted extensive experiments on a benchmark dataset. The experimental results show that our method outperforms the strong baseline models.

1. Introduction

Opinion mining can obtain valuable information from unstructured text. Aspect-based sentiment analysis has gained much attention as an essential tool for opinion mining. Peng et al. [1] have characterized it as the task of extracting a series of triplets from an input text, which identify the aspect being referred to, the opinion term, and whether it is positive, negative or neutral. For example, in Fig. 1, for the sentence 'The sea food is fantastic but not cheap', we can find two triplets, (sea food, fantastic, POS) and (sea food, not cheap, NEG). In this case, two opinions are being expressed simultaneously about one aspect *sea food*, first that it is *fantastic* (a positive sentiment) and second that it is *not cheap* (a negative sentiment).

Most of the work is for one of the three elements or a combination of both. For individual elements, aspect term extraction (ATE) [2–4] aims to extract all aspects in a sentence; opinion term extraction (OTE) [3,5,6] aims to extract all opinions from a sentence; finally, aspect-level sentiment classification (ASC) [7–9] determines the sentiment polarity of a given aspect term. Among them, the ASC task has gained more attention. Tasks for any two elements are also available. Aspect term extraction and sentiment classification (AESC) [10–12] combines the ATE and ASC tasks by extracting aspects and their corresponding sentiment polarities. Taking the sentence in Fig. 1 as an example, and only focusing on the opinion *fantastic*, we can obtain (sea food, POS). Meanwhile, pair extraction [3,13,14] combines the ATE and OTE tasks, and extracts aspects along with the corresponding opinions. There are two pairs (sea food, fantastic) and (sea food, not cheap) in Fig. 1. However, these tasks solve only part of ABSA. None of them can obtain aspects, opinions and sentiment polarities in one operation.

To provide a complete solution, Peng et al. [1] proposed a new task, aspect sentiment triplet extraction (ASTE), which can extract aspects, opinions and sentiment polarities simultaneously. As shown in Fig. 1, we can get two triplets. Furthermore, the authors proposed a two-stage method to extract triplets. First, two sequence-labeling models were employed to extract aspects with their sentiment polarities and opinions respectively. Then the two are combined to obtain triplets. Subsequently, many new tagging schemes have been designed to extract triplets jointly [15–18]. Other researchers have converted ASTE tasks into novel forms to accomplish, such as multi-task learning, pointer networks, multi-hop quiz and generating frameworks [15,19–22].

Although previous work has achieved significant results, they ignore the interaction relations that act on entity extraction and entity matching. Complex interactions between words can lead us to extract and

* Corresponding author.
E-mail addresses: xiatian@stumail.nwu.edu.cn (T. Xia), raindy@nwu.edu.cn (X. Sun), yangyidong1@stumail.nwu.edu.cn (Y. Yang), yl20051@essex.ac.uk (Y. Long), rsutcl@essex.ac.uk (R. Sutcliffe).

https://doi.org/10.1016/j.neucom.2024.128064
Received 3 August 2023; Received in revised form 26 May 2024; Accepted 12 June 2024
Available online 14 June 2024

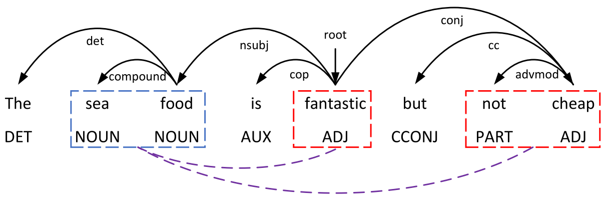


Fig. 1. Illustration of aspect sentiment triplet extraction based on an example sentence with the corresponding part-of-speech tags and syntactic dependency structures. This sentence has two triplets, (sea food, fantastic, POS) and (sea food, not cheap, NEG), respectively.

match entities, such as through syntactic and semantic relations. In entity extraction, we should pay more attention to the words themselves to determine which words are entities. Interaction relations should be taken as hints to discover multi-word entities. For example, in Fig. 1, if we recognize *food* as an entity, we can find that *sea* is also part of the entity through the compound relation, and get *sea food*.

In entity matching, we can directly utilize different types of interaction relations, which are direct indicators of relationships between entities. For example, as there is a *nsubj* relationship between *sea food* and *fantastic*, then they can probably be matched. By contrast, the relationship between *is* and *fantastic* is *cop* which should be ignored. Overall, therefore, there needs to be different emphases in utilizing interaction relationships to accomplish entity extraction and entity matching.

To fill the current research gap, we propose a dual relation-encoder network (DRN) for ASTE, which employs two relation-encoders to model the different emphases of interaction relations separately. First, we use BiLSTM/BERT to obtain sentence representations. Second, we use the two relation-encoders to model the interaction relations. The entity extraction relation-encoder (EER) treats syntactic dependency trees and multi-head self-attention scores as syntactic and semantic interaction relations, and transforms both into multi-channel adjacency matrices $\mathbf{A}^{sem}$ and $\mathbf{A}^{syn}$. EER uses multi-channel graph convolutional networks (MCGCN) from Chen et al. [23] to aggregate syntactic and semantic interactions for entity extraction. Then, entities are extracted by using sequence labeling.

The entity matching relation-encoder (EMR) concatenates $\mathbf{A}^{sem}$, $\mathbf{A}^{syn}$ and word pairs. We first perform preliminary fusion and then use criss-cross attention to obtain the relationships in other positions of the same row and column to obtain the global view for entity matching. To facilitate pairing aspects and opinions, we propose Tag Span Shrinking (TSS). Finally, a decoding algorithm is employed to obtain the triplets.

Our contributions can be summarized as follows:

1. This paper proposes a novel multi-task model based on dual relation-encoder networks. Such an architecture can refine the utilization of relations in comparison to previous work, by exploiting the interaction relation between words.
2. We use multi-channel graph convolutional networks and criss-cross attention in the two relation-encoders, respectively. Such a design can model the relationship between the two different tasks in ASTE.
3. We have conducted extensive experiments on benchmark data. The experimental results show the effectiveness of the proposed DRN model.

2. Related work

As previously discussed, aspect sentiment triplet extraction provides a complete solution for aspect-based sentiment analysis by extracting aspects, opinions, and the corresponding sentiment polarities. Peng et al. [1] first proposed this task, which employs a two-stage pipeline approach, and published a benchmark dataset[1] based on Pontiki et al. [24–26] and Fan et al. [27]. The pipeline approaches represented by Peng's work are limited by error propagation. Attempts were made to jointly extract triplets, using two approaches, by multi-task learning or by designing a new tag schema, respectively.

Since ASTE can be divided into three tasks, a multi-task learning approach can be employed. Zhang et al. [19] proposes a multi-task learning framework which adopts two sequence-labeling networks to extract aspects and opinions separately, and then match them. Subsequently, other researchers have proposed some innovative tagging schemas that can unify the three subtasks. The position-aware tagging scheme [15] introduces position information into tags and extracts triplets jointly. Wu et al. [16] propose a new grid tagging scheme, which uses a table-filling approach for ASTE tasks. Subsequently, many similar works emerged. Chen et al. [23] utilized four interactions to predict the labels of the grid tagging scheme and finally decoded the grid to obtain triplets. Liang et al. [17] designed a multi-dimensional labeling and greedy inference algorithm based [16]. Zhang et al. [18] performed an object detection task on a two-dimensional table to obtain triplets. Jiang et al. [28] adopted a dual encoder to enhance the semantic information and then employed Zhang et al.'s [18] classifier to obtain triplets.

The above methods do not directly model the interaction between spans (candidate entities); instead, a span-based approach can solve the problem. It enumerates the spans of different lengths in a sentence and explicitly models the interaction between spans. Xu et al. [29] first employ the span-based approach for ASTE. In addition, Chen et al. [30] produce triplets through two directions with the help of spans. The span-based methods generate numerous spans, which can be a burden. To reduce the burden, Li et al. [31] and Jin et al. [32] utilize syntactic reduction of spanning enumerations. But the number of spans is not reduced actually. Li et al. [33] reduced the number of generated spans by syntactic dependencies and Part-of-Speech (P-o-S) tags to reduce the computational burden.

With the development of large models, it is natural to propose methods based on them. Chen et al. [21] and Mao et al. [34] obtained triplets through a reading comprehension mechanism. Yan et al. [22] use the powerful generative model BART to generate multiple indices representing triplets. Fei et al. [35] present a non-autoregressive decoding method which models the ASTE task as an unordered triplet set prediction problem. Luo et al. [36] and Mukherjee et al. [37] employed sequence labeling to facilitate T5's [38] generation of triplets. However, Mukherjee et al. initially utilized contrastive learning for pre-training, distinguishing their approach.

The closest to our work is that of Chen et al. [23] The original idea is the same: to utilize inter-word interactions to help the model extract triplets. Conceptually, DRN aims to differentially utilize interaction relations based on ASTE subtask characteristics, while EMC-GCN aims to predict grid labels directly through various interaction relations. This causes a series of differences in the model implementation, which are described in Section 3.

3. Proposed DRN model

3.1. Outline

The detailed design of the proposed DRN model is shown in Fig. 2. In the Sentence Encoder, the input sentence is first parsed to obtain a syntactic dependency tree, as well as P-o-S tags for each word. We then use BiLSTM/BERT to get the sentence representation and splice it with the P-o-S embedding. The Entity Extraction Relation-encoder uses multi-channel graph convolutional networks to inject syntactic and

[1] https://github.com/xuuuluuu/SemEval-Triplet-data/tree/master/ASTE-Data-V1-AAAI2020

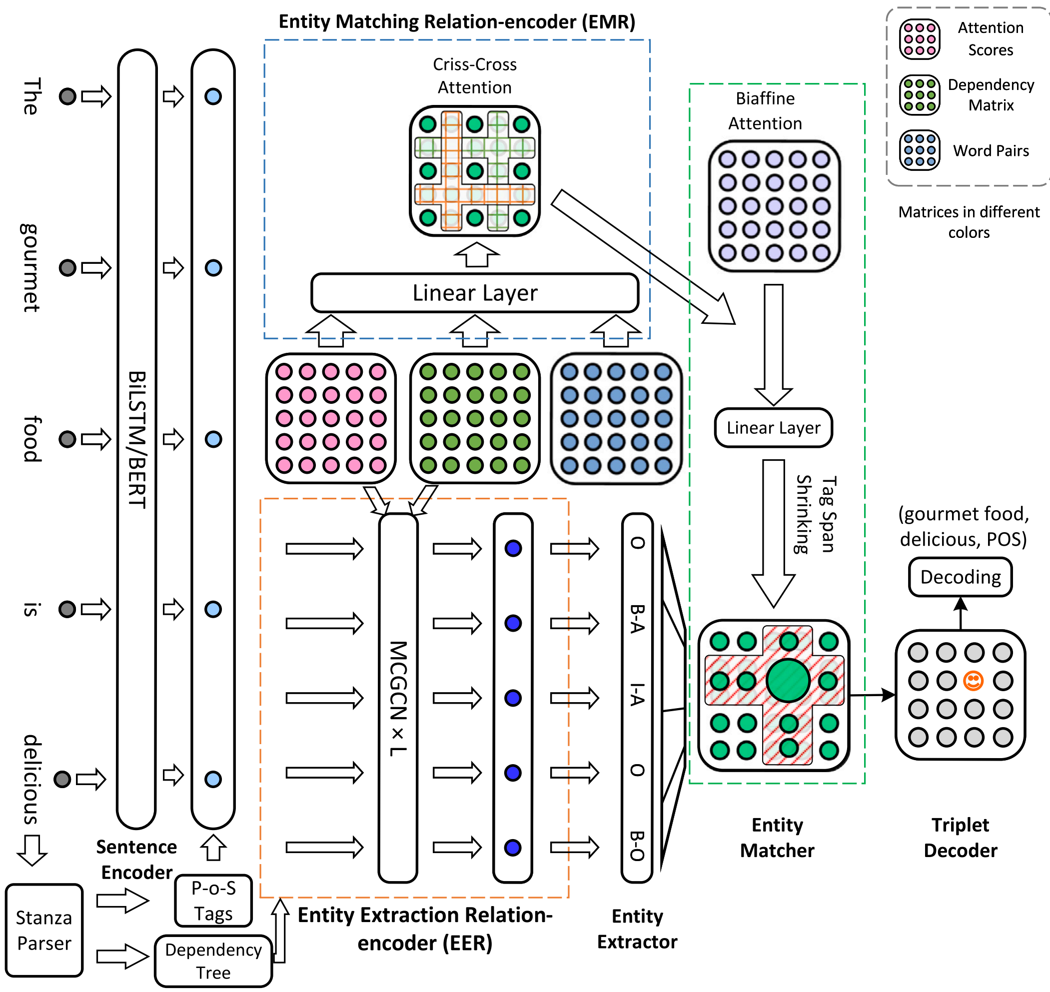


Fig. 2. The overall architecture of DRN, comprising the entity extraction relation-encoder (EER) and the entity matching relation-encoder (EMR).

semantic information into the sentence representation; then, the Entity Extractor performs sequence labeling on the sentence representation to obtain aspects and opinions. The Entity Matcher predicts the Span Shrunken Tags (SST) on the interactions fused by The Entity Matching Relation-encoder. Finally, the aspects, opinions and SST are passed through the Triplet Decoder to obtain triplets.

In the next section, we start by defining the aspect sentiment triple extraction (ASTE) task formally. Following that, we explain the other components of the model.

3.2. ASTE task definition

Given a sentence $S = \{w_1, w_2, w_3, \ldots, w_n\}$ consisting of N words, the ASTE task is to extract all of $T = \{(a, o, p)\}_{k=1}^{|T|}$ in S, where T denotes a set of triplets, and a, o and p represent the aspect, the opinion and the sentiment polarity expressed by the opinion towards the aspect. Aspect and opinion are composed of one or more words and a sentiment polarity $p \in \{POS, NEU, NEG\}$.

We adopt sequence labeling to obtain entities, which include both aspects and opinions. Entity tagging is based on BIO tags [39]. The scheme assigns $y_i^e \in C^e = \{B\text{-}A, I\text{-}A, B\text{-}O, I\text{-}O, O\}$ to each word w_i (see Table 1).

Now that we have obtained the BIO tags for each word, they can mark the position of the aspect and opinion in the sentence (Fig. 3). As shown in Fig. Fig. 5(a), if we fill the grid of the sentence with sentiment labels at locations where aspects and opinions overlap, we can get triplets through the decoding algorithm. But to facilitate the handling of multi-word entities, we propose Tag Span Shrinking (Fig. Fig. 5(b)) to match aspects and opinions (see Section 3.7.1 below).

Fig. 3. Sequence labeling to extract entity terms.

Table 1
BIO tagging scheme [39].

Tag	Meaning
B-A	Beginning of an aspect
I-A	Within an aspect
B-O	Beginning of an opinion
I-O	Within an opinion
O	Other word

3.3. Sentence encoder

The input is a sentence $S = \{w_1, w_2, \ldots, w_N\}$ expressing one or more opinions about one or more aspects (e.g. Fig. 1). We start by parsing it with Stanza[2] to obtain P-o-S tags for each token. We then utilize BiLSTM or BERT to obtain the sentence encoding.

[2] https://stanfordnlp.github.io/stanza/

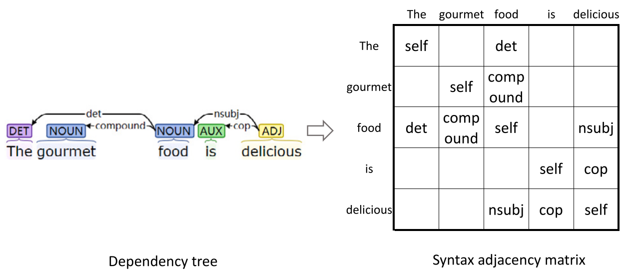


Fig. 4. Example of a syntactic adjacency tree and its conversion into a syntactic adjacency matrix.

BiLSTM: We use GloVe [40] to obtain a sequence of word embeddings $E = \{e_1, e_2, \dots, e_N\}$ for a sentence S. BiLSTM [41] is then employed to obtain the sentence representation from E:

$$\{\mathbf{v_1}, \mathbf{v_2}, \dots, \mathbf{v_N}\} = BiLSTM(E) \tag{1}$$

BERT: Another alternative available is BERT [42]. To obtain sentence encoding for each word, we add $[CLS]$ and $[SEP]$ markers to the start and end positions of the original sentence S. Then, we feed it to BERT.

$$\{\mathbf{v_1}, \mathbf{v_2}, \dots, \mathbf{v_N}\} = BERT(S) \tag{2}$$

As a fundamental language feature, P-o-S tags reveal information about how a word relates to those around it. For example, the P-o-S for each word within a compound noun is NOUN. Given a P-o-S label for a word (obtained by Stanza), we look up the corresponding trainable embedding $\mathbf{x}_i$ through a special embedding matrix, which is updated by back-propagation. We then concatenate $\mathbf{v}_i$ and $\mathbf{x}_i$ to create $\mathbf{h_i}$:

$$\mathbf{h_i} = \left[\mathbf{v_i}, \mathbf{x_i}\right] \tag{3}$$

In this way, we obtain a vector $\mathbf{h_i} \in \mathbb{R}^{d_h}$ containing a semantic embedding and a P-o-S embedding for each word. Then the sentence representation is $\mathbf{H} = \left[\mathbf{h}_1, \mathbf{h}_2, \mathbf{h}_3, \dots, \mathbf{h}_n\right]$.

3.4. Entity extraction relation-encoder

EER aims to capture semantic and syntactic interaction information for entity extraction. Multi-head attention scores [43] and syntactic dependency trees are treated as semantic and syntactic interaction relations. Then, two multi-channel graph convolutional networks (MCGCNs) [23] are adopted to capture interaction informations. One MCGCN is employed to capture semantics and the other to capture syntactic dependencies. Next, we construct two adjacency matrices $\mathbf{A}^{sem}$ and $\mathbf{A}^{syn}$ as required by each of the two MCGCNs.

The semantics adjacency matrix $\mathbf{A}^{sem} \in \mathbb{R}^{n \times n \times d_{head}}$ is computed using self-attention [43], where d_{head} is the number of self-attention heads. Self-attention calculates the attention score between each pair of words in parallel. For a word, attention scores reflect the attention paid to other words, which is more flexible in the calculation and is less sensitive to syntax. We calculate the attention score matrix as $\mathbf{A} \in \mathbb{R}^{n \times n}$:

$$\mathbf{A} = \mathrm{softmax}\left(\frac{\mathbf{H}\mathbf{W}^{Q} \times \left(\mathbf{H}\mathbf{W}^{K}\right)^{\mathrm{T}}}{\sqrt{d}}\right) \tag{4}$$

where $\mathbf{W}^{Q}$ and $\mathbf{W}^{K}$ are learnable weight matrices, while d is the dimension of the input vector feature. We obtain $\mathbf{A}^{sem}$ by concatenating the attention scores of all heads.

The syntactic adjacency matrix is defined as $\mathbf{A}^{syn} \in \mathbb{R}^{n \times n \times d_{dep}}$, where d_{dep} is the dimension of the dependency embedding. As shown in Fig. 4, we convert the dependency tree into a matrix in which each position represents a dependency relationship between words. *self* is a particular relationship related to itself. All relations are replaced with dependency embeddings to get $\mathbf{A}^{syn}$. The embeddings are looked up in a special matrix updated by back-propagation.

As mentioned above, we use MCGCNs, a form of graph convolutional network (GCN) [44] to capture the interaction. Inspired by convolutional neural networks (CNNs), a GCN can process graphs efficiently. A graph contains nodes and edges. A GCN can aggregate information about surrounding nodes through edges. Given a sentence with n words, the graph is initially represented with an adjacency matrix $\mathbf{A}$. $\mathbf{A}_{i,j}$ denotes the relationship between two words. Specifically, $\mathbf{A}_{i,j} = 1$ if the ith node is directly connected to the jth node, $\mathbf{A}_{i,j} = 0$ otherwise. However, the standard GCN cannot handle a multi-channel adjacency matrix. Hence we use a MCGCN [23]. Its adjacency matrix $A \in \mathbb{R}^{n \times n \times d}$ is no longer a scalar but a vector, where d is the number of channels. It can be formulated as:

$$\widetilde{\mathbf{H}}_i = \sigma\left(\mathbf{A}_{:,:,k}\mathbf{H}\mathbf{W}_k + \mathbf{b}_k\right) \tag{5}$$

$$\hat{\mathbf{H}} = f\left(\widetilde{\mathbf{H}}_1, \widetilde{\mathbf{H}}_2, \dots, \widetilde{\mathbf{H}}_d\right) \tag{6}$$

where $\mathbf{A}_{:,:,k}$ denotes the kth channel slice of $\mathbf{A}$, $\mathbf{W}_k$ and $\mathbf{b}_k$ are the learnable weight and bias, and σ is a RELU activation function. A pooling function $f(\cdot)$ is applied over the hidden node representations of all channels.

The above MCGCN is generic. The inputs to the first MCGCN are $\mathbf{H}$ and $\mathbf{A}^{sem}$. The inputs to the second MCGCN are $\mathbf{H}$ and $\mathbf{A}^{syn}$. MCGCNs can be stacked in multiple layers. The output of the current MCGCN is the input of the next MCGCN. The adjacency matrix of the MCGCN inputs of different layers is the same. L denotes the number of layers of the MCGCN. Thus the outputs of the last layer of GCN are $\hat{\mathbf{H}}^{sem}$, containing semantic information and $\hat{\mathbf{H}}^{syn}$, containing syntactic information.

3.5. Entity extractor

The entity extractor aims to extract entities using sequence labeling. The starting point is $\mathbf{h}_i$ (Section 3.3, Eq. (3)) which is a vector containing, for each word, the concatenation of a semantic word embedding with a P-o-S embedding. Let $\hat{\mathbf{h}}_i^{sem}$ be a vector from $\hat{\mathbf{H}}^{sem}$, and $\hat{\mathbf{h}}_i^{syn}$ be a vector from $\hat{\mathbf{H}}^{syn}$. $\hat{\mathbf{h}}_i^{sem}$ contains semantic information, and $\hat{\mathbf{h}}_i^{syn}$ contains syntactic information. The obtained representation is then fed into a linear layer, to produce a BIO tag probability distribution for each word:

$$\hat{\mathbf{y}}_i^e = \mathrm{softmax}\left(W^e\left(\left[\mathbf{h}_i, \hat{\mathbf{h}}_i^{sem}, \hat{\mathbf{h}}_i^{syn}\right]\right) + \mathbf{b}^e\right) \tag{7}$$

where $\mathbf{W}^e$ and $\mathbf{b}^e$ are the trainable weight matrix and bias. As the equation shows, output results are normalized with a $softmax$ function.

Consequently, the entity extractor's loss function can be formulated based on the cross-entropy loss:

$$\mathcal{L}_e = -\sum_{c \in C^e} \mathbf{y}_i^e(c) \log \hat{\mathbf{y}}_i^e(c) \tag{8}$$

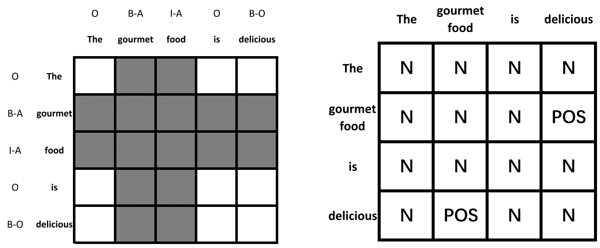
The true BIO label probability distribution for the ith word is $\mathbf{y}_i^e$, and $\mathbf{y}_i^e(c)$ is the probability that the true label is c.

After training, the Entity Extractor will assign BIO tags to input tokens, allowing entity extraction (see Fig. 3).

3.6. Entity matching relation-encoder

The main task of the EER, as described above, is to demarcate entities in the sentences, i.e. the aspects (e.g. 'gourmet food') and the opinions (e.g. 'delicious'). Next, the entity matcher relation-encoder (EMR) tries to establish the relationship between such entities (e.g. 'delicious' is an opinion about 'gourmet food').

Semantic and syntactic adjacency matrices obtained in the previous step are the critical cueing information. We concatenate $\mathbf{A}^{sem}$, $\mathbf{A}^{syn}$ and word pair embeddings together to get multi-channel interaction



(a) (b)

Fig. 5. Fig. 5(a) shows the matrix of word-word relationships, and Fig. 5(b) shows the labels of the span-span relationships after Tag Span Shrinking.

information $\left[\mathbf{A}^{sem}, \mathbf{A}^{syn}, \mathbf{S}, \mathbf{S}'\right] \in \mathbb{R}^{n\times n\times d}$, where $d = d_{head} + d_{dep} + 2 \times d_h$ (Sections Section 3.3, 3.4). $\mathbf{S}$ is created by stacking $\mathbf{H}$ n times, where n is the number of words in a sentence. The first and second dimensions of $\mathbf{S}$ are exchanged to get $\mathbf{S}'$. Then, $\mathbf{A}^{sem}$, $\mathbf{A}^{syn}$, $\mathbf{S}$, and $\mathbf{S}'$ are concatenated in the third dimension.

Next, we use a linear layer to reduce the number of channels and remove redundant information:

$$\mathbf{A} = \mathbf{W}^c \left[\mathbf{A}^{sem}, \mathbf{A}^{syn}, \mathbf{S}, \mathbf{S}'\right] + \mathbf{b}^c \tag{9}$$

where $\mathbf{W}^c$ and $\mathbf{b}^c$ are the trainable weight matrix and bias. To propagate the impact of terms in the same row or column and get the global view, we apply criss-cross attention [45] which collects contextual information in horizontal and vertical directions to enhance pixel-wise representative capability:

$$\mathbf{A}'_{\mathbf{u}} = \sum_{i=0}^{2n-1} \mathbf{W}_{i,\mathbf{u}} \boldsymbol{\Phi}_{i,\mathbf{u}} + \mathbf{A}_{\mathbf{u}} \tag{10}$$

where $\boldsymbol{\Phi}_u$ is a collection of feature vectors in $\mathbf{V}$ which are in the same row or column with position $\mathbf{u}$, and $\mathbf{V}$ is generated from $\mathbf{A}'$ with 1×1 filters. $\mathbf{A}$ undergoes criss-cross attention to get $\mathbf{A}'$, which contains a global view. $\mathbf{A}'_u$ is a feature in $\mathbf{A}' \in \mathbb{R}^{n\times n\times d'}$ at position u, and $\mathbf{W}_{i,\mathbf{u}}$ is a scalar value at channel i and position $\mathbf{u}$ in $\mathbf{W}$ that represents the attention score for $\boldsymbol{\Phi}_{i,u}$.

3.7. Entity matcher

3.7.1. Tag span shrinking

To pair aspects and opinion words, tag span shrinking uses four tags $C^p = \{POS, NEU, NEG, None\}$, which respectively represent positive, neutral, negative and no relation. In cases where an entity is a compound word, we combine the cells corresponding to its span. The resulting single cell in the grid is filled with a sentiment polarity tag. The process is shown in Fig. 5. To obtain grid tags filled with sentiment polarity labels, i.e., span-shrunken tags (SSTs), we first compress the shaded part of Fig. 5(a) to create Fig. 5(b). The triplet (gourmet food, delicious, POS) can be obtained by decoding the SST grid.

3.7.2. Biaffine attention

Biaffine attention [46] is a simple and efficient way to compute the relation probability distribution of each word pair in a sentence. So we adopt it to assist in predicting the relationship between entities. The formula is as follows:

$$\mathbf{A}^{bi} = \mathrm{Biaffine}\left(\mathrm{MLP}\left(\mathbf{H}\right), \mathrm{MLP}\left(\mathbf{H}\right)\right) \tag{11}$$

where $\mathbf{A}^{bi} \in \mathbb{R}^{n\times n\times d_{bio}}$, and $d_{bio} = |C^p|$ is the number of SST sentiment tags (i.e. POS, NEU, NEG, N).

Finally, we shrink spans (Section 3.7.1) and predict each cell in the SST grid:

$$\hat{\mathbf{Y}}^m = \mathrm{softmax}\left(\mathbf{W}^m \left(\left[\mathbf{A}', \mathbf{A}^{bi}, \mathbf{y}^{tips}, \mathbf{y}^{tips\mathrm{T}}\right]\right) + \mathbf{b}^m\right) \tag{12}$$

where $\mathbf{W}^m$ and $\mathbf{b}^m$ are the trainable weight matrix and bias, and $\mathbf{y}^{tips}$ is the span tips from the entity extractor. During the training process, $\mathbf{y}^{tips}$ is the ground truth of the entity extractor, because the model should learn the accurate span information. In order not to reveal the accurate label to the model, we make the setting $\mathbf{y}^{tips} = \mathrm{argmax}\left(\hat{\mathbf{y}}_i^e\right)$ in the prediction stage. $\hat{\mathbf{y}}^m_{sp_i,sp_j} \in \mathbb{R}^{d_{bio}}$ is the predicted distribution for an SST grid from $\hat{\mathbf{Y}}^m$. The loss function for the entity matcher can be formulated by the cross-entropy loss function between the predicted distribution and the ground truth $\mathbf{y}^m_{i,j}$ of the SST grid:

$$\mathcal{L}_m = -\sum_{sp_i\in\mathcal{T}} \sum_{sp_j\in\mathcal{T}} \sum_{c\in C^p} \mathbb{I}\left(\mathbf{y}^m_{i,j} = c\right) \log\left(\hat{\mathbf{y}}^m_{sp_i,sp_i|c}\right) \tag{13}$$

where $\mathbb{I}(\cdot)$ is the indicator function, and $\mathcal{T}$ is the set containing all spans in a sentence.

3.8. Triplet decoding

Once the BIO tags and SST grid tags are predicted, we can extract triplets by decoding. The details are shown in Algorithm 1. Firstly, we extract aspects and opinions using the BIO tags. Secondly, we enumerate the opinions to check the SST tag for each aspect. If $P(s_i, s_j) \neq None$, we add the triplet (a, o, p) to the final triplet set T. Since the tags of the upper and lower triangular parts of the grid are symmetric, we only need to check the tags of the upper triangular part of P. To ensure access to the upper triangle area of P, we need to obtain the positional index of the first word of the aspect and opinion. Then we determine whether to access $P(a, o)$ or $P(o, a)$, based on the relationship between i and j, as shown in lines 6–13 of the algorithm.

Algorithm 1 Decoding Algorithm for ASTE

Input: The tagging results B of a sentence in *BIO* mode. The SST grid, P, of the sentence. $P(s_i, s_j)$ denotes the predicted tag of the span-pair (s_i, s_j).
Output: Triplets T of the given sentence.

1: Initialize $A = \{\}$, $O = \{\}$, $T = \{\}$.
2: ▷ Extract aspects and opinions from BIO tags.
3: $A \leftarrow \mathrm{GetAspect}(B)$, $O \leftarrow \mathrm{GetAspect}(B)$
4: **for all** $a \in A$ **do**
5:   **for all** $o \in O$ **do**
6:     ▷ Get the first word position index.
7:     $i \leftarrow \mathrm{GetFirstWordIndex}(a)$
8:     $j \leftarrow \mathrm{GetFirstWordIndex}(o)$
9:     **if** $i < j$ **then**
10:       $p \leftarrow P(a, o)$
11:     **else**
12:       $p \leftarrow P(o, a)$
13:     **end if**
14:     **if** $p \neq None$ **then**
15:       $T \leftarrow T \cup (a, o, p)$
16:     **end if**
17:   **end for**
18: **end for**
19: **return** T

3.9. Loss function and training procedure

In the training process of the proposed DRN model, the loss function of joint training is defined for the entity extractor and the entity matcher:

$$\mathcal{L} = \alpha\mathcal{L}_e + (1-\alpha)\mathcal{L}_m \tag{14}$$

where $\mathcal{L}_e$ is the cross-entropy loss function of the entity extractor, $\mathcal{L}_m$ is the cross-entropy loss function of the entity matcher, and α is the adjustment coefficient.

Table 2
Statistics of ASET-Data-v2. #S, #A, #O, and #T indicate the number of sentences, aspects, opinions, and triplets, respectively.

Methods	14res			14lap			15res			16res		
	Train	Dev	Test	Train	Dev	Test	Train	Dev	Test	Train	Dev	Test
#S	1266	310	492	906	219	328	605	148	322	857	210	326
#A	2051	500	848	1280	295	463	862	213	432	1198	296	452
#O	2061	497	844	1254	302	466	935	236	460	1300	319	474
#T	2338	577	994	1460	346	543	1013	249	485	1394	339	514

4. Experiments

4.1. Experimental settings

All experiments are done on an RTX 3090. The number of MCGCN layers $L = 2$, and the dropout of MCGCN is set to 0.1. We adopted the AdamW optimizer to train the parameters. To get the effect of the word on itself, we add self-loops to both adjacency matrices $\mathbf{A}^{sem}$ and $\mathbf{A}^{syn}$. All the above are the same settings for DRN with GloVe and DRN with BERT, and the different settings are as follows:

DRN with GloVe: We use the 300-dimension domain-general embedding from GloVe.[3] The hidden dimension of BiLSTM is set to 300 with dropout rate 0.3. The learning rate is 1×10^{-3}. The training batch size is set to 8 with 100 epochs. The output of MCGCNs is 330. P-o-S tags and dependency labels are initialized using the normal distribution, and their embedding dimensions are set to 30. The biaffine attention input dimension is set to 165.

DRN with BERT: We used bert-base-uncased[4] as the encoder. The learning rate of BERT is set to 2×10^{-5}, and the dropout is set to 0.5. The learning rate of the other parts of the model is 1×10^{-4}. The training batch size is set to 16 with 100 epochs. The output of BERT has 768 dimensions, while that of the MCGCNs is 390. P-o-S tags and dependency labels are initialized using the normal distribution, and their embedding dimensions are set to 12 and 20, respectively. The biaffine attention input dimension is set to 300.

4.1.1. Datasets

We evaluate the model on ASTE-Data-V2[5] which contains four datasets. In the following sections, we refer to them as 14res, 15res, 16res, and 14lap. 14res, 15res, and 16res are datasets in the restaurant domain, whereas 14lap is in the laptop domain. They were created by Peng et al. [1] based on SemEval Challenges and refined by Xu et al. [15]. The details of the datasets are shown in Table 2.

4.1.2. Metrics

We chose the widely-used P, R and F1 scores as metrics. In addition, we consider that a triplet is correct when all elements in this triplet are consistent with the ground truth.

4.2. Baselines

We selected competitive ASTE methods as baselines to demonstrate the effectiveness of DRN. Some baselines support only GloVe or Transformer-based word embeddings, and some support both. These methods are as follows:

- **CMLA**+ [3], **RINANTE**+ [47], and **Li-unified-R**+ [48] were modified for triplet extraction by Peng et al. [1] in the original work. CMLA and RINANTE jointly extract aspect words and opinion words in comments. Li-unified-R extracts pairs of (aspect term, sentiment polarity) from comments.
- **Peng-unified** [1] employs a two-stage pipeline approach. The method obtains aspects with sentiment polarities and opinions and then determines whether the two can form triplets.
- **OTE-MTL** [19] is a multi-task learning framework which extracts aspects and opinions, and then uses biaffine attention to pair them.
- **JET (M = 6)** [15] proposes an end-to-end model with a position-aware tagging scheme which is capable of jointly extracting the triplets.
- **GTS** [16] proposes a Grid Tagging Scheme (GTS), to address the ASTE task in an end-to-end way.
- **PASTE** [20] presents a tagging-free solution for the task, which adapts an encoder–decoder architecture with a pointer network.
- **DE-OTE-BISDD** [49] proposes a method based on double embedding and a bidirectional sentiment-dependence detector.
- **DGEIAN_V2** [50] incorporates an interactive attention mechanism for ASTE, considering the contextual and syntactic representations in an iterative interaction manner.
- **BMRC** [21] transforms the ASTE task into a multi-hop quiz format. Two directions are used to extract as many triplets as possible.
- **BART-ABSA** [22] is based on the unified task formulation, and uses BART [51] to generate triplets in an end-to-end process.
- **EMC-GCN** [23] explores a variety of relations between words and proposes a novel refining strategy to conduct the ASTE task.
- **Span-DualDecoder** [52] designs two different transformer-based decoders to extract triplets for reducing cascading errors due to sequential decoding.
- **SSJE** [31] proposes a span-sharing joint extraction framework to extract aspect sentiment triplets from sentences in an end-to-end fashion.
- **BDTF** [18] performs an object detection task to obtain triplets in a grid composed of words.

4.3. Experimental results

Table 3 shows the results of DRN compared to all baseline methods using the P, R and F1 metrics.[6] First, we observe the entire table. DRN's F1 values are either the best or second best in most cases. Although the DRN has a low R value, the P value exceeds most baselines. The above suggests that DRN with dual relation-encoders is able to find the triplets more accurately but is relatively conservative. DRN is a multitask learning model with higher F1 values than strong single task learning methods such as DE-OTE-BISDD, PASTE and EMC-GCN. DRN is also ahead of Span-DualDecoder and SSJE, which are span-based methods.

DRN with GloVe: If the four datasets are considered as a whole, DRN achieves the best results on the balanced metric F1, and is only slightly lower than DGEIAN_V2 on the 14res dataset. It can also be

[3] https://nlp.stanford.edu/data/glove.840B.300d.zip
[4] https://huggingface.co/bert-base-uncased
[5] https://github.com/xuuuluuu/SemEval-Triplet-data/tree/master/ASTE-Data-V2-EMNLP2020

[6] We reproduced the baseline from the published source code to exclude hardware effects. JET adopts the data from the original paper due to source code issues.

Table 3
The experimental results (%) on the ASTE task. The best scores are highlighted in bold. Second best scores are highlighted by underlining.

Methods	14res			14lap			15res			16res		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1
With GloVe												
CMLA+[a]	39.18	47.13	42.97	30.09	36.92	33.16	34.56	39.84	37.01	41.34	42.10	41.72
RINANTE+[a]	31.42	39.38	34.95	21.72	18.66	20.07	29.88	30.06	29.97	25.68	22.30	23.87
Li-unified-R+[a]	41.04	67.35	51.00	40.56	44.28	42.34	44.72	51.39	47.82	37.33	54.51	44.31
Peng-unified[a]	43.24	63.66	51.46	37.38	50.38	42.87	48.07	57.51	52.32	46.96	64.24	54.21
OTE-MTL[c]	63.00	55.10	58.70	49.20	40.50	45.10	57.90	42.70	48.90	60.30	53.40	56.50
JET(M=6)[b]	61.50	55.13	58.14	53.03	33.89	41.35	64.37	44.33	52.50	70.94	57.00	63.21
GTS-BiLSTM[c]	66.47	56.26	60.94	56.49	38.35	45.68	71.34	44.79	55.03	68.43	54.83	60.88
PASTE[c]	64.93	62.58	63.73	55.90	47.32	51.25	54.53	54.64	54.58	59.47	65.37	62.28
DE-OTE-BISDD[b]	68.57	59.17	63.53	56.17	46.20	50.70	61.54	48.43	54.21	65.20	61.34	63.21
DGEIAN_V2[c]	70.45	61.73	**65.80**	56.15	38.82	45.90	61.84	50.99	55.89	62.13	61.40	61.77
DRN	69.35	61.30	65.08	60.72	47.92	**53.57**	62.52	51.02	**56.19**	68.08	59.51	**63.51**
With Transformer-based Embedding												
JET(M=6)[b]	70.56	55.94	62.40	55.39	47.33	51.04	64.45	51.96	57.53	70.42	58.37	63.83
GTS-BERT[c]	67.40	67.30	67.40	54.90	52.10	53.50	63.70	55.10	59.10	65.40	68.00	66.70
PASTE[c]	65.60	60.30	62.80	57.90	50.60	54.00	59.90	60.80	55.70	62.90	65.00	63.90
B-MRC[c]	70.17	64.94	67.45	62.53	53.47	57.65	58.66	55.82	57.21	66.81	67.96	67.38
BART-ABSA[c]	68.92	62.68	65.65	61.41	56.19	58.69	58.91	60.00	59.45	66.11	69.46	67.74
EMC-GCN[c]	70.94	67.54	69.20	66.58	48.98	56.44	60.40	61.65	61.02	68.52	67.45	67.98
Span-DualDecoder[c]	69.68	66.50	68.05	58.37	56.75	57.54	55.87	56.91	56.38	73.26	61.28	66.74
SSJE[c]	66.28	69.42	67.81	60.21	52.87	56.30	61.28	57.11	59.12	69.14	68.87	69.01
BDTF[c]	75.53	73.24	**74.35**	68.94	55.97	**61.74**	68.76	63.71	**66.12**	70.45	72.37	**71.40**
DRN	75.24	64.49	69.45	66.99	52.61	58.94	68.61	55.47	61.34	73.30	64.42	68.57

[a] The results are retrieved from [15].
[b] The results from the original article.
[c] We reproduce it from the open source code.

Table 4
Overall ablation results under the metric of F1(%) on four datasets released in [15].

Method	14res	14lap	15res	16res
DRN	69.45	58.94	61.34	68.57
w/o P-o-S	68.72(↓ 0.73)	57.26(↓ 1.68)	59.41(↓ 1.93)	66.99(↓ 1.58)
w/o EER	68.78(↓ 0.67)	59.22(↑ 0.28)	59.67(↓ 1.67)	66.17(↓ 2.40)
w/o EMR	66.49(↓ 1.96)	58.05(↓ 0.89)	59.08(↓ 2.26)	66.92(↓ 1.65)
w/o both	68.81(↓ 0.64)	58.06(↓ 0.88)	57.57(↓ 3.77)	65.59(↓ 2.98)
w/o BA	68.56(↓ 0.89)	58.14(↓ 0.80)	59.23(↓ 2.11)	67.25(↓ 1.32)
w/o TSS	67.06(↓ 2.39)	53.53(↓ 5.41)	56.46(↓ 4.88)	65.69(↓ 2.88)

observed that pipeline methods F1 are poorer because of error propagation, such as CMLA+, RINANTE+, Li-unified-R+, and Peng-unified.

DRN with Transformer-based embedding: The BDTF is the state-of-the-art model for the ASTE task, but the model complexity is inferior to that of the DRN (see Section 4.7). DRN achieves second best F1 values on the 14res, 14lap. Compared to DRN with GloVe, the F1 value is improved by 4.99% on average just by using BERT. It can be seen that BERT contains rich contextual information. In conclusion, the above results show that DRN can fully utilize inter-word interactions and thus improve the ASTE task results.

4.4. Ablation study

To verify the validity of each module of the model, we conducted ablation experiments. The results of the experiments are shown in Table 4, where 'w/o' means without a module. ↓ (↑) indicates a decrease (improvement) in performance when the specified module(s) are removed.

Taking a cursory look at Table 4, we can see that the F1 value decreases regardless of which module is removed, proving the effectiveness of the DRN design. Without P-o-S tags, F1 score decreases over all datasets (mean 1.48%), indicating that it helps the model identify more triplets. Label 'w/o both' indicates the removal of both EER and EMR. We can see that in the 'w/o EER', 'w/o EMR' and 'w/o both' cases, F1 scores have decreased on both the 15res and 16res datasets. In the 'w/o both' case, it drops more than both 'w/o EER' and 'w/o EMR' on F1 score. It is different for 14res; in the 'w/o both' case, it drops less than either 'w/o EER' or 'w/o EMR'. We speculate that the model may have overfitted because the 14res dataset has the largest amount of data among the four datasets. In the case of 'w/o EER', the F1 score shows a 0.28% improvement on 14lap. We can observe that all baselines have the worst results on 14lap, which has more complex data. The interaction information is actually considered as a 'hint' in the extracted triplets, but it may disturb the model in 14lap instead. 'w/o BA' indicates that biaffine attention does not assist in predicting SST. It can be seen that the F1 value has decreased a bit in 14res, 14lap and 16res datasets, which indicates that biaffine attention has a cueing effect when matching entities. However, in the 15res dataset, the F1 value decreases more. Table 2 shows that the 15res data is the least and any change in the model has a more significant impact on it, so biaffine attention has a greater cueing effect on the 15res dataset. Without using TSS, meaning that the span does not shrink, we can see that the F1 scores all significantly decrease (the mean is 3.89%) across the four datasets. We think that, without TSS, the model will have more labels to fit, making it harder to train and prone to inconsistent sentiment polarity in prediction. TSS thus reduces the burden on the model.

4.5. Effect of α

We evaluate the effect of the coefficient α on the results of the ASTE experiments. α adjusts the weights of the entity extractor and entity matcher loss functions. The variation curves are shown in Fig. 6, and we can see that the model achieves the best results on all four data sets when the entity extractor takes larger weights. The best F1 scores are obtained on 14lap and 15res when $\alpha = 0.6$. When $\alpha = 0.7$, the F1 score is highest on the 14res dataset, while the best results are obtained on the 16res dataset when $\alpha = 0.8$. From the above results, it can be seen that the entity extraction task has a greater impact on the ASTE task. If the entities in the sentences cannot be extracted, the subsequent matching task is powerless.

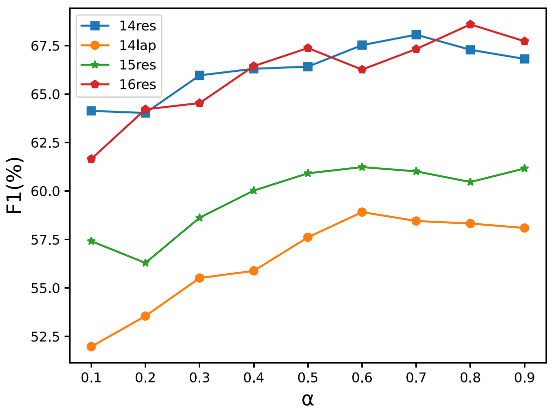


Fig. 6. Effect of the adjustment coefficient α.

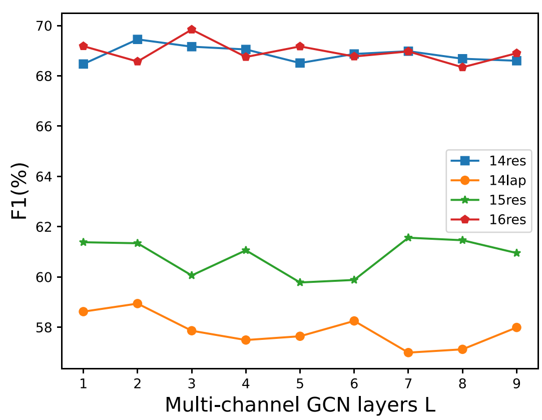


Fig. 7. Effect of the Number of Multi-Channel GCN Layers.

Table 5
Experimental results on model complexity for strong baselines.

Methods	Params (M)	MACs (G)
EMC-GCN	87.34	4.35
SSJE	86.84	**3.92**
BDTF	101.20	43.09
Ours	**85.89**	5.20

4.6. Effect of L

As shown in Fig. 7, we evaluated the effect of the number of MCGCN layers L on the experimental results. When $L = 2$, the best F1 score is obtained on res14 and lap14. Then, when $L = 3$, the F1 score is best on 16res. The difference for 15res is that the best result is obtained when $L = 7$. It may be because the interactions in the 15res dataset are complex, so it is necessary to stack multiple layers of MCGCN to mine them.

4.7. Model complexity

The theoretical time complexity of our model is $O(N^2)$. To accurately depict the model's complexity, we evaluated model parameters (Params) and multiply-accumulate operations (MACs) for several robust baselines, including BDTF [18], at a sentence length of 40. The experimental results are shown in Table 5. Among them, DRN has the smallest Params; MACs are comparable to EMC-GCN and SSJE. BDTF, a state-of-the-art ASTE model, creatively treats the ASTE task as object detection. However, BDTF has the most parameters and has eight times more MACs than DRN. Additionally, as the sentence length increases, the increase in MACs for BDTF surpasses that of DRN. Consequently, DRN offers a relatively balanced model complexity.

4.8. Case study

We compare the DRN and the two strong baseline BMRC and BART-ABSA predictions on three samples from the dataset. As shown in Table 6, DRN correctly extracted all three triplets, while BMRC and BART-ABSA had prediction errors and omissions. The first example, S1, is simple, with BMRC, BART-ABSA and DRN all correctly extracting triplets. S2 contains compound words, and aspects and opinions are one-to-many relationships. BMRC correctly extracts triplets where aspects and opinions are close together. BART-ABSA obtains correct opinions but does not work with compound words. BMRC employs a reading comprehension mechanism, and we speculate that feeding long texts of samples and questions into the model weakens the model's ability to capture long-distance dependencies. BART-ABSA uses the powerful generative model BART to generate triplets with randomness directly. The most obvious in S3 is that aspects and opinions are many-to-one relationships. BMRC predictions have omissions, and BART-ABSA has prediction errors and omissions. Both BMRC and BART-ABSA lack interaction relation injection, making it difficult to handle complex correspondences. Conversely, DRN can handle compound words and complex correspondences between aspects and opinions.

5. Conclusions

In this paper, we propose a dual relation-encoder framework, DRN, for extracting aspect sentiment triplets from review texts. DRN differs from previous approaches by modeling the emphasis of interaction relations on different subtasks. To accomplish this, we design two relation-encoders, the entity extraction relation-encoder (EER) and the entity matching relation-encoder (EMR). EER uses MCGCNs to capture multi-channel interaction relations between words for entity extraction. EMR operates on the relationship matrix and explicitly models the relationships themselves. EMR initially fuses the relations and then uses criss-cross attention to obtain a global view for pairing entities. Finally, we design a decoding algorithm to get triplets. We conduct experiments on four widely-used datasets to verify the effectiveness of DRN. The experimental results demonstrate that our method exceeds the competitive baselines. However, the ASTE task is still not fully solved, and our future work will focus on how to make the best use of interactions.

Limitations

Despite the DRN model's competitive achievements, it still possesses the following limitations.

- DRN relies too much on entity extraction. Matching aspects and opinions is dependent on the accuracy of those extracted aspects and opinions. Attempting to match entities that have been incorrectly extracted is futile.
- TSS has difficulty overcoming the effects of long-distance dependencies when there are multiple compound word entities, or entities with too many words, in a sentence.
- DRN has lower recall than precision. This is because DRN minimizes false predictions, which also reduces correct predictions.

We believe addressing the above limitations can improve the model's performance without losing its original strengths.

Table 6
Case study of ASTE. Incorrect results are marked with ✗.

Sentences	Ground truth	BMRC	BART-ABSA	DRN (Ours)
S1: Great food but the service was dreadful!	{(service, dreadful, NEG), (food, Great, POS)}	{(service, dreadful, NEG), (food, Great, POS)}	{(service, dreadful, NEG), (food, Great, POS)}	{(service, dreadful, NEG), (food, Great, POS)}
S2: I highly recommend the grand marnier shrimp, it's insanely good.	{(grand marnier shrimp, good, POS), (grand marnier shrimp, recommend, POS)}	{✗, (grand marnier shrimp, recommend, POS)}	{(marnier shrimp✗, good, POS), (marnier shrimp✗, recommend, POS)}	{(grand marnier shrimp, good, POS), (grand marnier shrimp, recommend, POS)}
S3: You must try Odessa stew or Rabbit stew; salads -all good ; and kompot is soo refreshing during...	{(kompot, refreshing, POS), (salads, good, POS), (Rabbit stew, good, POS), (Odessa stew, good, POS) }	{(kompot, refreshing, POS), (salads, good, POS), ✗, ✗}	{(kompot, refreshing, POS), (salads, good, POS), (stew, good, POS)✗, ✗}	{(kompot, refreshing, POS), (salads, good, POS), (Rabbit stew, good, POS), (Odessa stew, good, POS) }

CRediT authorship contribution statement

Tian Xia: Conceptualization, Data curation, Investigation, Methodology, Software, Validation, Visualization, Writing – original draft, Writing – review & editing. **Xia Sun:** Funding acquisition, Project administration, Resources, Supervision, Writing – original draft, Writing – review & editing. **Yidong Yang:** Software. **Yunfei Long:** Resources, Writing – original draft, Writing – review & editing, Supervision. **Richard Sutcliffe:** Supervision, Writing – original draft, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgments

This work was supported by the National Natural Science Foundation of China under Grant No. 61877050, and by the Key R & D plan of Xianyang of China under Grant No. 2021ZDYF-GY-0033.

References

[1] H. Peng, L. Xu, L. Bing, F. Huang, W. Lu, L. Si, Knowing What, How and Why: a near complete solution for aspect-based sentiment analysis, in: Proceedings of the AAAI Conference on Artificial Intelligence, 34, 2020, pp. 8600–8607, http://dx.doi.org/10.1609/aaai.v34i05.6383.

[2] P. Liu, S. Joty, H. Meng, Fine-grained opinion mining with recurrent neural networks and word embeddings, in: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Lisbon, Portugal, 2015, pp. 1433–1443, http://dx.doi.org/10.18653/v1/D15-1168.

[3] W. Wang, S.J. Pan, D. Dahlmeier, X. Xiao, Coupled multi-layer attentions for co-extraction of aspect and opinion terms, Proc. AAAI Conf. Artif. Intell. 31 (1) (2017) http://dx.doi.org/10.1609/aaai.v31i1.10974.

[4] D. Ma, S. Li, F. Wu, X. Xie, H. Wang, Exploring sequence-to-sequence learning in aspect term extraction, in: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Florence, Italy, 2019, pp. 3538–3547, http://dx.doi.org/10.18653/v1/P19-1344.

[5] J. Yu, J. Jiang, R. Xia, Global inference for aspect and opinion terms co-extraction based on multi-task neural networks, IEEE/ACM Trans. Audio Speech Lang. Process. 27 (1) (2019) 168–177, http://dx.doi.org/10.1109/TASLP.2018.2875170.

[6] M. Wu, W. Wang, S.J. Pan, Deep Weighted MaxSAT for Aspect-based Opinion Extraction, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), Association for Computational Linguistics, Online, 2020, pp. 5618–5628, http://dx.doi.org/10.18653/v1/2020.emnlp-main.453.

[7] C. Chen, Z. Teng, Y. Zhang, Inducing target-specific latent structures for aspect sentiment classification, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP, Association for Computational Linguistics, Online, 2020, pp. 5596–5607, http://dx.doi.org/10.18653/v1/2020.emnlp-main.451.

[8] Q. Lu, X. Sun, R. Sutcliffe, Y. Xing, H. Zhang, Sentiment interaction and multi-graph perception with graph convolutional networks for aspect-based sentiment analysis, Knowl.-Based Syst. 256 (2022) 109840, http://dx.doi.org/10.1016/j.knosys.2022.109840.

[9] Y. Yang, X. Sun, Q. Lu, R. Sutcliffe, J. Feng, A sentiment and syntactic-aware graph convolutional network for aspect-level sentiment classification, in: ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP, IEEE, 2023, pp. 1–5.

[10] R. He, W.S. Lee, H.T. Ng, D. Dahlmeier, An interactive multi-task learning network for end-to-end aspect-based sentiment analysis, in: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Florence, Italy, 2019, pp. 504–515, http://dx.doi.org/10.18653/v1/P19-1048.

[11] M. Zhang, Y. Zhang, D.T. Vo, Neural networks for open domain targeted sentiment, in: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Lisbon, Portugal, 2015, pp. 612–621, http://dx.doi.org/10.18653/v1/D15-1073.

[12] G. Yu, J. Li, L. Luo, Y. Meng, X. Ao, Q. He, Self Question-answering: Aspect-based Sentiment Analysis by Role Flipped Machine Reading Comprehension, in: Findings of the Association for Computational Linguistics: EMNLP 2021, Association for Computational Linguistics, Punta Cana, Dominican Republic, 2021, pp. 1331–1342, http://dx.doi.org/10.18653/v1/2021.findings-emnlp.115.

[13] S. Chen, J. Liu, Y. Wang, W. Zhang, Z. Chi, Synchronous double-channel recurrent network for aspect-opinion pair extraction, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Online, 2020, pp. 6515–6524, http://dx.doi.org/10.18653/v1/2020.acl-main.582.

[14] H. Zhao, L. Huang, R. Zhang, Q. Lu, h. xue, SpanMlt: a span-based multi-task learning framework for pair-wise aspect and opinion terms extraction, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Online, 2020, pp. 3239–3248, http://dx.doi.org/10.18653/v1/2020.acl-main.296.

[15] L. Xu, H. Li, W. Lu, L. Bing, Position-aware tagging for aspect sentiment triplet extraction, 2021, arXiv:2010.02609.

[16] Z. Wu, C. Ying, F. Zhao, Z. Fan, X. Dai, R. Xia, Grid tagging scheme for aspect-oriented fine-grained opinion extraction, 2020, arXiv:2010.04640.

[17] S. Liang, W. Wei, X.-L. Mao, Y. Fu, R. Fang, D. Chen, STAGE: span tagging and greedy inference scheme for aspect sentiment triplet extraction, Proc. AAAI Conf. Artif. Intell. 37 (11) (2023) 13174–13182, http://dx.doi.org/10.1609/aaai.v37i11.26547.

[18] Y. Zhang, Y. Yang, Y. Li, B. Liang, S. Chen, Y. Dang, M. Yang, R. Xu, Boundary-driven table-filling for aspect sentiment triplet extraction, in: Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, 2022, pp. 6485–6498.

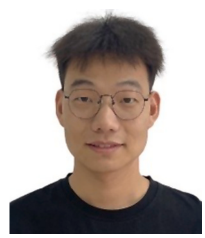
[19] C. Zhang, Q. Li, D. Song, B. Wang, A multi-task learning framework for opinion triplet extraction, in: Findings of the Association for Computational Linguistics: EMNLP2020, Association for Computational Linguistics, Online, 2020, pp. 819–828, http://dx.doi.org/10.18653/v1/2020.findings-emnlp.72.

[20] R. Mukherjee, T. Nayak, Y. Butala, S. Bhattacharya, P. Goyal, PASTE: a tagging-free decoding framework using pointer networks for aspect sentiment triplet extraction, in: Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Online and Punta Cana, Dominican Republic, 2021, pp. 9279–9291, http://dx.doi.org/10.18653/v1/2021.emnlp-main.731.

[21] S. Chen, Y. Wang, J. Liu, Y. Wang, Bidirectional machine reading comprehension for aspect sentiment triplet extraction, Proc. AAAI Conf. Artif. Intell. 35 (14) (2021) 12666–12674, http://dx.doi.org/10.1609/aaai.v35i14.17500.

[22] H. Yan, J. Dai, T. Ji, X. Qiu, Z. Zhang, A unified generative framework for aspect-based sentiment analysis, in: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), Association for Computational Linguistics, Online, 2021, pp. 2416–2429, http://dx.doi.org/10.18653/v1/2021.acl-long.188.

[23] H. Chen, Z. Zhai, F. Feng, R. Li, X. Wang, Enhanced multi-channel graph convolutional network for aspect sentiment triplet extraction, in: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Dublin, Ireland, 2022, pp. 2974–2985, http://dx.doi.org/10.18653/v1/2022.acl-long.212.
[24] M. Pontiki, D. Galanis, H. Papageorgiou, I. Androutsopoulos, S. Manandhar, M. AL-Smadi, M. Al-Ayyoub, Y. Zhao, B. Qin, O. De Clercq, et al., Semeval-2016 task 5: Aspect based sentiment analysis, in: ProWorkshop on Semantic Evaluation, SemEval-2016, Association for Computational Linguistics, 2016, pp. 19–30.
[25] M. Pontiki, D. Galanis, H. Papageorgiou, S. Manandhar, I. Androutsopoulos, Semeval-2015 task 12: Aspect based sentiment analysis, in: Proceedings of the 9th International Workshop on Semantic Evaluation, SemEval 2015, 2015, pp. 486–495.
[26] M. Pontiki, D. Galanis, J. Pavlopoulos, H. Papageorgiou, I. Androutsopoulos, S. Manandhar, SemEval-2014 task 4: Aspect based sentiment analysis, in: Proceedings of the 8th International Workshop on Semantic Evaluation, SemEVal 2014, Association for Computational Linguistics, Dublin, Ireland, 2014, pp. 27–35, http://dx.doi.org/10.3115/v1/S14-2004, URL https://aclanthology.org/S14-2004.
[27] Z. Fan, Z. Wu, X.-Y. Dai, S. Huang, J. Chen, Target-oriented opinion words extraction with target-fused neural sequence labeling, in: Proceedings of the 2019 Conference of the North, Association for Computational Linguistics, Minneapolis, Minnesota, 2019, pp. 2509–2518, http://dx.doi.org/10.18653/v1/N19-1259.
[28] B. Jiang, S. Liang, P. Liu, K. Dong, H. Li, A semantically enhanced dual encoder for aspect sentiment triplet extraction, Neurocomputing 562 (2023) 126917, http://dx.doi.org/10.1016/j.neucom.2023.126917.
[29] L. Xu, Y.K. Chia, L. Bing, Learning span-level interactions for aspect sentiment triplet extraction, 2021, arXiv:2107.12214.
[30] Y. Chen, K. Chen, X. Sun, Z. Zhang, A span-level bidirectional network for aspect sentiment triplet extraction, 2022, arXiv:2204.12674.
[31] Y. Li, Y. Lin, Y. Lin, L. Chang, H. Zhang, A span-sharing joint extraction framework for harvesting aspect sentiment triplets, Knowl.-Based Syst. 242 (2022) 108366, http://dx.doi.org/10.1016/j.knosys.2022.108366.
[32] Z. Jin, M. Tao, X. Wu, H. Zhang, Span-based dependency-enhanced graph convolutional network for aspect sentiment triplet extraction, Neurocomputing 564 (2024) 126966, http://dx.doi.org/10.1016/j.neucom.2023.126966.
[33] P. Li, P. Li, K. Zhang, Dual-channel span for aspect sentiment triplet extraction, in: Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Singapore, 2023, pp. 248–261, http://dx.doi.org/10.18653/v1/2023.emnlp-main.17.
[34] Y. Mao, Y. Shen, C. Yu, L. Cai, A joint training dual-MRC framework for aspect based sentiment analysis, Proc. AAAI Conf. Artif. Intell. 35 (15) (2021) 13543–13551, http://dx.doi.org/10.1609/aaai.v35i15.17597.
[35] H. Fei, Y. Ren, Y. Zhang, D. Ji, Nonautoregressive encoder-decoder neural framework for end-to-end aspect-based sentiment triplet extraction, IEEE Trans. Neural Netw. Learn. Syst. (2021) 1–13, http://dx.doi.org/10.1109/TNNLS.2021.3129483.
[36] L. Xianlong, M. Yang, Y. Wang, Tagging-assisted generation model with encoder and decoder supervision for aspect sentiment triplet extraction, in: Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Singapore, 2023, pp. 2078–2093, http://dx.doi.org/10.18653/v1/2023.emnlp-main.129.
[37] R. Mukherjee, N. Kannen, S. Pandey, P. Goyal, CONTRASTE: supervised contrastive pre-training with aspect-based prompts for aspect sentiment triplet extraction, in: Findings of the Association for Computational Linguistics, EMNLP 2023, Association for Computational Linguistics, Singapore, 2023, pp. 12065–12080, http://dx.doi.org/10.18653/v1/2023.findings-emnlp.807.
[38] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, P.J. Liu, Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer.
[39] L.A. Ramshaw, M.P. Marcus, Text chunking using transformation-based learning, 1995, arXiv preprint cmp-lg/9505040.
[40] J. Pennington, R. Socher, C. Manning, Glove: global vectors for word representation, in: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP, Association for Computational Linguistics, Doha, Qatar, 2014, pp. 1532–1543, http://dx.doi.org/10.3115/v1/D14-1162.
[41] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural Comput. 9 (8) (1997) 1735–1780.
[42] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, 2018, arXiv preprint arXiv:1810.04805.
[43] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, Adv. Neural Inf. Process. Syst. 30 (2017).
[44] T.N. Kipf, M. Welling, Semi-supervised classification with graph convolutional networks, 2017, arXiv:1609.02907.
[45] Z. Huang, X. Wang, Y. Wei, L. Huang, H. Shi, W. Liu, T.S. Huang, CCNet: criss-cross attention for semantic segmentation, 2020, arXiv:1811.11721.
[46] T. Dozat, C.D. Manning, Deep biaffine attention for neural dependency parsing, 2017, arXiv:1611.01734.
[47] H. Dai, Y. Song, Neural aspect and opinion term extraction with mined rules as weak supervision, in: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Florence, Italy, 2019, pp. 5268–5277, http://dx.doi.org/10.18653/v1/P19-1520.
[48] X. Li, L. Bing, P. Li, W. Lam, A unified model for opinion target extraction and target sentiment prediction, Proc. AAAI Conf. Artif. Intell. 33 (01) (2019) 6714–6721, http://dx.doi.org/10.1609/aaai.v33i01.33016714.
[49] D. Dai, T. Chen, S. Xia, G. Wang, Z. Chen, Double embedding and bidirectional sentiment dependence detector for aspect sentiment triplet extraction, Knowl.-Based Syst. 253 (2022) 109506, http://dx.doi.org/10.1016/j.knosys.2022.109506.
[50] L. Shi, D. Han, J. Han, B. Qiao, G. Wu, Dependency graph enhanced interactive attention network for aspect sentiment triplet extraction, Neurocomputing 507 (2022) 315–324, http://dx.doi.org/10.1016/j.neucom.2022.07.067.
[51] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, L. Zettlemoyer, Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension, 2019, arXiv preprint arXiv:1910.13461.
[52] Y. Chen, Z. Zhang, G. Zhou, X. Sun, K. Chen, Span-based dual-decoder framework for aspect sentiment triplet extraction, Neurocomputing 492 (2022) 211–221, http://dx.doi.org/10.1016/j.neucom.2022.04.022.



Tian Xia received the bachelor's degree from Northwest University, Xi'an, China, in 2021. He is currently a Master candidate in the School of Information Science and Technology, Northwest University, Xi'an, China. His research interests include natural language processing and sentiment analysis.

Xia Sun received the Ph.D. degree from Xi'an Jiaotong University, Xi'an, China, in 2006. She is currently a Professor with the School of Information Science and Technology, Northwest University, Xi'an. She has coauthored more than 60 articles and is an editor or a coeditor of four books. Her current research interests include natural language processing and intelligent education. Prof. Sun has reviewed many journals, including IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS, Pattern Recognition, and Chinese Journal of Electronics.

Yidong Yang received the B.S. degree in computer science and technology from Northwest University, Xi'an, China, in 2022, where he is currently pursuing the M.S. degree. His research interests include natural language processing and intelligent education for programming.

Yunfei Long (Member, IEEE) received the Ph.D. degree from the Department of Computing, The Hong Kong Polytechnic University, Hong Kong, in 2019. From 2019 to March 2020, he was a Research Fellow with the School of Medical Science, University of Nottingham, Nottingham, U.K. Since April 2020, he has been a Lecturer (Assistant Professor) with the School of Computer Science, University of Essex, Colchester, U.K. His research interests include innovative natural language processing (NLP) techniques; explainable artificial intelligence (AI) models for analyzing healthcare data, the utilization of lexical semantics in affective analysis, and other NLP applications; and the applications of NLP in media, legal, and crime studies.

Richard Sutcliffe received the Ph.D. degree from the University of Essex, Colchester, U.K., in 1989. He is a Senior Lecturer in Natural Language Processing at University of Essex, UK, and a former Associate Professor at Northwest University, Xi'an, China. His current research interests include natural language processing, information retrieval, and music information retrieval. He has coauthored 101 articles and coedited 3 books and 10 conference proceedings. Dr. Sutcliffe has reviewed for many journals and conferences.