Automatic Construction of a Chinese Review Dataset for Aspect Sentiment Triplet Extraction via Iterative Weak Supervision

Chia-Wen Lu, Ching-Wen Yang, Wei-Yun Ma*
Academia Sinica, National Cheng Kung University, Academia Sinica
qwe9887476@gamil.com, P76114511@gs.ncku.edu.tw, ma@iis.sinica.edu.tw

Abstract

Aspect Sentiment Triplet Extraction (ASTE), introduced in 2020, is a task that involves the extraction of three key elements: target aspects, descriptive opinion spans, and their corresponding sentiment polarity. This process, however, faces a significant hurdle, particularly when applied to Chinese languages, due to the lack of sufficient datasets for model training, largely attributable to the arduous manual labeling process. To address this issue, we present an innovative framework that facilitates the automatic construction of ASTE via Iterative Weak Supervision, negating the need for manual labeling, aided by a discriminator to weed out subpar samples. The objective is to successively improve the quality of this raw data and generate supplementary data. The effectiveness of our approach is underscored by our results, which include the creation of a substantial Chinese review dataset. This dataset encompasses over 60,000 Google restaurant reviews in Chinese and features more than 200,000 extracted triplets. Moreover, we have also established a robust baseline model by leveraging a novel method of weak supervision. Both our dataset and model are openly accessible to the public.

Keywords: Aspect Sentiment Triplet Extraction, Sentiment Analysis, Weakly-Supervised

1. Introduction

People often resort to online reviews when deciding on restaurants, using the ratings as a gauge for quality. Yet, extracting nuanced evaluations of various restaurant aspects from these reviews can prove challenging. Google Maps, while offering tags to sort reviews, does not provide in-depth assessment on specific elements such as the dishes, ambiance, or customer service. To address this, our goal is to construct such a dataset based on reviews in Google Maps and a baseline model that can effectively perform Aspect Sentiment Triplet Extraction (ASTE) tasks. Such effort would enable us to dissect popular dishes and perform comprehensive evaluations of a restaurant across a multitude of facets. As depicted in Figure 1, the output from our baseline will include aspect identification, opinion expressions, and sentiment polarity for each review. This approach allows for a more precise and detailed analysis of restaurant reviews.

While there have been numerous studies focused on Aspect Sentiment Triplet Extraction (ASTE) since its introduction in 2020 (Peng et al., 2020), one significant hurdle continues to pose challenges: the acquisition of large, well-labeled training datasets. This issue is particularly prominent in the case of Chinese language datasets, which are substantially less abundant than their English counterparts. This paucity of Chinese datasets hinders advancements in the field of Chinese sentiment analysis research. To address this issue, we present an innovative framework that facilitates the automatic construction of ASTE via Iterative Weak Supervision (IWS), negating the need for manual labeling. Our method initiates with a rule-based model producing a large amount of raw data. This data, though initially of variable quality, serves as training input for an iteratively refining transformer model that adopts an encoder-decoder architecture, aided by a discriminator to weed out subpar samples. The objective is to successively improve the quality of this raw data and generate supplementary data.

The efficacy of our method is clearly demonstrated by our achievements, one of which includes the construction of a considerable dataset of Chinese restaurant reviews. This dataset is composed of more than 60,000 reviews sourced from Google Maps, and it features over 200,000 extracted triplets. Additionally, we've set a strong baseline model using a unique approach of weak supervision. Both the dataset and the model are made available to the public[1], reflecting our commitment to open access and collaborative advancement in this field.

2. Related Work

2.1. Overview

Aspect Sentiment Triplet Extraction (ASTE) is first introduced by Peng et al. (2020) as a subtask of ABSA (Aspect-Based Sentiment Analysis) (Zhang et al., 2022), which aims to extract (a, o, p)(a: $aspect$,

*Corresponding author

[1] https://github.com/chiawen0104/chn_review_aste.git

LREC-COLING 2024, pages 1871–1882
20-25 May, 2024.

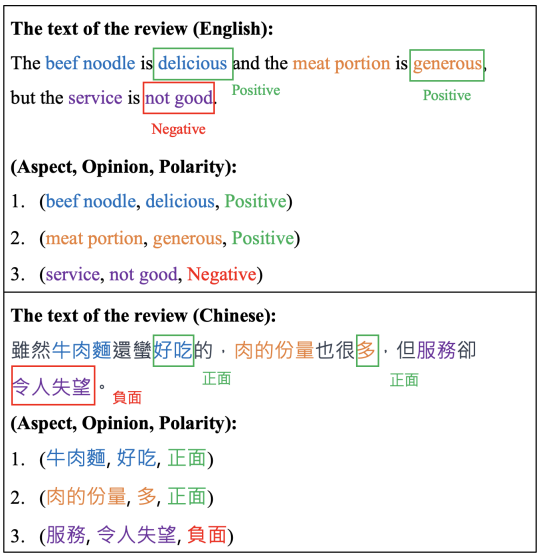

Figure 1: An example of ASTE with both English and Chinese versions. The extraction results of aspect sentiment triplets (AST) from a restaurant review are presented. Each triplet includes three components: an aspect, an opinion, and its corresponding polarity.

o: $opinion$, p: $polarity$) triplets from a given review. Peng et al. utilized the extended SemEval dataset annotated with opinion terms by Fan et al. (2019) as the experiment dataset, and proposed a two-stage framework: the first stages mines candidate aspects with sentiment polarities and candidate opinion terms, and the second stage pairs them up to form valid triplets. Compared to single ABSA tasks proposed such as Aspect Category Detection or Aspect Term Extraction, a solution to ASTE is more suited for real-world application, and has been gaining attention in recent years.

2.2. Modeling Paradigm

Past works model ASTE into different paradigms. Peng et al. built a two-stage pipeline comprising of 2 sequence labeling and 1 sequence classification submodules; Xu et al. (2020) defined complicated token annotation and condense the workflow into 1 sequence labeling; Chen et al. (2021) employed a machine-reading comprehension paradigm to first inquire the machine about the location of the aspect terms, and then inquire about the opinion term and corresponding polarity; Zhang et al. (2021) adopted a seq2seq framework, which can do the task in one pass, and finally compare the performance between using annotation-style and extraction-style output format.

2.3. ASTE Datasets

Many benchmark datasets for ASTE are originally derived from the SemEval (2014, 2015, 2016) benchmark datasets in the laptop and restaurant domains. However, these datasets do not include annotations for opinion terms until Xu et al. performed additional annotation and made the dataset publicly available. Subsequently, these datasets have gained widespread usage in subsequent studies on ASTE.

When it comes to research on Chinese sentiment analysis, Bu et al. (2021) is one of the first Chinese dataset with 46,730 genuine user reviews, was constructed from reviews across many restaurants from the Diaping app. However, this dataset falls short of fulfilling the requirements of ASTE tasks as it lacks extracted aspect and polarity information. Instead, it only provides a predefined set of categories and corresponding ratings. Li et al. (2023) introduced a novel task "dialogue ABSA" and released the initial dialogue ABSA dataset, DiaASQ, focusing on the cellphone domain on Weibo platform. In addition to aspect (a), opinion term (o), and polarity (p), DiaASQ further annotates the mentioned targets (t), resulting in quadruples of (t, a, o, p). All conversation texts were labeled by a team of crowd-workers who underwent pre-training using the SemEval ABSA (Pontiki et al., 2014) annotation guideline. A key challenge faced by this study, similar to the SemEval datasets, was the intensive manual labeling process, leading to a small size of the dataset - only a total of 7,452 utterances, and 5,742 are sentiment quadruples. This challenge has sparked our motivation to explore automatic annotation methods for ASTE tasks.

3. Methodology

3.1. Iterative Weak Supervision

Our proposed framework of IWS, depicted in Figure 2, initially employs a rule-based system founded on pipeline methodologies to generate a significant volume of labeled data. This data, despite being somewhat noisy, serves as the training foundation for the enhancement of our encoder-decoder transformer model. As a result of self-training, our models are able to not only execute ASTE tasks, but also discern the validity of ASTs present within the training data. This approach allows for continuous refinement and improvement in the quality of the model's performance.

3.2. Rule-Based System

Figure 3 illustrates the operation process of the proposed rule-based system. Firstly, we use the Chinese Natural Language Processing tool, CKIP

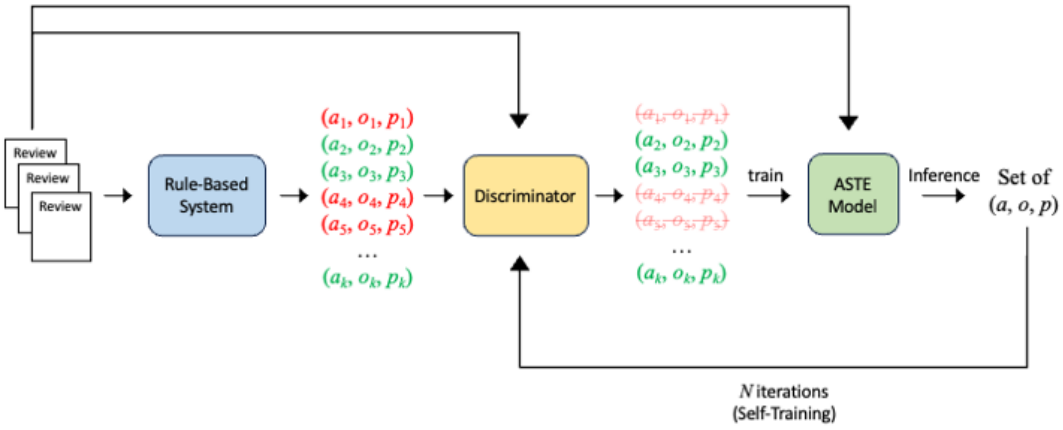

Figure 2: The Iterative Weak Supervision (IWS) Framework. The rule-based system initially generates multiple aspect sentiment triplets (ASTs), with red triplets representing incorrect ASTs and green triplets representing correct ASTs. Next, a discriminator is employed to filter out the incorrect ASTs, retaining only the correct ones as training data for an ASTE model. The model is fine-tuned by using this filtered training data. Subsequently, the fine-tuned ASTE model is utilized to infer on the training data, and the resulting outputs are fed back into the discriminator. These steps are performed iteratively to refine the training process using a technique known as "self-training".

Tagger (Li et al., 2020) to conduct word segmentation and assign a part-of-speech (POS) tag to each word. Next, lexicon detection and dependency parsing techniques are employed to identify the various aspects and opinions within the text. Subsequently, the concept of the shortest path in the dependency parse tree is utilized to effectively associate each aspect with its corresponding opinion. Finally, the opinions are categorized into one of three polarities (positive, negative, neutral) based on their sentiment valence value.

3.2.1. Opinion Lexicon Detection

We utilize E-HowNet (Ma and Shih, 2018; Chen et al., 2005), a general Chinese WordNet system, to construct an opinion lexicon. This system builds a comprehensive vocabulary model based on the semantic structure and complex relationships of words. E-HowNet encompasses various semantic categories, each with manually assigned valence values. We assign the same valence value to all words within the same semantic category, facilitating subsequent sentiment polarity classification. Following tokenization and POS tagging, if the segmented results match any entry in this opinion lexicon, we can include the mentioned item from the review into the collection of detected opinions.

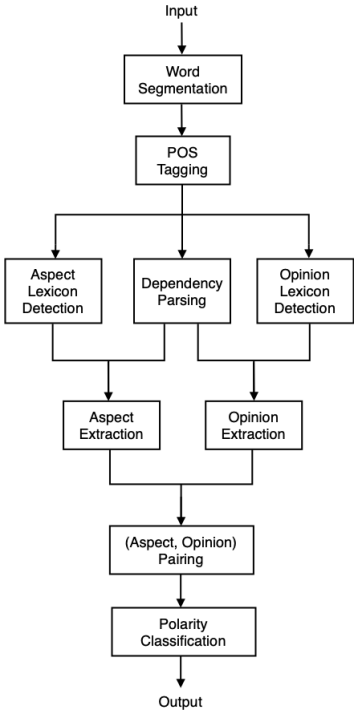

Figure 3: Pipeline processing flowchart of rule-based system. The system extracts a review's triplets of (aspect, opinion, polarity) through the pipeline process.

3.2.2. Aspect Lexicon Detection

To construct our aspect lexicon for ASTE, we include several aspects such as "food," "service," "price," and "atmosphere," among others. Specifically, we expand the "food" aspect in a fine-grained manner to cover a broad spectrum of dish names. In order to accurately capture dish names within

texts, we extract thirty thousand dish names from Ytower Recipe Website (楊桃美食網)[2]. This site offers a vast compilation of recipes categorized into six main groups: Japanese, Chinese, Western, Korean, South Asian, and miscellaneous. To expand the size of this aspect lexicon, we also add the segmented results of the crawled dish names as well as a few food names from E-HowNet. If the segmentation results match any entry in the aspect lexicon, the system will incorporate the mentioned item or dish from the review into the detected aspects.

3.2.3. Dependency Parsing

The tokenization and POS results are fed into the system's dependency parser, CKIP Dependency Parser (Ooi, 2022), which generates a dependency tree. Next, combining the detection results from the lexicon with the following two dependency parsing rules, we can obtain set A and set O, which are extractions of aspect terms and opinion terms. The dependency relationship and definition in this parser follows Marneffe et al. (2014).

After the extraction of aspects and opinions, we apply the shortest path method to get a set P containing all pairs of aspects and opinions. Note that the shortest path refers to the aspect node that is closest to a specific opinion node among all the aspect nodes in the dependency tree.

Shortest Path For each opn in O, for each asp in A, if (opn and asp are in the same sentence) && (asp is opn's the closest aspect in the dependency tree), then add (asp,opn) into set P.

After extracting all pairs of aspects and opinions, the system further enhances the precision of set P through a verification process. This involves checking for any nodes that are in a conjunction relationship with a specific aspect term, as well as identifying nodes that function as attributes of a particular aspect term. If a conjunction relationship is found, new aspect-opinion pairs are added to P. Additionally, if an attribute is detected, the resulting aspect term will include the original aspect term along with the identified attribute.

Conjunction and Attribute Detection For each (asp,opn) in P, for each $neighbor$ in $asp.neighbors$, if (dependency relation of asp and $neighbor$ is **conj**), then add ($nighbor$,opn) in P; if (dependency relation of asp and $neighbor$ is **assmod**) || (dependency relation of asp and $neighbor$ is **amod** && $neighbor$ is noun), representing asp's attribute is $neighbor$.

[2] https://www.ytower.com.tw/recipe/

3.2.4. Polarity Classification

According to the valence value of each paired opinion term, sentiment polarity is classified. The valence value is a positive number ranging from 1 to 10. We define values above 5.0 as positive, values below 5.0 as negative, and values equal to 5.0 as neutral. In this step, the system also checks if each opinion term has a negation relationship in the dependency parse tree. If a negation relationship is found, the polarity of the opinion term is reversed.

3.3. Encoder-Decoder Transformer Model

The encoder-decoder transformer model (Vaswani et al., 2017; Raffel et al., 2020) incorporates multiple layers of self-attention mechanisms and feed-forward neural networks, enabling it to effectively capture complex relationships between input and output sequences. This characteristic makes it highly suitable for Seq2Seq problems, including our ASTE tasks. Unlike Zhang et al.'s approach, we employ two transformer models: one dedicated to performing ASTE and another serving as the discriminator. Our goal is to exceed the performance of the rule-based system by leveraging these two models.

3.3.1. Self-Training

A base model M become M' through first fine-tuning using training data. After that, use M' to inference on training data to get labeled data, then generate new training data by blending the labeled predictions with training data. The new set is utilized to retrain M to obtain M'', then this process and previous steps continue iteratively. Note that in each iteration, the same base model M is fine-tuned, and this fine-tuned model is used for inference on its training data.

3.3.2. Discriminator

Our approach incorporates a classification mechanism that allows an encoder-decoder transformer model to act as a ternary classifier for extracting accurate ASTs.

Problem Definition In order to determine the correctness of the ASTE results, we need to address the following two questions based on given aspect, opinion, and polarity associated with a specific review text.

> Q1: Is `{opinion}` a description of `{aspect}`?
> Q2: Is `{opinion}` a `{polarity}` description of `{aspect}`?

The above questions have three possible combinations of answers.:

1."**YY**" (Yes / Yes): `{opinion}` describes `{aspect}` with correct `{polarity}`.
2."**YN**" (Yes / No): `{opinion}` describes `{aspect}` with wrong `{polarity}`.
3."**NX**" (No / -): `{opinion}` does not describes `{aspect}`. In this case, the answer to the second question does not matter.

Although only opinions classified as "YY" are included in the training data for self-training, the other two labels will be useful in subsequent experimental analysis. Additionally, the training data for the discriminator model is derived from a rule-based system. Initially organized at the review level, we restructure the ASTE results within each review into triplets consisting of aspect, opinion, and polarity. These triplets can be utilized to generate the aforementioned questions and their corresponding answers, serving as the input and output in a question-and-answer format for the discriminator.

Generate Training Data According to the definitions of the three labels mentioned above, we define "YY" data as positive examples, while "YN" and "NX" are considered negative examples. The following outlines the process of generating training data for the discriminator model.
Positive examples:
Label "YY" for all triplets of (aspect, opinion, polarity) generated by the rule-based system.
Negative examples:
Change the polarity of each "YY" triplet to a randomly selected incorrect polarity and label it as "YN". In the case of "NX" data, for each review, we start by generating a set that includes multiple pairs of (aspect, opinion) by combining all possible aspects and opinions. We subsequently remove the pairs that correspond to the "YY" data, regardless of their polarity. From this modified set, we randomly select a number of pairs twice the quantity of "YY" data. We assign a random polarity to each selected pair, forming a triplet labeled as "NX". If the quantity of remaining pairs is insufficient to reach twice the quantity of "YY" data, we select all the remaining pairs. Thus, it is possible that a review may not have any "NX" data.

4. Experiments

4.1. Datasets

We utilize the web scraping tool Outscraper to obtain all the reviews for 189 restaurants located in the Da'an District of Taipei City from Google Maps. In total, we collect 104,358 original review data written in Traditional Chinese. The restaurant reviews are processed by our rule-based system to extract the ASTs. Among these reviews, 74,028 are successfully analyzed, resulting in 798,614 ASTs. However, for the remaining 30,330 reviews, aspects or opinions may be missing, leading to the absence of ASTs. The ASTE results obtained from the analysis of the 74,028 reviews are used as training data for both the ASTE model and the discriminator in the self-training process.

4.2. mT5 Model

We use mT5 (Xue et al., 2021) as the experimental encoder-decoder transformer model. The base model is `mt5-base` pretrained on the DRCD (Shao et al., 2018) (Delta Reading Comprehension Dataset) for both ASTE model and discriminator. We utilize the question-answering format as described in the work of Xue et al. (2021) to structure the input and output of our model. In the following sections, we will provide detailed information regarding the processed input and output.

4.2.1. Discriminator

To address the defined problem in 3.3.2, the model receives an aspect sentiment triplet (AST) and its associated review text as original input and generates a corresponding label as the original output. We transform the original input and output into a question-answer format. These transformed questions and answers are used as the processed input and output for the mT5 discriminator model. Please refer to Figure 8 in the appendices for a detailed example. Table 1 displays data distributions of the discriminator. It is evident that the training set is considerably larger in size compared to the test and validation sets. We anticipate that such a substantial amount of training data will enable the model to possess strong discriminative capabilities.

Dataset	#(a, o, p)	#YY	#YN	#NX
train	778614	236463	239524	302627
valid	10000	3071	3082	3847
test	10000	3090	3021	3889

Table 1: The table shows the quantity of triplets of (aspect, opinion, polarity) as well as the number of three labels for the the discriminator. Note that ASTs from the same review may be distributed across different datasets.

This test, validation data, and training data, are all derived from the results of a rule-based model. After training, they are used to preliminarily evaluate the discriminator. However, since these results are not entirely accurate and the data is extensive, we must evaluate the performance of the discriminator through manually annotation. Section 3.3.2 explains positive and negative examples, where

positive examples are not completely correct, and negative examples are definitely incorrect.

4.2.2. ASTE Model

The initial input to the model is review text, while its primary output includes numerous ASTs that encapsulate the aspect, opinion, and polarity. As demonstrated in Table 2, the data distribution reveals a predominance of positive triplets. Consequently, it is anticipated that the model will exhibit enhanced performance in identifying positive expressions. In the subsequent section, a comprehensive discussion will be provided on the methodology employed to assess the ASTE model's performance, alongside a detailed explanation of the processed input and output.

Dataset	#reviews	#(a, o, p)	#Pos	#Neg	#Neu
train	64028	210100	147524	38568	24008
valid	5000	16232	11435	2953	1844
test	5000	16378	11501	3051	1826

Table 2: The table provides an overview of the distribution of reviews and their corresponding triplets with different polarities (positive, negative, and neutral) for the ASTE model.

Evaluation Metrics ASTE is one of multi-label tasks, thereby, we use example-based classification metrics (Zhang and Zhou, 2014) to compute $Accuracy$, $Precision$, $Recall$, and F_1 to evaluate the performance of mT5 ASTE model.

Processed Input and Output In order to determine the best processing template, we conduct an experiment comparing two candidate templates, as illustrated in Figure 4 and 5. The primary distinction between the two templates lies in whether the input question specifies the aspects. We hypothesize that the aspect lexicon and aspect expansion rule in our rule-based system are sufficiently powerful to extract most aspects, allowing the ASTE model to focus on identifying their corresponding opinions. To evaluate the performance of the templates, we train our discriminator and utilize it along with evaluation metrics to assess their effectiveness. The comparison results of the two templates are presented in Table 3, indicating that template 2 outperforms template 1. Therefore, we use template 2 for the mT5 ASTE model in the following experiments.

4.3. Experimental Setup

We design four self-training experiments, namely self-train-**A**, self-train-**B**, self-train-**C**, and self-train-**D**. The following pseudo-code in 4.3 and notations

Processed Input 1:

```
question: What are the aspects, opinions and
polarities in the following context? context:
While the beef noodle is indeed delicious and
the portion of meat is generous, it is
unfortunate that the service does not meet
expectations.
```

Processed Output 1:

```
beef noodle: delicious(positive).
portion of meat: generous(positive).
service: does not meet expectations(negative).
```

Processed Input 2:

```
question: What are the aspects in the
following context? What is the description of
the beef noodle in the following context? What
is the description of the meat portion in the
following context? What is the description of
the service in the following context? context:
While the beef noodle is indeed delicious and
the portion of meat is generous, it is
unfortunate that the service does not meet
expectations.
```

Processed Output 2:

```
beef noodle, portion of meat, service.
beef noodle: delicious(positive).
portion of meat: generous(positive).
service: does not meet expectations(negative).
```

Figure 4: The above figure depicts an example of template 1, while the below illustrates the same example of template 2.

Template	#(a, o, p)	YY(%)	$A.$	$P.$	$R.$	F_1
Template1	**16571**	93.7	73.8	79.1	80.8	79.4
Template2	16499	**95.3**	**91.4**	**93.9**	**95.2**	**94.2**

Table 3: The table presents the number of aspect sentiment triplets (ASTs), the "YY" labels assigned by the discriminator, and the evaluation scores of the mT5 ASTE model using two different templates. The data is derived from the output of the rule-based system, and its distribution corresponds to the information presented in Table 2.

outline the process and steps involved in conducting these experiments. Let S be the training set initially generated by the rule-based system. M denotes the base mT5 in self-training and M_i is a fine-tuned ASTE model in iteration i. D is a discriminator that output data with one of three labels ("YY", "YN", "NX"). s^* represents the subset of s containing opinions not exist in the opinion lexicon.

Here is the summary of four self-training experiments:

A: Before iteration 0, the training data is filtered using the discriminator to select only the correct ASTs. Then, in each iteration, the inference results are filtered using the discriminator to select only

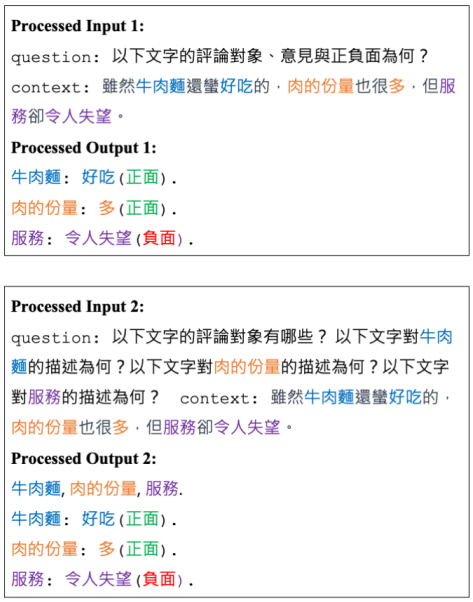

Figure 5: Chinese version of Figure 4.

Algorithm 1 SELF-TRAINING

if self-train-A **or** self-train-C **then**
  $S_0 =$ YY triplets in D.pred(S)
else
  $S_0 = S$
end if
$M_0 = M$.train(S_0)
for iteration $i = 1, 2, \ldots, n$ **do**
  $R = M_{i-1}$.pred(S_{i-1})
  if self-train-A **or** self-train-B **then**
    $R' =$ YY triplets in D.pred(R^*)
  else if self-train-C **then**
    $R' =$ YY triplets in D.pred(R)
  else
    $R' = R$
  end if
  $S_i = S_{i-1} \cup R'$
  $M_i = M$.train(S_i)
end for
return M_n

the ASTs that contain opinions not present in the opinion lexicon.
B: Do not use the discriminator before iteration 0. Other settings are similar to **A**.
C: Use the discriminator to filter training data before iteration 0, and all the inference results are filtered using the discriminator in each iteration.
D: Throughout the entire process, the discriminator is not used.

4.4. Evaluation

We plan to use Amazon Web Services (AWS) Mechanical Turk Workforce or ChatGPT to generate golden answers for evaluating the ASTE model. Generating ASTE results for each review using these services is extremely difficult, time-consuming, and expensive. Therefore, our approach involves uniting the results from four self-trained models and the rule-based system, and then using these services to label each AST as correct or not, considering the set of correct ASTs as the golden answers. To compare the accuracy of the two services and to evaluate the ASTE model and discriminator more accurately, we randomly selected 300 samples from the test set of the ASTE model. We then take the union of results from the final models of four experiments as well as the rule-based system, resulting in a total of 1175 ASTs. We let a worker follow the definition in 3.3.2 to manually assign the three labels to this data, which serves as the golden answer for this small-scale test set.

	Labels			Accuracy	
Method	#YY	#YN	#NX	(a, o)	(a, o, p)
without mT5 Discr.	1175	0	0	0.61	0.56
mT5 Discr.	1083	10	82	0.65	0.6
AWS	885	136	154	0.59	0.48
ChatGPT	606	86	483	**0.85**	**0.83**
manual annotation	658	62	455	1	1

Table 4: The table presents the label distribution and accuracy across five labeling methods: without mT5 discriminator, with mT5 discriminator, AWS, ChatGPT (gpt-3.5-turbo-16k), and manual annotation, which serves as the golden answer.

Table 4 displays the accuracy and label distribution based on the human-labeled golden answers. It indicates that the discriminator achieves higher accuracy compared to not using it. Furthermore, ChatGPT exhibits a high accuracy rate of 80%, which is significantly higher than accuracy rate of labeling service in AWS. Given the challenges associated with manually labeling a large-scale dataset, we utilize ChatGPT to generate the golden answers for the complete test set of the ASTE models. This table also shows that ChatGPT's labeling accuracy is close to manually labeling, making it suitable for evaluating the complete test dataset of 5000 reviews.

Table 5 presents the evaluation metrics for both the rule-based system and four mT5 ASTE models with self-training. In all four experiments, the R scores of the ASTE models outperform those of the rule-based system, demonstrating the effective enhancement of recall through iterative weak supervision while maintaining precision at a level similar to that of the rule-based system. Notably, self-train-C exhibits a recall 2% higher than that of the rule-based system, indicating its capability to generate more correct ASTs. For example, in the sentence `The water spinach with plenty of garlic goes well with rice` (空心菜蒜頭很多，很下飯), the rule-based system only detects `(garlic`蒜頭`, goes well with rice`下飯`, positive`正面`)`. In contrast,

Model	#reviews	$A.$	$P.$	$R.$	F_1
Rule-Based	300	67.49	69.26	86.67	74.57
Self-train-A		68.38	69.85	88.05	75.49
Self-train-B		67.41	68.85	87.31	74.45
Self-train-C		68.00	69.04	88.45	75.12
Self-train-D		67.34	68.51	88.08	74.40
Rule-Based	5000	52.04	53.25	75.59	59.85
Self-train-A		52.52	53.66	76.51	60.44
Self-train-B		52.36	53.33	76.68	60.24
Self-train-C		52.67	53.53	77.42	60.60
Self-train-D		52.37	53.17	77.62	60.33

Table 5: Model comparison between rule-based system and self-training's last iteration of different settings. The red numbers represent the highest score in each dataset, while the blue numbers denote the lowest values. The golden answers for the 300 test samples above were manually labeled by an annotator, while the golden answers for the 5000 test samples below were generated by ChatGPT.

self-train-C is capable of generating not only the original AST but also an additional AST: `(water spinach空 心 菜, goes well with rice下飯, positive正面)`.

Furthermore, based on the F_1 scores, we have the following findings.

A better than B: Instead of completely relying the output of the rule-based system, it is better to use the discriminator to filter the training data first.

C better than D: Using discriminator may be useful for the ASTE model during iterative training.

A & B ≈ C & D: No significant difference in performance.

4.5. Analysis

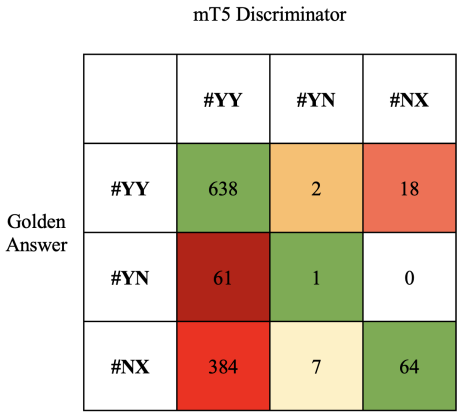

Figure 6: Distribution of correct and incorrect labels of mT5 discriminator. The number in each cell shows the quantity of labels.

Building on the results presented in Table 4, Figure 6 illustrates that the mT5 discriminator primarily classifies most cases as "YY," indicating a limited ability to identify negative examples despite the large size of the negative data.

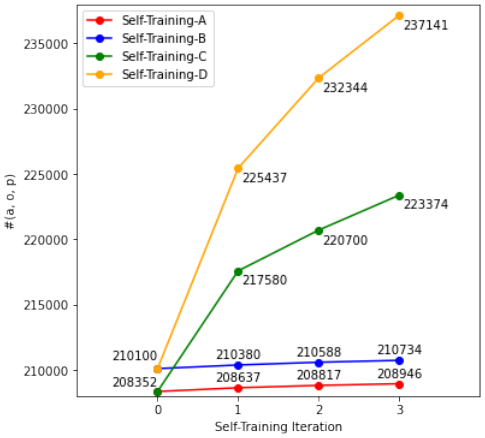

Figure 7: The quantity of triplets in training data of self-training with different settings.

Figure 7 illustrates the growth trend of the number of Aspect Sentiment Triplets (ASTs) in the training data across each iteration. To enhance the ASTE model's comprehension of diverse opinions, we incorporate ASTs generated during inference into the training dataset, excluding any duplicate triplets. Furthermore, we employ a discriminator to meticulously filter and select accurate ASTs, ensuring the high quality of the training data is maintained. The blue and red lines are flatter because self-training-A and B only add few triplets with opinions not exist in the lexicon to training data in each iteration. Table 5 illustrates that although there isn't a marked improvement in performance, our models excel at producing a significantly larger volume of training data than traditional rule-based methods. Figure 7 further emphasizes a notable growth in the quantity of ASTs across all four models. Additionally, the data in Table 5 not only confirms the stability of our models' quality but also indicates a slight enhancement in their performance.

Dataset	Model	(a, o, p)* (‰)	$A.$
test	Self-train-A	0.50	25.00
	Self-train-B	0.97	25.00
	Self-train-C	0.36	0.00
	Self-train-D	1.11	**36.84**
valid	Self-train-A	0.69	**45.45**
	Self-train-B	0.98	43.75
	Self-train-C	1.03	41.18
	Self-train-D	1.86	15.63

Table 6: The table shows the ratio and accuracy of triplets with opinions not present in the opinion lexicon in the test set of 5000 reviews. Note that the golden answers of these small data is determined by a labeling worker.

In Table 6, we analyze the proportion of ASTs in the test set that contain expressions not present in

the opinion lexicon. The full table and example are displayed in Table 8 and Figure 9 in the appendices. Although the ASTE model is capable of generating a small number of out-of-lexicon expressions, the actual accuracy of these ASTs is relatively lower. This also represents one of the important issues that we can further explore in our future research.

5. Chinese ASTE Dataset

Table 7 provides information about our proposed dataset. We choose "self-train-C" as the source for the training set because it outperforms other models, as shown in Table 5. Additionally, we offer data generated by the rule-based system for validation, as well as two different test set sizes with two types of annotations to accommodate various applications in the relevant field. To the best of our current knowledge, this is the first complete Chinese ASTE dataset, is significantly larger than other Chinese ABSA datasets, such as DiaASQ (Li et al., 2023), encompassing over 200,000 ASTs extracted from more than 60,000 restaurant reviews. We also release the training and inference methods of our ASTE mT5 models to enable the public to use them.

Dataset	#reviews	Source Model	Golden Answer
train	64007	Self-train-C	-
valid	5000	Rule-Based	-
test	300	Union of Models	Labeling worker
	5000	Union of Models	ChatGPT

Table 7: Data distribution of our dataset. "Union of Models" means union of four self-train models and the rule-based system.

6. Conclusion

Confronted with the scarcity of ASTE datasets, we have devised a unique framework for the automatic creation of ASTE. This framework leverages a novel weakly supervised self-learning methodology, which negates the necessity for labor-intensive manual labeling. Assisted by a discriminator to eliminate inferior samples, which can incrementally enhance the quality of the initial noisy labeled data while also generating additional data. The success of our approach is affirmed by our results, which include the assembly of a considerable Chinese review dataset. Moreover, our experimental findings reveal that we have successfully established a robust baseline model. Demonstrating our commitment to the research community, both our dataset and model have been made publicly accessible.

7. Ethical Considerations

We prioritize privacy concerns, ensuring that the published dataset omits personal information, restaurant names, and related specifics to protect individual privacy.

8. Limitations

Our rule-based system relies on lexicon detection to extract opinion words, mainly adjectives or nouns. As a result, the mT5 model trained on this data struggles to identify complex expressions, such as opinion descriptions in sentence form, such as `"I will never come to this restaurant again."` Our approach can only detect phrases or idioms at most. This limitation is a common challenge observed in various ASTE datasets, including popular benchmarks like SemEval. As one of pioneers for the Chinese ASTE dataset, we hope that future research can build upon our work and enhance the ability to extract complex and ambiguous opinion terms.

9. Acknowledgments

We are grateful for the insightful and valuable comments from anonymous reviewers. This work is supported by the Ministry of Science and Technology of Taiwan under grant numbers 111-2634-F-001-001. We thank to National Center for High-performance Computing (NCHC) for providing computational and storage resources.

10. Bibliographical References

Jiahao Bu, Lei Ren, Shuang Zheng, Yang Yang, Jingang Wang, Fuzheng Zhang, and Wei Wu. 2021. ASAP: A chinese review dataset towards aspect category sentiment analysis and rating prediction. CoRR, abs/2103.06605.

Keh-Jiann Chen, Shu-Ling Huang, Yueh-Yin Shih, and Yi-Jun Chen. 2005. Extended-hownet- a representational framework for concepts. In Proceedings of OntoLex 2005 - Ontologies and Lexical Resources.

Shaowei Chen, Yu Wang, Jie Liu, and Yuelin Wang. 2021. Bidirectional machine reading comprehension for aspect sentiment triplet extraction. CoRR, abs/2103.07665.

Zhifang Fan, Zhen Wu, Xin-Yu Dai, Shujian Huang, and Jiajun Chen. 2019. Target-oriented opinion words extraction with target-fused neural sequence labeling. In Proceedings of

the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pages 2509–2518, Minneapolis, Minnesota. Association for Computational Linguistics.

Bobo Li, Hao Fei, Fei Li, Yuhan Wu, Jinsong Zhang, Shengqiong Wu, Jingye Li, Yijiang Liu, Lizi Liao, Tat-Seng Chua, and Donghong Ji. 2023. Diaasq : A benchmark of conversational aspect-based sentiment quadruple analysis.

Peng-Hsuan Li, Tsu-Jui Fu, and Wei-Yun Ma. 2020. Why attention? analyze bilstm deficiency and its remedies in the case of ner. In Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence, volume 34, pages 8236–8244.

Wei-Yun Ma and Yueh-Yin Shih. 2018. Extended hownet 2.0 – an entity-relation common-sense representation model. In Proceedings of Language Resources and Evaluation Conference (LREC 2018), Miyazaki, Japan. European Language Resources Association (ELRA).

Marie-Catherine de Marneffe, Timothy Dozat, Natalia Silveira, Katri Haverinen, Filip Ginter, Joakim Nivre, and Christopher D. Manning. 2014. Universal stanford dependencies: A cross-linguistic typology. In Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14), pages 4585–4592, Reykjavik, Iceland. European Language Resources Association (ELRA).

Chin-Yin Ooi. 2022. Head index refinement model in dependency parsing.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library.

Haiyun Peng, Lu Xu, Lidong Bing, Fei Huang, Wei Lu, and Luo Si. 2020. Knowing what, how and why: A near complete solution for aspect-based sentiment analysis. Proceedings of the AAAI Conference on Artificial Intelligence, 34(05):8600–8607.

Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Ion Androutsopoulos, Suresh Manandhar, Mohammad AL-Smadi, Mahmoud Al-Ayyoub, Yanyan Zhao, Bing Qin, Orphée De Clercq, Véronique Hoste, Marianna Apidianaki, Xavier Tannier, Natalia Loukachevitch, Evgeniy Kotelnikov, Nuria Bel, Salud María Jiménez-Zafra, and Gülşen Eryiğit. 2016. SemEval-2016 task 5: Aspect based sentiment analysis. In Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016), pages 19–30, San Diego, California. Association for Computational Linguistics.

Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Suresh Manandhar, and Ion Androutsopoulos. 2015. SemEval-2015 task 12: Aspect based sentiment analysis. In Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015), pages 486–495, Denver, Colorado. Association for Computational Linguistics.

Maria Pontiki, Dimitris Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. 2014. SemEval-2014 task 4: Aspect based sentiment analysis. In Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014), pages 27–35, Dublin, Ireland. Association for Computational Linguistics.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. Journal of Machine Learning Research, 21(140):1–67.

Chih-Chieh Shao, Trois Liu, Yuting Lai, Yiying Tseng, and Sam Tsai. 2018. Drcd: a chinese machine reading comprehension dataset. ArXiv, abs/1806.00920.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In Advances in Neural Information Processing Systems, volume 30. Curran Associates, Inc.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Transformers: State-of-the-art natural language processing. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, pages 38–45, Online. Association for Computational Linguistics.

Lu Xu, Hao Li, Wei Lu, and Lidong Bing. 2020. Position-aware tagging for aspect sentiment triplet extraction. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 2339–2349, Online. Association for Computational Linguistics.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. mT5: A massively multilingual pre-trained text-to-text transformer. In Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 483–498, Online. Association for Computational Linguistics.

Min-Ling Zhang and Zhi-Hua Zhou. 2014. A review on multi-label learning algorithms. Knowledge and Data Engineering, IEEE Transactions on, 26:1819–1837.

Wenxuan Zhang, Xin Li, Yang Deng, Lidong Bing, and Wai Lam. 2021. Towards generative 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 2: Short Papers), pages 504–510, Online. Association for Computational Linguistics.

Wenxuan Zhang, Xin Li, Yang Deng, Lidong Bing, and Wai Lam. 2022. A survey on aspect-based sentiment analysis: Tasks, methods, and challenges. IEEE Transactions on Knowledge and Data Engineering, pages 1–20.

11. Appendices

11.1. Implementation Details

The discriminator is trained initially for three epochs with a maximum input length of 1024 tokens and a maximum output length of 64 tokens. Model selection is based on the accuracy achieved on the validation set. Each self-training setting consists of three iterations, with each iteration trained for five epochs. The maximum input length is set at 1024 tokens and the maximum output length is set at 512 tokens. The optimal model is selected based on the F_1 score on the validation set. Training and inference are performed using a single NVIDIA Tesla V100-SXM2-32GB GPU and a single NVIDIA Quadro RTX 8000 GPU. We use a batch size of 2 and implement the models using Hugging Face Transformers (Wolf et al., 2020) version 4.28.1 and PyTorch (Paszke et al., 2019) version 2.0.0.

Figure 8: The example of mT5 discriminator's processed input and output.

Figure 9: The correct and incorrect example of AST with opinion not present in the opinion lexicon.

Dataset	Model	#reviews	#(a, o, p)	#(a, o, p)*	(a, o, p)* (‰)	#YY	#YN	#NX	$A.$
test	Self-train-A	5000	16085	8	0.50	2	0	6	25.00
	Self-train-B		16518	16	0.97	4	1	11	25.00
	Self-train-C		16649	6	0.36	0	1	5	0.00
	Self-train-D		17170	19	1.11	7	0	12	36.84
valid	Self-train-A	5000	15891	11	0.69	5	0	6	45.45
	Self-train-B		16405	16	0.98	7	1	8	43.75
	Self-train-C		16480	17	1.03	7	0	10	41.18
	Self-train-D		17207	32	1.86	5	1	26	15.63

Table 8: Inference results of triplets with opinions not present in the opinion lexicon.