symmetry

MDPI

Article

"Opinion Target: Opinion Word" Pairs Extraction Based on CRF

Yan Yan [1,*], Faguo Zhou [1], Yifan Ge [1], Cheng Liu [1] and Jingwu Feng [2]

[1] School of Mechanical Electronic and Information Engineering, China University of Mining and Technology Beijing, Beijing 100083, China; zhoufaguo@cumtb.edu.cn (F.Z.); sqt1900405071@student.cumtb.edu.cn (Y.G.); 1810410221@student.cumtb.edu.cn (C.L.)

[2] School of Computer Science, China University of Geosciences, Wuhan 430078, China; emppenguin@cug.edu.cn

* Correspondence: yanyan@cumtb.edu.cn

Abstract: With the spread of mobile applications and online interactive platforms, the number of user reviews are increasing explosively and becoming one of the most important ways for users to voice opinions. Opinion target extraction and opinion word extraction are two key procedures used to determine the helpfulness of reviews. In this paper, we implement a system to extract "opinion target:opinion word" pairs based on the Conditional Random Field (CRF). Firstly, we used the CRF model to extract opinion targets and opinion words, then combined these into pairs in order. In addition, Node.js was used to build a visualization system to display "opinion target:opinion word" pairs. In order to verify the effectiveness of the system, experiments were conducted on the Laptop and Restaurant datasets of SemEval-2014-task4, and the accuracy of the F value extracted by the model reached 86% and 90%, respectively. All the code and datasets for this experiment are available on GitHub.

Keywords: CRF; opinion target; opinion word

Citation: Yan, Y.; Zhou, F.; Ge, Y.; Liu, C.; Feng, J. "Opinion Target: Opinion Word" Pairs Extraction Based on CRF. *Symmetry* **2021**, *13*, 251. https://doi.org/10.3390/sym13020251

Academic Editor: Giuseppe Bagliesi
Received: 4 January 2021
Accepted: 29 January 2021
Published: 2 February 2021

1. Introduction

With the popularity of mobile payment, the increasing numbers of consumers are causing the numbers of reviews of products to increase exponentially. If these evaluation sentences are condensed and only the opinion target and opinion words are retained, it can help consumers to shop online, such as when booking hotels, ordering takeaways, etc., allowing users to quickly locate the information they need. Figure 1 shows an example of a sentence that has been condensed and whose opinion target and opinion word were retained. The sentence "The battery life seems to be very good, and have had no issues with it; enabling the battery timer is useless." includes two opinion targets, and each target matches one opinion word. In this condition, customers could be interested in different parts of the item, so they want to obtain the key information with its corresponding evaluation rather than a single sentiment polarity score to help them learn everything about the product. At the same time, e-commerce sellers can easily obtain consumers' concerns and gain instant feedback to improve their service. It can be seen that the research on "opinion target:opinion word" pair extraction is very meaningful. Thus, we aimed to extract the opinion targets and opinion words from reviews. The traditional methods for opinion target extraction are based on frequency or manually designed rules [1–4], and it is not difficult to find obvious disadvantages, such as the fact that they are easily influenced by noise and have low flexibility, etc. With the development of machine learning theory, increasing numbers of researchers would like to choose a CRF-based model [5] to complete the same task. Current research into opinion target extraction is limited to indicators such as accuracy, without extracting them and making pairs. In view of the research situation mentioned above, we designed a method based on the CRF model to extract opinion targets and opinion words and match them into pairs, obtaining better performance due to the use of properly defined templates compared to most of other methods. In order to display the

results visually, we built a system with Node.js which is able to simplify the input review sentence, allowing customers to easily obtain the information they need. The domains we focused on are restaurant and laptop reviews, and we conducted experiments on relevant datasets from SemEval task 4.

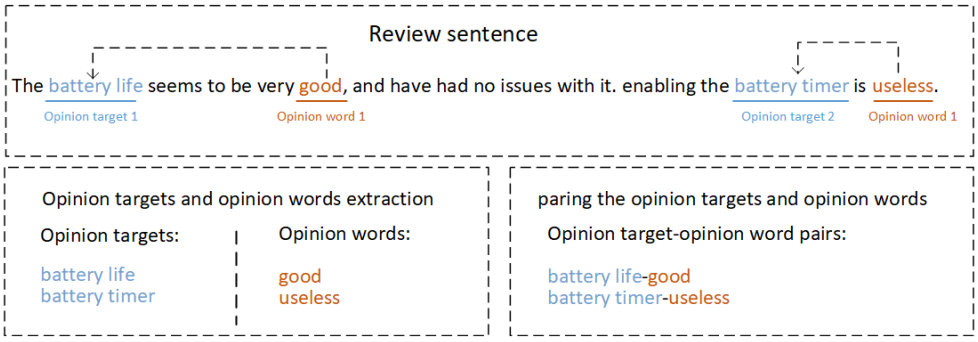

Figure 1. An example of extraction Opinion target and Opinion word.

2. Related Work

2.1. Overview of the Methods of Opinion Targets Extraction

The main methods currently applied to extract opinion targets are as follows: (1) Frequency-based methods [1,2]; (2) Rule/template-based methods [3,4]; (3) Graph theory-based methods [6–8]; (4) Based on CRF or combined CRF and deep learning methods [5,9–25].

The frequency-based method is quite simple and has a high recognition rate for opinion targets with high frequency, but it is easy for it to increase noise and miss opinion targets with low frequency [1]. While the method based on rules/templates has high extraction accuracy, it is limited to a specific field, resulting in the poor generality of the model [5,9]. Methods based on graph theory often assume that the opinion targets are nouns, nominal phrases, adjectives or adjective phrases, which makes it very difficult for the method to recognize targets that do not correspond to those types. The CRF-based method is widely used in the field of opinion target extraction, and its performance is better than the previous models. Because the traditional opinion target recognition method relies on a large number of manually selected features, it is time-consuming. On the other side, the ability of feature extraction and the nonlinear fitting of the neural network model can play an important role in opinion target extraction [10]; thus, in recent years, deep learning methods have begun to be applied to the extraction of opinion targets. There have been many models that combine neural networks and CRFs. For instance, Yin [11] serialized embedding learned by neural networks as features for CRF models. Wang [12] combined CRF and recurrent neural network to extract opinion targets, reducing the work of artificially constructing features. Al-Smadi [13] combined long short-term memory (LSTM) and CRF to construct a comment target extraction model. Hu [14] designed a bidirectional long short-term memory (BiLSTM)–CRF model to achieve opinion extraction of product reviews. Danish [15] proposed a bidirectional encoder representations from transformers (BERT)–BiLSTM–CRF model for entity-seeking of multi-sentence. Huang [16] also used the BiLSTM-CRF model for sequence labeling tasks. Zhao [17] designed a span-based framwork with BERT encoder and BiLSTM encoder to co-extract aspect and opinion terms. Using deep learning methods to extract opinion targets is a research trend, but there are still problems when faced with high computational complexity and a large number of data samples. In contrast, simple CRF-based methods can achieve higher accuracy and a lower time cost.

2.2. Conditional Random Field (CRF)

2.2.1. Principle

The CRF model was first proposed by Lafferty [18] and is an improvement of Hidden Markov Model (HMM) and Maximum Entropy Markov Model (MEMM) [19]. Compared

with the two models above, CRF can better capture context information [9] and can perform the global normalization of the probability, so it is widely-used for tasks such as sequence labeling [20]; in fact, opinion target extraction can be regarded as the same task. Let x={x_1,x_2,x_3,...,x_n} be the observation sequence, and y={y_1,y_2,y_3,...,y_n} be the state sequence. In the case of a given observation sequence x, the probability formula for calculating y is as follows:

$$p(y|x,\lambda) = \frac{1}{z(x)} exp(\sum_{i=1}^{n}\sum_{j=1}^{K}\lambda_j f_j(y_{i-1}, y_i, x, i)) \quad (1)$$

$$z(x) = \sum_{y} exp(\sum_{i=1}^{n}\sum_{j=1}^{K}\lambda_j f_j(y_{i-1}, y_i, x, i)) \quad (2)$$

where $z(x)$ is a normalization factor, which means the sum of feature functions of sequence y, K is the number of features, $f_j(y_{i-1}, y_i, x, i)$ is the feature function, and λ_j is the weight.

The feature function is the core of CRF, which is defined according to the selected features, tags and templates, Shaped as follows:

if(y_i==T&&x==N) output 1 else output 0

The meaning of the above feature function is that when the observation sequence x satisfies a certain condition N and the state sequence is marked as T, it outputs 1; otherwise, it outputs 0. Each feature function will have a weight, which can be obtained after training. The value of weight λ_j indicates the probability of y_i being marked as T. The task of sequence labeling is as follows: given an observation sequence, a state sequence y with the greatest possibility can be finally obtained.

2.2.2. Research Situation

There have been many research works on the extraction of opinion targets based on the CRF model. Jakob [21] used CRF model to extract opinion targets in single-domain and cross-domain datasets. Hu [10] based on the CRF method, and with the help of the recurrent neural network (RNN) model, carried out related research on the extraction of English and Dutch named entity recognition(NER) datasets, and achieved good results in terms of accuracy. Tran [22] combined bidirectional gated recurrent unit (BiGRU) and CRF to extract aspects of the SemEval-2014 dataset, obtaining better accuracy than the state-of-the-art methods. The methods above achieved good results in terms of extraction accuracy.

However, the research works above only focused on the accuracy of model extraction, recall rate, etc., which cannot indicate the corresponding relationship between opinion targets and opinion words in a review sentence. Wang [12] proposed the Recursive Neural Conditional Random Fields model to learn a high-level feature representation of the context, which is able to co-extract the opinion target and opinion words. Veyseh [23] designed a Graph Convolutional Neural Network model called ONG with Syntax-Model Consistency and Representation Regularization, which exploited the syntactic information.

On the basis of the above research listed in Table 1, we extracted "opinion target: opinion words" pairs for matching and then displayed them with a front-end interface to achieve the purpose of streamlining the comment sentence and visualizing the effective part of the comment.

Table 1. Summarization of the methods above.

	Method	Dataset	Precision	Recall	F1
Jakob [21]	CRF	movies	66.3	59.2	62.5
		web-services	62.4	39.4	48.3
		cars	25.9	42.6	32.2
		cameras	42.3	43.1	42.6
Hu [10]	NCRF	CoNLL-2003 English	-	-	91.66
		CoNLL-2002 Dutch	-	-	81.94
Tran [22]	BiGRU+CRF	SemEval-2014 Restaurant	-	-	85.00
		SemEval-2014 Laptop	-	-	78.50
Wang [12]	RNCRF	SemEval-2014 Restaurant	-	-	84.52
		SemEval-2014 Laptop	-	-	78.93
Veyseh [23]	ONG	SemEval-2014 Restaurant	83.23	81.46	82.33
		SemEval-2014 Laptop	73.87	77.78	75.77

3. Methods

We used the CRF++0.58 toolkit to build the CRF model. The toolkit can automatically generate feature functions based on the input features in the training dataset, user-defined tags and templates and can calculate weights for each feature function. Through these trained feature functions, the toolkit can mark the sentences of the test dataset (user-defined marks). This is the principle of the CRF model for opinion target extraction. The building of the model of the process consists of three parts: (1) data cleaning, (2) feature selection and template definition, and (3) model training, seeing as Figure 2.

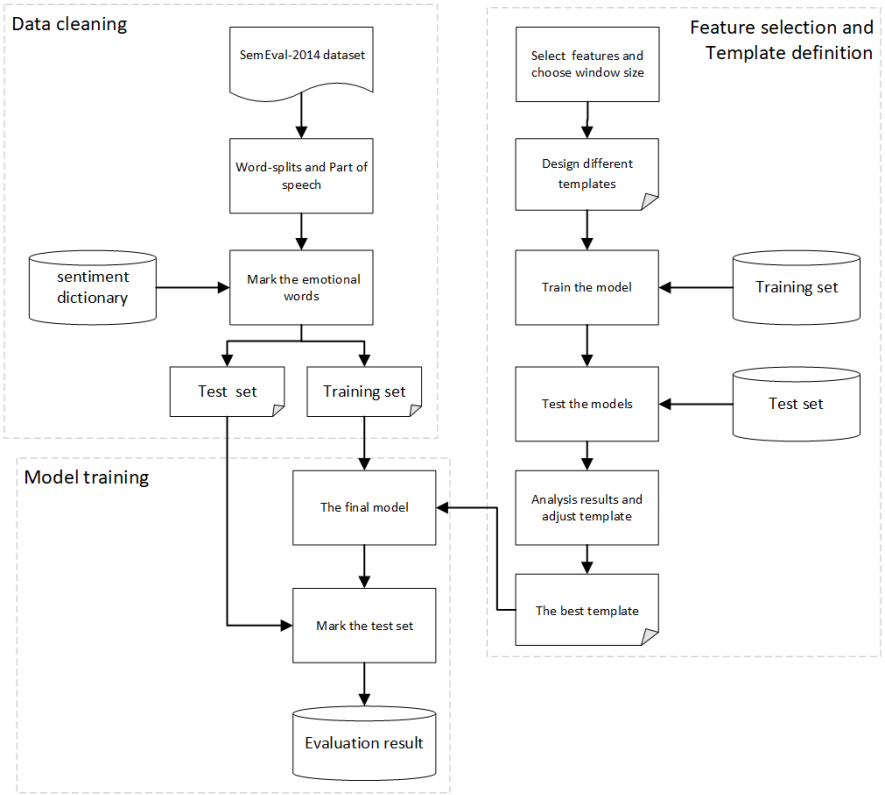

Figure 2. The overall architecture of the CRF framework.

3.1. Data Cleaning

CRF is a semi-supervised machine learning model that requires a large labeled corpus for training. The data cleaning mainly comprises three steps: the first is to add the chosen features to the sentence corpus, such as parts of speech, so that the CRF can make use of them; the second is to add marks, and use the marks to identify the opinion target and opinion words; the third is to standardize the format of the corpus to the input data format. The details are as follows.

3.1.1. Word-Splits and Part of Speech

Word segments and parts of speech are also effective features for opinion target extraction, and tagging the two features are two indispensable steps in natural language processing. These two parts of work are processed by using the NLTK library http://www.nltk.org.

3.1.2. Mark Emotional Words

Because the SemEval dataset itself lacks labels for opinion words, we used the sentiment dictionary of NLTK library http://www.nltk.org/api/nltk.sentiment.html?highlight=sentiment to mark opinion words to obtain a large amount of labeled data.

3.1.3. The Label Set

CRF was used to extract opinion targets, annotating the sequence of review sentences. We used {O, T, C} to mark the dataset, where O represented irrelevant items, T represented opinion targets and C indicated opinion words. An example sentence is as follows: "But the stuff was so horrible to us". According to the input data format requirements of CRF++0.58, the output format matrix after cleaning is shown in Table 2. The data were divided into five columns. The first four columns were regarded as features by the CRF++0.58 toolkit and corresponded to words and parts of speech, whether it is an emotional word and the syntactic relationship. The last column was regarded as a tag.

Table 2. Final format after data cleaning.

Word	Part of Speech	Emotion	Relation	Tag
But	NN	N	OP	O
the	DT	N	NP	O
stuff	NN	N	NP	T
was	VB	N	VP	O
so	RB	N	OP	O
horrible	JJ	Y	NP	C
to	TO	N	OP	O
us	PRP	N	OP	O

3.2. Feature Selection and Template Definition

Features were the basis for the CRF model to mark the sequence and allowed the model to identify the components of the sentence. For instance, opinion targets are generally nouns, so parts of speech are very helpful for CRF to identify opinion targets. The template determined the format of the generated feature functions, which contained the window size and feature selection. It can be seen that choosing appropriate features and defining proper templates were the keys to a model with good performance.

3.2.1. Feature Selection

Because of the unstructured and optional nature of the comment sentences, we selected words, parts of speech and syntactic relationships as features to extract opinion targets. Part of speech features were implemented through the NLTK library, and syntactic relations were extracted by using regular expressions. We chose four syntactic relationships: NP

(noun phrase), P (preposition), PP (prepositional phrase) and VP (verb phrase). Whether a word was an emotional word was determined through the emotional dictionary.

3.2.2. Template Definition

CRF uses a token to mark the current word. When training the model, each sentence was converted into the CRF standard format as shown in Table 2; an N-word sentence was converted into an N *rows* $\times$ 5 *column* (four feature columns and tag column) matrix, then the token moved from the first row to the end, reading different words and its features. Templates extracted different features according to the template rule. We used the function w, p, r, e to represent words, parts of speech, syntactic relationships and whether words were emotional.

We designed five templates to investigate the influence of features on model performance, of which the last template (Template 5) was the integration of other templates. The window sizes of Template 1 and Template 2 were different from those of Template 3 and Template 4, in order to investigate the influence of the window size on the performance of the model. Template 1 and Template 3 used different feature combinations than those of Template 2 and Template 4 to investigate the influence of dependency syntax relationship features. Template 5 was the ultimate template based on the experimental results of the above four templates. See Table 3 for details.

Table 3. Templates.

Name	Template
Template1	w(−1)/w(0)/w(1)/w(−1,0)/w(0,1)/w(−1,0,1) p(−1)/p(0)/p(1)/p(−1,0)/p(0,1)/p(−1,0,1) e(0)p(0)
Template2	w(−1)/w(0)/w(1)/w(−1,0)/w(0,1)/w(−1,0,1) p(−1)/p(0)/p(1)/p(−1,0)/p(0,1)/p(−1,0,1) r(−1)/r(0)/r(1)/r(−1,0)/r(0,1)/r(−1,0,1) e(0)p(0)
Template3	w(−1)/w(0)/w(1)/w(−2)/w(2)/w(1,0)/w(0,1)/ w(−1,0,1)/w(−2,−1,0)w(0,1,2)/w(−2,−1,0,1,2) p(−1)/p(0)/p(1)/p(−2)/p(2)/p(1,0)/ph(0,1)/p(−1,0,1)/ p(−2,−1,0)p(0,1,2)/p(−2,−1,0,1,2) e(0)p(0)
Template4	w(−1)/w(0)/w(1)/w(−2)/w(2)/w(1,0)/w(0,1)/ w(−1,0,1)/w(−2,−1,0)w(0,1,2)/w(−2,−1,0,1,2) p(−1)/p(0)/p(1)/p(−2)/p(2)/p(1,0)/p(0,1)/p(−1,0,1)/ p(−2,−1,0)p(0,1,2)/p(−2,−1,0,1,2) r(−1)/r(0)/r(1)/r(−2)/r(2)/r(1,0)/r(0,1)/r(−1,0,1)/ r(−2,−1,0)r(0,1,2)/r(−2,−1,0,1,2) e(0)p(0)
Template5	w(−1)/w(0)/w(1)/ w(−1,0) p(−2)/p(−1)/p(0)/p(1)/p(2)/p(1,0)/p(0,1)/p(−1,0,1) r(−1)/r(0)/r(1)/r(−2)/r(2)/r(1,0)/r(0,1)/r(−1,0,1)/ r(−2,−1,0)r(0,1,2)/r(−2,−1,0,1,2) e(0)p(0)

Function $w(x_1, ..x_n)$ represents the combination of x_1th, ... and x_nth word, where 0 represents the current position and "-" or "+" ("x" means "$+x$") represents the previous or next position. Similarly, functions p, e and r represent the combination of each feature of words in different positions. For example, $w(-1, 0)$ means the combination of the previous word and current word, and $p(0, 1)$ means the combination of the current word's part of speech and the next word's part of speech. Conversely, the templates in Table 3 are in an abstract form, allowing us to comprehend the meanings easily. The real templates in

CRF++ are presented as constraint functions. With Template 1 used as an example, Table 4 shows its concrete form.

Table 4. Template interpretation, the two dimensions of function %x[] represent the word relative position and which feature selected.

Template	Meaning
U01:%x[−1,0]	previous word
U02:%x[0,0]	current word
U03:%x[1,0]	next word
U04:%x[−1,0]/%x[0,0]	Combination of previous and current word
U05:%x[0,0]/%x[1,0]	Combination of current and next word
U06:%x[−1,0]/%x[0,0]/%x[1,0]	Combination of previous, current and next word
U07:%x[−1,1]	Part of speech of previous word
U08:%x[0,1]	Part of speech of current word
U09:%x[1,1]	Part of speech of next word
U10:%x[−1,1]/%x[0,1]	Combination of previous and current word's part of speech
U11:%x[0,1]/%x[1,1]	Combination of current and next word's part of speech
U12:%x[1,1]/%x[0,1]/%x[1,1]	Combination of previous, current and next word's part of speech
U13:%x[0,2]/%x[0,1]	The combination of the current word's part of speech and whether it is an emotional word

%x[*row*,*column*] shows the *row*th word's *column*th feature, where *column* = 0, *column* = 1, *column* = 2 and *column* = 3 represent the word itself, its part of speech, its emotional content and its contextual relationship. CRF++0.58 could automatically generate a feature function according to the template we defined. Taking function U01 as an example, processing the sentence in Table 2, when token is in the position of the word "stuff", it generates functions as below:

if(y==T&&x==stuff) output 1 else 0
if(y==O&&x==stuff) output 1 else 0
if(y==C&&x= =stuff) output1 else 0
which means that, whatever the tag is, it will output 1 if the word is "stuff".

3.3. Train the Model of Opinion Targets Extraction

We trained the model according to the steps in Figure 3.

Firstly, the data were cleaned according to Section 3.1, dividing the processed data into a training set and testing set randomly. Secondly, based on CRF++0.58, the templates were designed to train the model. Thirdly, the test dataset was marked with the trained model. Finally, the words marked as T (opinion target) and C (opinion word) were separated in the result and matched to the "opinion target: opinion word" pairs.

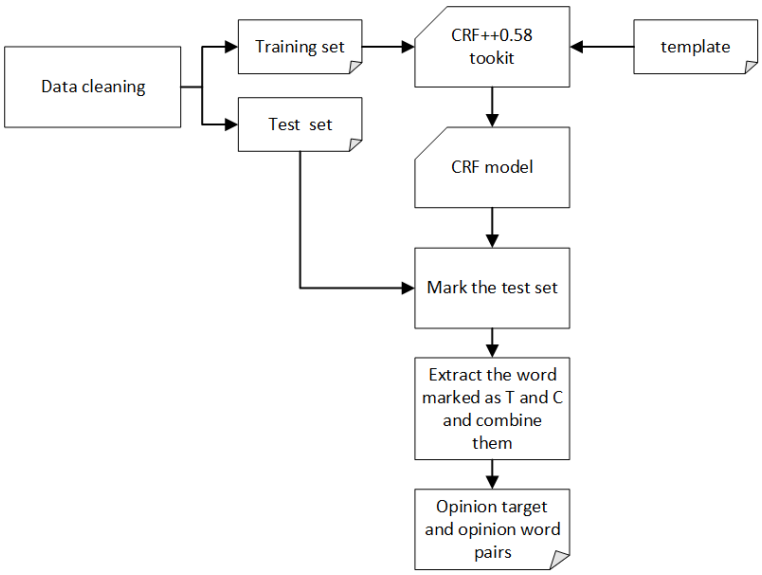

Figure 3. Opinion target extraction flowchart.

4. Experiments Results and Analysis

4.1. Dataset and Metrics

We use SemEval-2014 https://alt.qcri.org/semeval2014 dataset, which includes 3452 reviews on laptops and restaurants, to conduct our experiments. The description of the dataset is shown in Table 5.

Table 5. Description of the dataset.

Dataset	The Number of Sentences	The Size of the Training Set	The Size of the Test Set
Restaurant	2000	1655	345
Laptop	1452	1093	359

The accuracy (A), precision (P), recall (R) and F_1-score (F_1) were used to evaluate the performance of the extraction results. A, P, R and F_1 were computed as follows:

$$A = \frac{TP + TN}{TP + FP + TN + FN} \tag{3}$$

$$P = \frac{TP}{TP + FP} \tag{4}$$

$$R = \frac{TP}{TP + FN} \tag{5}$$

$$F_1 = \frac{2PR}{P + R} \tag{6}$$

where TP, FP, TN and FN represent the numbers of true positives, false positives, true negatives and false negatives, respectively.

4.2. Analysis of Results

We conducted our experiment with five templates on two domain datasets. All the code for this experiment are available on GitHub https://github.com/liucheng594/CRF-Evaluation-target-extraction. Besides, we compared our model with the following methods: RNCRF [12], a joint model based on RNN and CRF, proposed by Wang. CMLS [24], a joint model of multi-layer attentions proposed by Wang. SpanMlt [17], a multi-task model with a span generator and BERT–BiLSTM encoder, proposed by Zhao. LOTN [25], a model consists of PE-BiLSTM and attention-based BiLSTM network, proposed by Wu.

Table 6. Expriment results campared with other methods on two domain datasets.

Method	Restaurant				Laptop			
	A	P	R	F1	A	P	R	F1
RNCRF	-	-	-	84.52	-	-	-	78.93
CMLS	-	-	-	84.19	-	-	-	78.99
SpanMlt	-	-	-	85.70	-	-	-	82.56
LOTN	-	84.00	82.52	82.21	-	77.08	67.62	72.02
Template1$_{(ours)}$	96.92	90.02	89.16	89.58	96.45	87.39	81.10	84.12
Template2$_{(ours)}$	96.87	90.19	88.52	89.36	96.40	87.02	80.98	83.89
Template3$_{(ours)}$	96.39	88.17	87.41	87.79	96.17	86.17	79.76	82.84
Template4$_{(ours)}$	96.44	88.50	87.41	87.95	96.21	86.41	79.88	83.01
Template5$_{(ours)}$	**97.14**	**90.67**	**90.00**	**90.33**	**96.95**	**89.32**	**83.66**	**86.40**

Compared with other models based on deep learning methods in Table 6, our model obtained better results. The main reasons were as follows: (1) we introduced proper features such as parts of speech, syntactic relationships, etc., which meant that we gave sufficient text information to the model; (2) we made great efforts to define and test different templates and at last found proper parameters and almost the best suitable window size for each feature.

Compared the results of the five templates defined in Table 3 , we can conclude the following: (1) although the window sizes of Template 1 and Template 2 were smaller than those of Template 3 and Template 4, the extraction performance of the former were better than the latter, which was due to the excessive interference caused by the increase of noise for an overly large window size; (2) compared to Template 1, the performance of Template 2 was inferior after adding the dependency syntactic relationship feature. Conversely, comparing Template 3 with Template 4, the performance of Template 4 was better than Template 3 after adding the same feature, which indicated that the dependency syntax relationship feature required a larger window to achieve a better effect.

Based on the above experimental results, we reduced the word and part of speech feature window size and enlarged the dependency syntactic relationship feature window size and then obtained Template 5, which was better than other templates.

5. "Opinion Target: Opinion Word" Pair System Development

5.1. System Development Process

We adopted the conventional c/s model for system development, which is designed to process English comment sentences, extract opinion targets and opinion words and then visualize them. The development environment was IDEA, and the development language was JavaScript. The system flowchart is shown in Figure 4.

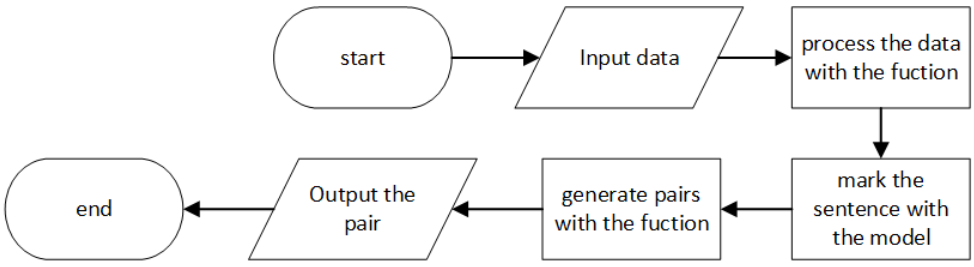

Figure 4. System flowchart.

5.2. Front-End Design

The front-end development of this paper used vue-cli3+elementUI. There are two pages in total. The Home page shows the instructions for using the system, and the Sentiment page is the core page of the system. Users can input a comment sentence into

the system and get the "opinion target: opinion word" pair. The detailed realization of the two pages is as follows.

The Home page uses the carousel component of element UI to show the appearance and structure of the system in pictures. The components can be played automatically, or users can use the mouse to select content to play. At the top of the Home page is a router connection for users to switch between the Home page and the Sentiment page.

At the top of the Sentiment page is a router connection that is the same as the Home page for users to switch pages. Below the option bar are two text boxes. The overall layout of the text box is constituted by two el-containers: the left text box is bound to the data entered through the v-model, and the right text box is used to display the analysis results. At the bottom right is a "start" button for transmitting the data entered by the front-end to the back-end and to display the results in the right text box. The page is shown in Figure 5.

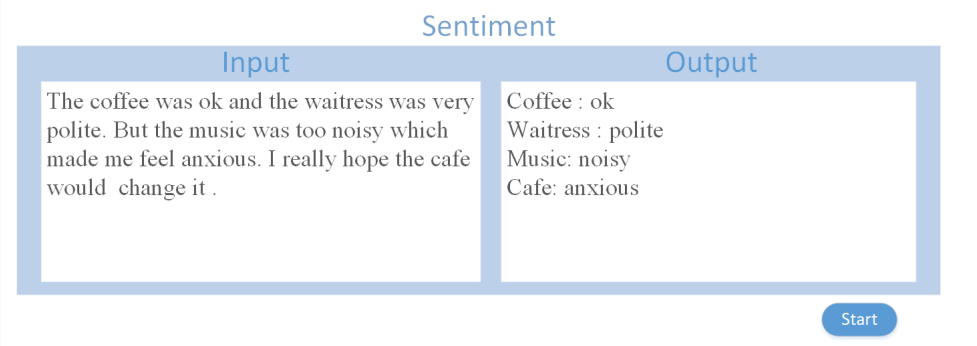

Figure 5. Sentiment page with an example sentence and corresponding "opinion target:opinion word" pair.

5.3. Back-End Design

We used Node.js + express to build the back-end, and the entire back-end system is a model analysis module. The front-end and the back-end interact through the axios library. The data entered by the user at the front-end can be processed into the format required by CRF++ with the script program for data cleaning mentioned above. Then the "opinion target: opinion word" pairs of the entire sentence are extracted with the trained model, and the results are finally displayed in the front-end. The specific processes are as follows:

First, the entered comment is submitted to the back-end through the axios.post function. The back-end server receives the data provided by the front-end through express.router and then the data are written to a local file through the fs file system. Second, the data cleaning script and the trained model are run through the node-cmd function to obtain marked sentences. Third, the extraction and matching script are used to extract the opinion target and opinion word and then match them into pairs. The result is written into the final.data file. Finally, the fs file system is used to obtain the data from the file, and the split function is used in the front-end to convert it into an array and display it in the container.

It is necessary to pay attention to asynchronous errors during data processing and extraction. Node.js uses asynchronous operations by default. When processing the data in the front-end, because the steps are time-consuming and the data remaining in the previous analysis need to be cleaned, we convert the asynchronous operation to a synchronous operation.

6. Conclusions

In this paper, we trained a model for "opinion target: opinion words" pair extraction based on the CRF and tested it using the SemEval-2014 Restaurant and Laptop datasets. Our experimental results demonstrate that it is effective to introduce part of speech and syntactic relationship features to extract opinion targets. We also found that the window size needs to be adapted to the features to extract opinion words. In addition, we developed

a system that can display "opinion target: opinion words" pairs in a front-end page. While the model has been built, there are still some shortcomings; furthermore, the deep learning-based methods have great potential, although they did not achieve good performance. This is what we will focus on.

Author Contributions: Conceptualization, Y.Y. and F.Z.; methodology, Y.Y. and F.Z.; software, C.L. and Y.G.; validation, Y.G and J.F.; formal analysis, Y.G and J.F.; investigation, C.L. and J.F; resources, C.L. and J.F; data curation, Y.G and J.F.; writing–original draft preparation, Y.Y. and Y.G.; writing–review and editing, Y.Y. and Y.G.; visualization, C.L. and Y.G; supervision, Y.Y and F.Z.; project administration, Y.Y. and F.Z.; funding acquisition, Y.Y. and F.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This work was partly supported by the Fundamental Research Funds for the Central Universities (8000150A082) by Y.Y.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Li, S.; Zhou, L.; Li, Y. Improving aspect extraction by augmenting a frequency-based method with web-based similarity measures. *Inf. Process. Manag.* **2015**, *51*, 58–67. [CrossRef]
2. Ahmad Rana, T.; Cheah, Y.-N. A two-fold rule-based model for aspect extraction. *Expert Syst. Appl.* **2017**, 89, 273–285. [CrossRef]
3. Wang, J.; Peng, Y.; Lin, Y.; Wang, K. Template Based Industrial Big Data Information Extraction and Query System. Data Mining and Big Data - Second International Conference(DMBD). In *International Conference on Data Mining and Big Data*; Springer: Cham, Switzerland, 2017; pp. 247–254.
4. Liu, Q.; Gao, Z.; Liu, B.; Zhang, Y. Automated Rule Selection for Aspect Extraction in Opinion Mining. In Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, Buenos Aires, Argentina, 25–31 July 2015; pp. 1291–1297.
5. Jochim, C.; Deleris, L.A. Named Entity Recognition in the Medical Domain with Constrained CRF Models. In Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers, Valencia, Spain, 3–7 April 2017; pp. 839–849.
6. Zhou, Y.; Jiang, W.; Song, P.; Su, Y.; Guo, T.; Han, J.; Hu, S. Graph Convolutional Networks for Target-oriented Opinion Words Extraction with Adversarial Training. In Proceedings of the 020 International Joint Conference on Neural Networks (IJCNN), Glasgow, UK, 19–24 July 2020; pp. 1–7.
7. Chutmongkolporn, K.; Manaskasemsak, B.; Rungsawang, A. Graph-based opinion entity ranking in customer reviews. International Symposium on Communications and Information Technologies. In Proceedings of the 2015 15th International Symposium on Communications and Information Technologies (ISCIT), Nara, Japan, 7–9 October 2015; pp. 161–164.
8. Ansari, G.; Saxena, C.; Tanvir Ahmad, M.; Doja, N. Aspect Term Extraction using Graph-based Semi-Supervised Learning. *Procedia Comput. Sci.* **2020**, *167*, 2080–2090. [CrossRef]
9. Samha, A.K.; Li, Y.; Zhang, J. Aspect-Based Opinion Mining from Product Reviews Using Conditional Random Fields. In Proceedings of the 13th Australasian Data Mining Conference, Sydney, Australia, 8–9 August 2015; pp. 119–128.
10. Hu, K.; Ou, Z.; Hu, M.; Feng, J. Neural CRF Transducers for Sequence Labeling. In Proceedings of the ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Brighton, UK, 12–17 May 2019; pp. 2997–3001.
11. Yin, Y.; Wei, F.; Li, D.; Xu, K.; Zhang, M.; Zhou, M. Unsupervised Word and Dependency Path Embeddings for Aspect Term Extraction. *arXiv* **2016**, arXiv:1605.07843.
12. Wang, W.; Pan, S.J.; Dahlmeier, D.; Xiao, X. Recursive Neural Conditional Random Fields for Aspect-based Sentiment Analysis. *arXiv* **2016**, arXiv:1603.06679.
13. Al-Smadi, M.; Talafha, B.; Al-Ayyoub, M.; Jararweh, Y. Using long short-term memory deep neural networks for aspect-based sentiment analysis of Arabic reviews. *Int. J. Mach. Learn. Cybern.* **2019**, *10*, 2163–2175. [CrossRef]
14. Hu, J.; Zheng, X. Opinion Extraction of Government Microblog Comments via BiLSTM-CRF Model. Joint Conference on Digital Libraries. In Proceedings of the ACM/IEEE Joint Conference on Digital Libraries in 2020, Wuhan, China, 19–23 June 2020; pp. 473–475.
15. Contractor, D.; Patra, B.; Singla, P. Constrained BERT BiLSTM CRF for understanding multi-sentence entity-seeking questions. *Nat. Lang. Eng.* **2020**, *27*, 65–87. [CrossRef]
16. Huang, Z.; Xu, W.; Yu, K. Bidirectional LSTM-CRF Models for Sequence Tagging. *arXiv* **2015**, arXiv:1508.01991.
17. Zhao, H.; Huang, L.; Zhang, R.; Lu, Q. SpanMlt: A Span-based Multi-Task Learning Framework for Pair-wise Aspect and Opinion Terms Extraction. ACL. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Florence, Italy, 5–10 July 2020; pp. 3239–3248.
18. Lafferty, J.; McCallum, A.; Pereira, F.C. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In Proceedings of the International Conference on Machine Learning (ICML), Williamstown, MA, USA, 28 June–1 July 2001; pp. 282–289.

19. Baksi, R.P.; Upadhyaya, S.J. Decepticon: A Hidden Markov Model Approach to Counter Advanced Persistent Threats. *Commun. Comput. Inf. Sci.* **2019**, *1186*, 38–54.
20. Alzaidy, R.; Caragea, C.; Giles, C.L. Bi-LSTM-CRF Sequence Labeling for Keyphrase Extraction from Scholarly Documents. In Proceedings of the World Wide Web Conference, San Francisco, CA, USA, 13–17 May 2019; pp. 2551–2557.
21. Jakob, N.; Gurevych, I. Extracting Opinion Targets in a Single and Cross-Domain Setting with Conditional Random Fields. In Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing, San Francisco, CA, USA, 13–17 May 2010; pp. 1035–1045.
22. Tran, T.U.; Hoang, H.T.T.; Huynh, H.X. Aspect Extraction with Bidirectional GRU and CRF. In Proceedings of the 2019 IEEE-RIVF International Conference on Computing and Communication Technologies (RIVF), Danang, Vietnam, 20–22 March 2019; pp. 1–5.
23. Veyseh, A.P.B.; Nouri, N.; Dernoncourt, F.; Dou, D.; Nguyen, T.H. Introducing Syntactic Structures into Target Opinion Word Extraction with Deep Learning. *arXiv* **2020**, arXiv:2010.13378.
24. Wang, W.; Pan, S.J.; Dahlmeier, D.; Xiao, X. Coupled Multi-Layer Attentions for Co-Extraction of Aspect and Opinion Terms. In Proceedings of the AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 4–9 February 2017; pp. 3316–3322.
25. Wu, Z.; Zhao, F.; Dai, X.Y.; Huang, S.; Chen, J. Latent Opinions Transfer Network for Target-Oriented Opinion Words Extraction. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020; pp. 9298–9305.