Knowledge-Based Systems 148 (2018) 66–73

Contents lists available at ScienceDirect

Knowledge-Based Systems

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

A hybrid unsupervised method for aspect term and opinion target extraction

Chuhan Wu[a,*], Fangzhao Wu[b], Sixing Wu[a], Zhigang Yuan[a], Yongfeng Huang[a]

[a] *Department of Electronic Engineering, Tsinghua University, Beijing, China*
[b] *Microsoft Research Asia, China*

ARTICLE INFO

Article history:
Received 23 July 2017
Revised 18 January 2018
Accepted 20 January 2018
Available online 31 January 2018

Keywords:
Aspect extraction
Opinion target extraction
Unsupervised learning
GRU

ABSTRACT

Aspect term extraction (ATE) and opinion target extraction (OTE) are two important tasks in fine-grained sentiment analysis field. Existing approaches to ATE and OTE are mainly based on rules or machine learning methods. Rule-based methods are usually unsupervised, but they can't make use of high level features. Although supervised learning approaches usually outperform the rule-based ones, they need a large number of labeled samples to train their models, which are expensive and time-consuming to annotate. In this paper, we propose a hybrid unsupervised method which can combine rules and machine learning methods to address ATE and OTE tasks. First, we use chunk-level linguistic rules to extract nominal phrase chunks and regard them as candidate opinion targets and aspects. Then we propose to filter irrelevant candidates based on domain correlation. Finally, we use these texts with extracted chunks as pseudo labeled data to train a deep gated recurrent unit (GRU) network for aspect term extraction and opinion target extraction. The experiments on benchmark datasets validate the effectiveness of our approach in extracting opinion targets and aspects with minimal manual annotation.

1. Introduction

Aspect term extraction (ATE) and opinion target extraction (OTE) are very useful to mine fine-grained sentiment information [1,2]. ATE aims to identify terms related to the aspects in the given domain [1], while OTE mainly intends to identify the targets associated with opinions [2]. OTE and ATE tasks are different because ATE focuses more on extracting the aspects of entities, rather than the entities themselves. An example is given in Fig. 1 to show such difference. The words "computer" and "battery" are both opinion targets. But since "computer" is an entity, it's not an aspect in the laptop domain. Both tasks are important for mining the fine-grained opinions [3,4]. There have been many methods proposed to deal with ATE and OTE tasks based on linguistic rules and supervised learning methods [5]. Qiu et al. proposed a rule-based method named double propagation (DP) [6]. DP algorithm extracts targets and opinion words jointly. In the extraction process, new opinion words and the associated opinion targets can be identified by direct and indirect dependency relations. The advantage of rule-based approaches such as DP is they are unsupervised and free from expensive manual annotation. However, they can't exploit the high level linguistic information to extract the aspect terms associated with neutral or implicit sentiment. There are also many supervised learning approaches to ATE and OTE. For example, the top system DLIREC in the SemEval-2014 task 4 competition[1] is based on CRF for ATE task [7]. This system mainly depends on a set of selected features like POS tags, head words and dependency relations to train a token-level CRF model. The top system EliXa in the SemEval-2015 task 12[2] is based on a perceptron for OTE task [8]. This system uses three different word clusters as customized features to train the perceptron-based OTE model. However, these approaches heavily rely on labeled data and the manually designed features. There are also several supervised approaches based on deep learning. For example, different neural networks have been successfully applied to ATE task like recurrent neural networks (RNN) [9], long short-term memory (LSTM) networks [9] and convolutional neural network (CNN) [10]. However, the main weakness of these supervised approaches is that they also require a large amount of labeled data for training. It's usu-

* Corresponding author.
E-mail addresses: wuch15@mails.tsinghua.edu.cn (C. Wu), wufangzhao@gmail.com (F. Wu), wu-sx15@mails.tsinghua.edu.cn (S. Wu), yuanzg14@mails.tsinghua.edu.cn (Z. Yuan), yfhuang@mail.tsinghua.edu.cn (Y. Huang).

[1] http://alt.qcri.org/semeval2014/task4/.
[2] http://alt.qcri.org/semeval2015/task12/.

https://doi.org/10.1016/j.knosys.2018.01.019
0950-7051/© 2018 Published by Elsevier B.V.

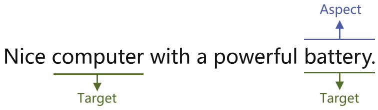

Fig. 1. Examples of OTE and ATE tasks.

ally time-consuming and expensive to manually annotate sufficient fine-grained samples for training [11].

In this paper, we propose a hybrid unsupervised method for ATE and OTE tasks. Since a target or aspect is usually a noun or a noun phrase, extracting it at chunk-level may be easier to identify complex phrases. Thus, we propose a set of rules at chunk-level to extract and prune nominal phrase (NP) chunks to get candidate opinion targets and aspects. Then we propose a method by using domain correlation based on word embedding to filter irrelevant candidates. Finally, these candidates are used to generate pseudo labeled data to train a deep GRU network for better prediction. Our method can utilize rules to generate reasonable predictions, and use deep neural networks to learn a higher level representation for better extraction performance. Thus it can combine the advantages of rules-based methods and supervised learning-based methods. The experiments on the benchmark datasets show the effectiveness of our approach. The contributions of our work are listed as follows:

- We propose a chunk-level extraction method which utilizes a set of rules based on dependency relations to extract NP chunks as candidate opinion targets and aspects.
- We propose to filter noisy candidates using a domain correlation measurement based on word embedding.
- We propose to use texts with candidate chunks as pseudo labeled data to train DNN models for ATE and OTE using a deep GRU architecture.

The remaining of this paper is organized as follows. In Section 2 we introduce related works on ATE and OTE. In Section 3 we present the details of our unsupervised method. The experiment settings and results are presented in Section 4. Finally, we conclude this paper in Section 5.

2. Related work

Existing approaches to ATE and OTE tasks are mainly based on rules or supervised learning methods. Rule-based methods usually rely on lexicons and dependency relations of sentences [6,12–23]. Hu and Liu [12] proposed a series of rules to extract opinions and frequent or infrequent features based on their dependency relations. Several extensions of this method are proposed by Popescu et al. [13] and Blair-Goldensohn et al. [15]. Qiu et al. [6] proposed a method called double propagation (DP). Their algorithm can jointly extract opinion words and aspects based on direct or indirect dependency relations. In their method, nouns associate with opinion words or other opinion targets can be extracted as new opinion targets. These approaches are unsupervised and free from expensive manual annotation. However, these rule-based methods can't utilize high level linguistic information, which makes them hard to extract the terms associated with neutral or implicit sentiment.

Supervised learning methods were successfully used in ATE and OTE tasks. By regarding them as sequential labeling tasks, hidden Markov model (HMM) [24] and conditional random field (CRF) [25–29] are employed to address these tasks. Usually such approaches need manual feature engineering to train their models. In SemEval-2014 [1], Toh and Wang [7] proposed a system named DLIREC using CRF with a set of designed features. These supervised methods have shown better performance than traditional unsupervised linguistic rule-based methods, but they heavily depend on the manually selected features and require enough annotated data for training.

Topic modeling methods such as Latent Dirichlet Allocation (LDA) and its variants are also widely used in aspect extraction and applied to construct aspect ontology for ATE and OTE tasks [30–35]. Typical LDA methods are based on constructing word frequency vectors of the context. In the extraction process, an aspect is selected by a multinomial distribution and a word is extracted by another multinomial distribution of the given aspect [36]. For example, a supervised approach based on seed aspect and LDA model was proposed by Mukherjee and Liu [37] to address ATE task. They use seed aspects to extract related product aspects from product reviews. Another semi-supervised model was proposed by Wang et al. [38]. They proposed two variants of LDA based on seed aspects and outperform the previous LDA models. However, these topic modeling methods can't mine high level linguistic and sentiment features from sentences. They usually also need supervised learning strategies to adapt to fine-grained ATE and OTE tasks.

In recent years, deep learning methods have shown better performance than CRFs in ATE and OTE tasks. For example, Liu et al. [9] compared the performance of different word embeddings and architectures of RNN in ATE task. Their approaches avoid manual feature engineering and mainly use POS tags and word embedding for model training. Poria et al. [10] proposed a new hybrid extraction method combines deep neural networks and linguistic patterns. Their approach used a deep CNN to predict the tag of each word and applies a set of rules based on syntax to refine predictions of CNN model. They successfully improved the performance of their CNN and outperform the top systems in SemEval-14 [1]. However, these supervised methods also require many labeled samples to train models, which need expensive and time-consuming manual annotation.

Compared with these related works, our approach differs from them significantly. First, since some opinion targets and aspects are complex phrases, we propose a set of rules to extract them at chunk-level. Second, we propose to filter the candidates by domain correlation based on word embedding weights. Finally, we propose to use the extracted candidates to generate pseudo labels of sentences to train a deep GRU network. Our method can mine useful information from raw texts by rules and learn a higher level representation by deep learning. Our method shows effectiveness in both ATE and OTE tasks, which is supported by the experimental results.

3. Unsupervised method for opinion target extraction and aspect term extraction

Our unsupervised method consists of three submodules. First, we extract and filter NP chunks to generate candidate opinion targets and aspects by chunk-level rules based on dependency relations. Second, we evaluate and filter these candidates using the domain correlation of extracted nouns. We also make additional pruning to adapt to ATE task in this process. Finally, we use the candidates to generate pseudo labels of sentences to train a deep GRU network. The structure of the system is shown in Fig. 2. We will introduce our method in detail in the following subsections.

3.1. Chunk extraction and filtering rules

Usually an aspect term or opinion target is a noun or noun phrase, and some phrases can be very complex such as “cod with pineapple tempura”. It may be easier to identify noun phrases by regarding them as a whole to extract at chunk-level. We use the Stanford Parser tool[3] to get POS tags and the dependency of an in-

[3] https://nlp.stanford.edu/software/lex-parser.shtml.

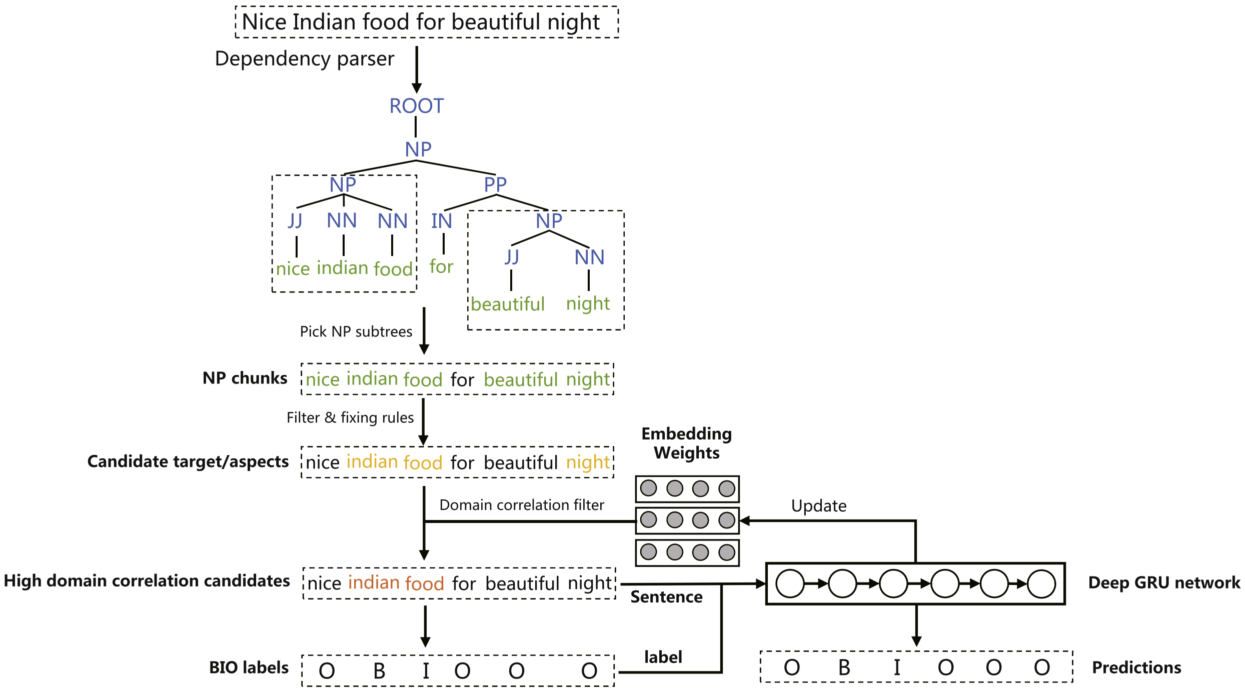

Fig. 2. Structure of our unsupervised method for ATE and OTE.

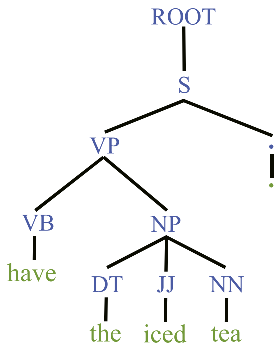

Fig. 3. An example of a dependency tree.

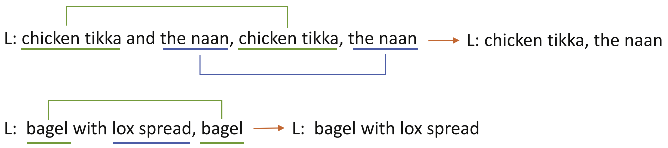
L: chicken tikka and the naan, chicken tikka, the naan
L: chicken tikka, the naan
L: bagel with lox spread, bagel
L: bagel with lox spread

Fig. 4. Examples of rule 2. The lines connect the same chunks and sub-chunks in *L*.

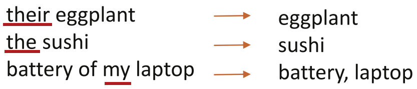

Fig. 5. Examples of rule 3.

Table 1
Rules to identify chunks modified by different expressions.

Rule	Example
Nouns modified by adjectives	nice **sushi**
Nouns modified by verbs	try the **lobster teriyaki**
Nouns modified by sentiment adverbs	well prepared **food**
Nouns coordinate with other nouns	**chicken tikka** and the **naan**

put sentence to construct a dependency tree [39]. An example of dependency tree is shown in Fig. 3. Since a 3-depth-subtree indicates a noun or simple noun phrase with its POS tag while a 4-depth one indicates a more complex noun phrase, we choose the subtrees with “NP” tagged root and 3 or 4 nodes’ depth. Then we get a list of extracted NP chunks and we denote it as *L*. We filter or fix them using the following rules.

1. For every NP chunks in *L*, we determine whether it should be reserved by rules shown in Table 1. We reserve a chunk if it contains nouns satisfying at least one of these rules, otherwise we filter this chunk.
   This set of rules are used to extract all possible opinion targets and aspects modified by different types of opinions. Different from DP [6], this rule can also extract the opinion targets and aspects associated with neutral or implicit sentiment. For example, the sentence “Try the lobster teriyaki!” doesn’t contain any affective words but it expresses a strong sentiment. In such case, DP will fail to extract the opinion targets but this set of rules can still extract the aspect “lobster teriyaki”.
2. If a chunk has conjunction or preposition and both sides of it are also in *L*, then we discard it, otherwise we remove the smaller ones. This rule is used to determine whether a chunk should be split when it has subchunks in *L*. An example of this rule is shown in Fig. 4.
3. For every chunk, remove articles and pronouns in it. If the removed word is in the middle of a chunk, then we break the chunk into two NP parts. Several examples are shown in Fig. 5.
4. For every chunk, we remove the adverbs, sentimental adjectives (including all comparative and superlative words), sentimental verbs, conjunctions and numerals in the front. As shown in Fig. 6, this rule is used to prune the extracted chunks to get expected opinion targets or aspects.
5. Finally we check if a chunk still has noun. If not, we filter it. The sentiment lexicon we use in the rules above is Bing Liu Opinion Lexicon provided by Hu and Liu [12]. This lexicon contains positive and negative opinion words and we use it mainly for handling opinion adverbs and prune extracted chunks.

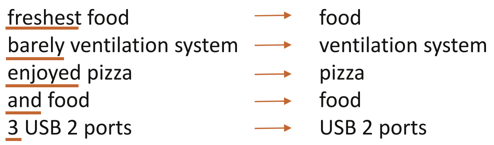
freshest food → food
barely ventilation system → ventilation system
enjoyed pizza → pizza
and food → food
3 USB 2 ports → USB 2 ports

Fig. 6. Examples of rule 4.

3.2. *Domain correlation evaluation and filter*

After the chunk-based NP extraction process, we get a candidate group of aspect terms and opinion targets of each sentence. But some of them are irrelevant to the domains. For example, “food” has a very high correlation with the restaurant domain while “height” is usually not an aspect in this domain. Therefore, we propose a method to evaluate the correlation between nouns and domains and determine whether the candidates should be filtered.

After getting the candidate opinion targets of each sentence, we collect every noun W_i in them and denote c_i as the frequency of it. Then we construct a set

$$S = \{W_i | c_i > T_h\}, \tag{1}$$

where T_h is a frequency threshold. We find the filtering results are not sensitive to the selection of T_h, and we set T_h to 5 in our experiments. Nouns with higher frequency are more likely to represent some of the semantic features of the task domain. In our experiment, the set S is constructed only on training set, except for testing on the cross-domain dataset.

Let e_i be the pretrained embedding weights of word W_i, then the domain correlation degree α_i of every noun W_i can be calculated by

$$\alpha_i = \frac{\sum_{W_j \in S} c_j < e_i, e_j >}{\sum_{W_j \in S} c_j}, \tag{2}$$

$$< e_i, e_j > = \frac{e_i \cdot e_j}{|e_i||e_j|}, \tag{3}$$

where $< e_i,\ e_j >$ means the cosine similarity between e_i and e_j. High α_i means that W_i has a high probability to be an aspect or opinion target in a specific domain. We sort all the nouns in the collected W_i set by α_i in descending order and divide it into two parts by ratio β. The lower α_i part is called “irrelevant noun set” I. If all nouns of a candidate target/term are in I, then it is filtered. In our experiment the filter threshold β is set to 0.5, therefore only nouns with higher correlations are used. For example, in the sentence “This restaurant meets my expectations”, the word “restaurant” will be reserved in our OTE experiment ($\alpha_i = 0.6137$) and “expectations” will be filtered ($\alpha_i = 0.5302$).

Additional fix for ATE

Since the ATE task focuses more on extracting the aspects of entities than OTE task, in order to work on both tasks together, we select several words, like the name of domain (e.g. restaurant, laptop) and several names of entities crawled from the google search engine (mac, hp etc.) as stop words in ATE task. Then we merge the stop words set into the “irrelevant noun set” to apply to the filter.

3.3. *Train GRU models*

Since deep neural networks can learn high-level linguistic and sentiment information, they can be used to make better predictions in ATE and OTE tasks. These two tasks can be formulated as sequential labeling problems. In network training, the encoded BIO labels of each input sentence are actually the extracted opinion targets or aspects given by the first two modules. Therefore, the training is free from manual annotation. After trained for several epoches, we use the fine-tuned embedding weights to update the weights in the second module. Then the second module regenerate labels and warm-setup the network’s training. The training details are as follows:

Table 2
An example of BIO encoding.

Kind	,	attentive	wait	staff	.
O	O	O	B	I	O

3.3.1. *Generate pseudo labels for sentences*

After obtaining the filtered candidate opinion targets or aspects, we use them to generate pseudo labels of sentences for network training. The labeling is based on begin-inside-outside (BIO) encoding. BIO encoding is widely used in sequence processing. The annotation and evaluation in our tasks are also based on BIO encoding. For example, Table 2 shows the BIO tags for a training sentence in the restaurant domain.

3.3.2. *Word embedding*

In our model, the embedding of each word is a v_1-dim vector. We use the Google embedding proposed by Mikolov et al. [40]. They proposed two models, one is called the CBOW model and the other is called the skip-gram model. We use the pre-trained embedding weights released by them which were trained by the skip-gram model on about 100-billion words on Google News.[4]

3.3.3. *Network architecture*

The architecture of the deep GRU network used in our system is shown in Fig. 7. We follow the structure of GRU cells proposed by Cho et al. [41] in our network. A sentence with its POS tags of each word is the input for the network. Different from using one-hot encoding of POS tags in others’ work, we embed POS tags into v_2-dim vectors as additional features. We apply dropout on both embedding layers to avoid over-fitting. Then the two tensors are merged into $(v1 + v2)$-dim as the input for a Bi-GRU layer. The Bi-GRU layer has two outputs with $2v_3$ dims of hidden states. The middle GRU layer outputs with v_4 dims of tensors. We concatenate the two flows to increase dimensions directly instead of adding them together. The top GRU layer fit the space-residual of them and outputs the softmax classification result of each word. The probability of k-th type of tag is calculated by

$$p_k = \frac{exp(h_k)}{\sum_j exp(h_j)}. \tag{4}$$

The word will be tagged with k if the prediction p_k is the biggest.

4. Experiment

4.1. *Experiment settings*

4.1.1. *Dataset*

We use the restaurant and laptop review dataset in SemEval-2014[5] for ATE task. The datasets for OTE task contain restaurant and hotel reviews in SemEval-2015[6] and the restaurant reviews in SemEval-2016.[7] Since these OTE datasets are not large enough and the ATE annotation criterion is different from OTE, we also manually re-annotated SemEval-2014 datasets for OTE task to fully test

[4] https://code.google.com/archive/p/word2vec/.

[5] http://alt.qcri.org/semeval2014/task4/index.php?id=data-and-tools.

[6] http://alt.qcri.org/semeval2015/task12/index.php?id=data-and-tools.

[7] http://alt.qcri.org/semeval2016/task5/index.php?id=data-and-tools.

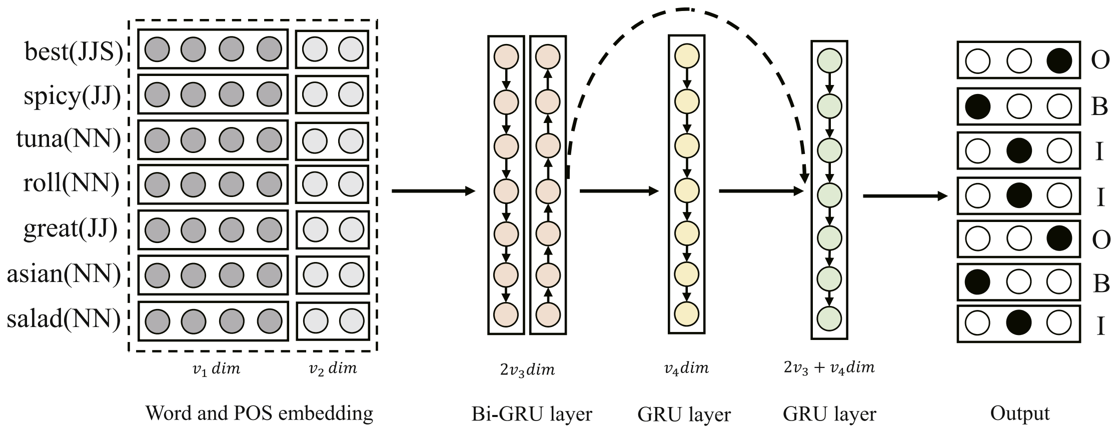

Fig. 7. Structure of our deep GRU network.

Table 3
Statistics of the datasets.

Task	SemEval-14		SemEval-15		SemEval-16
	ATE		OTE		
Domain	Restaurant	Laptop	Restaurant	Hotel	Restaurant
Training	3041	3045	1315	–	2000
Test	800	800	685	339	676
Total	3841	3845	2000	339	2676

our model. A typical example to show such annotation criterion difference is "I enjoy having apple products". It has no aspect but has a target "apple products". The statistics of these datasets are shown in Table 3.

4.1.2. Metrics

Metric. Three metrics are used in our experiments to evaluate the performance, i.e., precision, recall and F_1-score. Aspect terms or opinion targets are regarded to be correct if and only if they perfectly match the golden annotations. The precision, recall and F_1-score are computed as follows:

$$p = \frac{TP}{TP + FP}, \quad (5)$$

$$r = \frac{TP}{TP + FN}, \quad (6)$$

$$F_1 = \frac{2pr}{p + r}, \quad (7)$$

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

4.1.3. Word preprocess

Since the reviews contain many rare words including names, types and slangs, we preprocessed sentences as follows: First, if a word is a number, we replace it with "DIGIT". Second, if a word is mixed with digits and letters, we replace it with "TYPE". Finally if a word appears less than 3 times then we replace it with "UNKNOWN".

4.1.4. Parameter settings

In the deep GRU network, we follow several network settings provided by Liu et al. [9]. The word embedding dim v_1 is set to be 300, while the POS embedding dim v_2 is set to be 20. Therefore, the input dim for the GRU network is concatenate to be 320. The dim v_3 of hidden states in the Bi-GRU layer is set to 50. The output dim v_4 of the middle GRU layer is 20. The first training process lasts about 6–8 epoches and we warm setup the training for 3–4 epoches after updating the weights and labels.

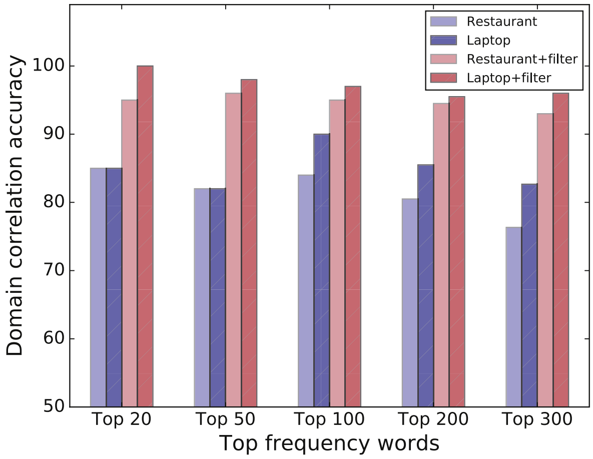

Fig. 8. Domain correlation accuracy of top frequency words.

4.2. Domain correlation evaluation result

In this subsection we present the experimental results of domain correlation evaluation module. The most and least correlative words in two domains are shown in Table 4. We can see that the top relevant words like "meal" and "food" have high correlation with the restaurant domain, while the top irrelevant words such as "amex" and "soh" are not related to the domain. In order to evaluate our method quantificationally, we invite a group of volunteers to evaluate the domain correlation accuracy of top frequency words before and after filtering. As shown in Fig. 8, we find that our method can identify irrelevant words effectively even with high word frequencies. For example, "money" has a very high frequency but it's not so relevant to the laptop domain. This result indicates that our evaluation method can mine domain information well and don't much rely on the word frequencies.

4.3. Performance evaluation

4.3.1. Performance in aspect term extraction task

The result of ATE task on the SemEval-2014 dataset is shown in Table 5. The baseline deep networks are trained on 2000 samples while the system DLIREC is trained on over 3000 samples. Our method is free from labeled data and can reach quite comparable performance with the deep networks trained on 2000 la-

Table 4
Top high and low correlation words in the restaurant and laptop domains.

Restaurant	High	meal,food,dessert,sushi,pizza,meals,sandwiches,burger, buffet,salad,desserts,soup,cuisine,sandwich
	Low	amex,soh,facie,og,ley,prima,modeling,echoes, frederick,attorney,3×,×4,achilles,seine,convert,nex,hurley
Laptop	High	lenovo,hardrive,macbook,powerbook,laptop,harddrives,desktop, osx,4gb,macbooks,bestbuy,computer
	Low	member,handed,statement,head,recalled,members,consultant, colleague,opposite,associate,speaking

Table 5
Results of aspect term extraction task. The (C) in the table means constrained and (U) means unconstrained [7].

Systems	Restaurant			Laptop		
	p	r	F_1	p	r	F_1
CRF	78.30	71.60	74.80	62.07	57.80	59.86
LSTM	75.70	75.84	75.78	63.87	61.62	62.72
Bi-LSTM	79.23	78.04	78.63	68.52	56.57	61.98
Deep LSTM	74.35	81.31	77.67	65.76	62.84	64.27
Double Propagation (DP)	37.48	47.27	41.81	31.03	27.52	29.17
Linguistic patterns on SenticNet	65.41	60.50	62.86	62.39	57.20	59.68
DLIREC(C)	84.04	73.37	78.34	79.31	63.30	70.41
DLIREC(U)	85.35	82.72	84.01	81.90	67.13	73.78
Our unsupervised system	72.81	79.81	76.15	55.91	66.51	60.75

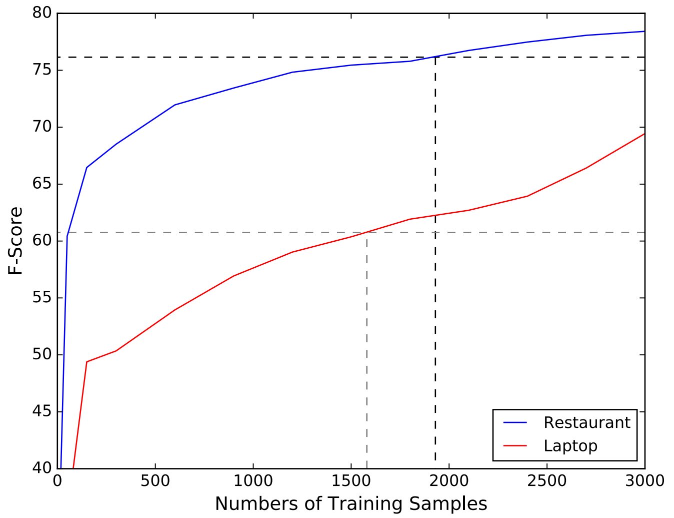

Fig. 9. *F*-Score performance of a single GRU layer trained by different volume of labeled data. The black and grey lines represent the level of our unsupervised method.

bel sentences. It indicates that our method can significantly reduce the dependency on labeled data and cut down the cost of manual annotation. Compared with other unsupervised methods like DP, our method outperforms them significantly. Our method can extract aspects associate with implicit sentiment. For example, the aspect “USB3” in the sentence “Having USB3 is why I bought this Mini” can be successfully extracted by our model. It shows that our method can exploit high level linguistic features by using deep networks.

In addition, we compare the performance of our unsupervised method with a single GRU layer. The dim of its hidden states is set to 100. We train it with different numbers of labeled sentences to compare the F-score performance with our approach. As shown in Fig. 9, the black and grey lines show the performance of our method. In our experiments, the performance of our method equals with the GRU layer trained on about 1900 sentences in the restaurant domain and near 1600 sentences in the laptop domain. These results validate that our method can save much labeled data and free users from heavy annotation tasks.

Table 6
Result for opinion target extraction in SemEval-2015 and SemEval-2016. The constrained and unconstrained results are provided by SemEval [2,42].

Systems	Semeval-15			Semeval-16		
	p	r	F_1	p	r	F_1
Unconstrained best	68.93	71.22	70.05	75.49	69.44	72.34
Constrained best	67.23	66.61	66.91	74.12	60.95	66.91
CRF	47.80	58.23	52.50	60.67	65.52	63.00
LSTM	66.11	58.41	62.02	71.33	67.48	69.35
Bi-LSTM	68.25	53.23	59.81	72.97	61.76	66.90
Deep LSTM	65.59	63.77	64.67	70.59	69.45	70.02
Ours	62.26	65.13	63.36	58.94	70.59	64.24

Table 7
Result for OTE on our re-annotated SemEval-2014 testing sets.

	Restaurant			Laptop		
	p	r	F_1	p	r	F_1
Double Propagation (DP)	38.90	48.04	42.99	39.49	44.38	41.79
Ours	**72.17**	**73.43**	**72.80**	**64.29**	**74.02**	**68.81**

Table 8
Performance comparison of different models in cross-domain OTE task.

Model	Hotel		
	p	r	F_1
CRF	48.52	76.64	59.42
LSTM	50.51	70.09	58.71
Bi-LSTM	52.06	64.95	57.80
Deep LSTM	49.08	75.23	59.41
DP	39.68	58.41	47.26
Ours	**55.86**	**84.58**	**67.29**

4.3.2. Performance in opinion target extraction task

The results of OTE task on the SemEval-2015 and SemEval-2016 datasets are shown in Table 6. Our method outperforms the deep networks trained on about 1300 samples in SemEval-15 dataset, and reaches comparable results with networks trained on 2000 samples. Such results fully show that our method can reduce the dependency on labeled data. In order to fully test our model, we manually re-annotated the reviews testing set in SemEval-2014 for evaluation of opinion target extraction task. The result can be seen in Table 7. Our method outperforms DP significantly, which proves our method can mine high level linguistic features by using deep networks to give better predictions in OTE task.

4.3.3. Result in cross-domain opinion target extraction

We compare the cross-domain (hotel domain) performance of supervised methods trained by labeled data in the restaurant domain. From Table 8, we can see that supervised methods like deep networks have poor performance, since hotel domain has very different features from the restaurant domain. It may indicate that these supervised methods heavily rely on the in-domain labeled data. Our unsupervised method performs well and shows good portability in such new domain, which is promising when facing a new domain without labeled data.

Table 9
Performance of sub modules and processes.

Domain	Restaurant						Laptop					
Submodule	ATE			OTE			ATE			OTE		
	p	r	F_1	p	r	F_1	p	r	F_1	p	r	F_1
Chunk-based extraction	54.28	77.16	63.73	54.73	**76.01**	63.64	47.02	**67.58**	55.45	58.93	**74.12**	65.66
+correlation filter	63.33	72.49	67.60	64.51	71.16	67.67	53.26	61.16	56.94	**65.21**	69.21	67.15
++deep GRU	72.46	77.95	75.11	**73.28**	71.01	72.13	53.58	66.36	59.29	64.28	74.02	**68.81**
++update filter weights	**72.81**	**79.81**	**76.15**	72.17	73.43	**72.80**	**55.91**	66.51	**60.75**	63.92	66.80	65.33

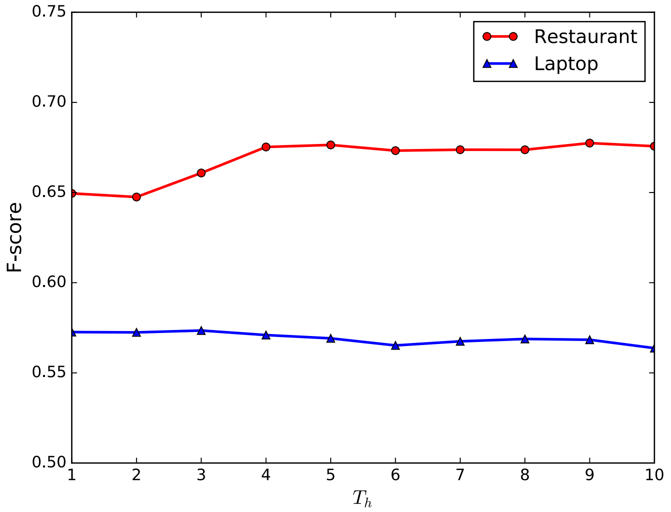

Fig. 10. *F*-score performance after domain correlation filtering with different threshold T_h.

4.3.4. *Result of sub modules*

The effects of each submodule of our system are shown in Table 9. We can see that our system performs better in the OTE task. Our system reaches a high recall after chunk-based extraction and filter, but the precision still needs to be improved. We can see that the domain correlation filter can improve the precision and F_1 score significantly with a minor loss of recall. It shows that the domain correlation evaluation method is efficient for both tasks. In addition, due to the feature extraction ability of the deep GRU network, the result is also improved in both domains. The improvement made by the network is very significant on the two tasks in the restaurant domain and the ATE task in the laptop domain. We update the embedding weights using the saved model and warm-setup the training with the regenerated annotation. The improvement can be seen but not so significant. The performance on the last task drops and it shows that the network may start to overfit the noisy training data generated by the previous processes.

4.3.5. *Sensitivity of threshold selection*

We test the F-score performance of the domain correlation submodule setting with different threshold T_h. The result is shown in Fig. 10. When we set the threshold $T_h \geq 4$, we can see that the performance is not sensitive to T_h in both domains. Thus, our selection ($T_h = 5$) is appropriate. An exception is when T_h is too small, the performance in the restaurant domain is not optimal. It may due to the infrequent nouns such as names of restaurants. For example, the name “Harumi” in “Harumi Sushi” is not so related to the restaurant domain. Such words are noisy and may hinder the filter.

Table 10
Time and space complexity of our proposed model.

Time		
Process	Theoretical	Actual/min
Sentence parsing	$O(n)$	**154**
Rules extraction	$O(n)$	0.01
Domain correlation	$O(n+v_1 f(n)^2)$	0.7
Network training	$O(n)$	5
Space		
Process	Theoretical	Actual/MB
Sentence parsing	$O(1)$	26
Rules extraction	$O(d)$	<1
Domain correlation	$O(n+v_1 f(n))$	70
Network training	$O(n)$	**450**

4.3.6. *Evaluation of time and space complexity*

The complexity of our model is shown in Table 10. Note that both actual running time and memory consumption are evaluated on the restaurant training set. We assume that the max length of sentences is given. Thus, the total time complexity of parsing is $O(n)$ and the space complexity is $O(1)$. Firstly, we convert all lexicons into a hash table. Therefore, the total query operation time is $O(n)$ with $O(d)$ space, where d is the volume of external lexicons. Since the extraction of one sentence need $O(1)$ time and space, the total consuming is $O(n)$ and $O(d)$. Next in domain correlation evaluation, the complexity depends on nouns in the candidate set. We assume that the amount of noun types can be approximated by a function $f(n)$. Although we can’t give a representation of $f(n)$, we can ensure the increasing of $f(n)$ is far behind n. Since we need to evaluate the cosine distance of each other, the time and space consumption is $O(n+v_1 f(n)^2)$ and $O(n+v_1 f(n))$ respectively, where v_1 is the word embedding size. Finally in network training, the model is given and the complexity is $O(n)$. According to the actual time and memory used, we find that the sentence parsing takes the longest time while network training takes most memory. Thus compared with typical methods using neural networks, our hybrid method only uses a little more computing resources.

5. Conclusion

In this paper, we propose an unsupervised method for opinion target extraction task and aspect term extraction task. Our approach consists of three submodules. First, we propose a chunk-level extraction method to get candidate opinion targets and aspects. Next, we propose a domain correlation measurement to filter these candidates. Finally, we use these texts with extracted chunks as pseudo labeled data to train a deep GRU network for final predictions. Our method can utilize rules to obtain reasonable predictions and use deep network to learn a higher level representation. Our unsupervised method is free from manual annotation and combines the advantages of rules-based methods and supervised learning-based methods. The experimental results on several benchmark datasets show the effectiveness of our approach.

Acknowledgments

This research is supported by the Key Program of National Natural Science Foundation of China (Grant nos. U1536201 and U1405254),the National Natural Science Foundation of China (Grant no. 61472092).

References

[1] M. Pontiki, D. Galanis, J. Pavlopoulos, H. Papageorgiou, I. Androutsopoulos, S. Manandhar, Semeval-2014 task 4: aspect based sentiment analysis, Proc. SemEval (2014) 27–35.

[2] 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), Association for Computational Linguistics, Denver, Colorado, 2015, pp. 486–495. URL http://www.aclweb.org/anthology/S15-2082.

[3] N. Kobayashi, K. Inui, Y. Matsumoto, Extracting aspect-evaluation and aspect-of relations in opinion mining, in: Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, 2007, pp. 521–574.

[4] J. Jagarlamudi, I. Hal Daum, R. Udupa, Incorporating lexical priors into topic models, in: Conference of the European Chapter of the Association for Computational Linguistics, 2009, pp. 204–213.

[5] T.A. Rana, Y.N. Cheah, Aspect extraction in sentiment analysis: comparative analysis and survey, Artif. Intell. Rev. (2016) 1–25.

[6] G. Qiu, B. Liu, J. Bu, C. Chen, Opinion word expansion and target extraction through double propagation, Comput. Ling. 37 (1) (2011) 9–27.

[7] Z. Toh, W. Wang, Dlirec: aspect term extraction and term polarity classification system, in: Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014), Association for Computational Linguistics and Dublin City University, Dublin, Ireland, 2014, pp. 235–240. URL http://www.aclweb.org/anthology/S14-2038.

[8] I.n. San Vicente, X. Saralegi, R. Agerri, Elixa: a modular and flexible absa platform, in: Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015), Association for Computational Linguistics, Denver, Colorado, 2015, pp. 748–752. URL http://www.aclweb.org/anthology/S15-2127.

[9] P. Liu, S.R. Joty, H.M. Meng, Fine-grained opinion mining with recurrent neural networks and word embeddings., in: EMNLP, 2015, pp. 1433–1443.

[10] S. Poria, E. Cambria, A. Gelbukh, Aspect extraction for opinion mining with a deep convolutional neural network, Knowl. Based Syst. 108 (2016) 42–49.

[11] C. Quan, F. Ren, Unsupervised product feature extraction for feature-oriented opinion determination, Inf. Sci. 272 (C) (2014) 16–28.

[12] M. Hu, B. Liu, Mining opinion features in customer reviews, in: National Conference on Artifical Intelligence, 2004, pp. 755–760.

[13] A.-M. Popescu, B. Nguyen, O. Etzioni, Opine: Extracting Product Features and Opinions from Reviews, 2005, pp. 9–28.

[14] L. Zhuang, F. Jing, X.-Y. Zhu, Movie review mining and summarization, in: Proceedings of the 15th ACM International Conference on Information and Knowledge Management, ACM, 2006, pp. 43–50.

[15] S. Blair-Goldensohn, K. Hannan, R. McDonald, T. Neylon, G.A. Reis, J. Reynar, Building a sentiment summarizer for local service reviews, in: WWW Workshop on NLP in the Information Explosion Era, 14, 2008, pp. 339–348.

[16] S.S. Htay, K.T. Lynn, Extracting product features and opinion words using pattern knowledge in customer reviews, Sci. World J. 2013 (6) (2013) 394758.

[17] S. Poria, E. Cambria, L.-W. Ku, C. Gui, A. Gelbukh, A rule-based approach to aspect extraction from product reviews, in: Proceedings of the Second Workshop on Natural Language Processing for Social Media (SocialNLP), 2014, pp. 28–37.

[18] Z. Hai, K. Chang, J.J. Kim, C.C. Yang, Identifying features in opinion mining via intrinsic and extrinsic domain relevance, IEEE Trans. Knowl. Data Eng. 26 (3) (2014) 623–634.

[19] S.M.J. Zafra, M.T. Mart-Valdivia, E. Mart-nez-Cmara, L.A.U. Lpez, Combining resources to improve unsupervised sentiment analysis at aspect-level, J. Inf. Sci. 42 (2) (2015).

[20] O. Gunes, T. Furche, G. Orsi, Structured aspect extraction., in: COLING, 2016, pp. 2321–2332.

[21] Q. Liu, Z. Gao, B. Liu, Y. Zhang, Automated rule selection for opinion target extraction, Knowl. Based Syst. 104 (2016) 74–88.

[22] Q. Liu, B. Liu, Y. Zhang, D.S. Kim, Z. Gao, Improving opinion aspect extraction using semantic similarity and aspect associations., in: AAAI, 2016, pp. 2986–2992.

[23] T.A. Rana, Y.-N. Cheah, A two-fold rule-based model for aspect extraction, Expert Syst. Appl. 89 (2017) 273–285.

[24] W. Jin, H.H. Ho, R.K. Srihari, Opinionminer: a novel machine learning system for web opinion mining and extraction, in: Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2009, pp. 1195–1204.

[25] N. Jakob, I. Gurevych, Extracting opinion targets in a single- and cross-domain setting with conditional random fields, in: Conference on Empirical Methods in Natural Language Processing, 2010.

[26] B. Yang, C. Cardie, Extracting opinion expressions with semi-Markov conditional random fields, in: Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, Association for Computational Linguistics, 2012, pp. 1335–1345.

[27] M. Mitchell, J. Aguilar, T. Wilson, B. Van Durme, Open domain targeted sentiment, in: Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, 2013, pp. 1643–1654.

[28] A.K. Samha, Y. Li, J. Zhang, Aspect-based opinion mining from product reviews using conditional random fields, in: Data Mining and Analytics: Proceedings of the 13th Australasian Data Mining Conference [Conferences in Research and Practice in Information Technology, Volume 168], Australian Computer Society, 2015, pp. 119–128.

[29] O. Gunes, Aspect term and opinion target extraction from web product reviews using semi-markov conditional random fields with word embeddings as features, in: WIMS, 2016.

[30] I. Titov, R. McDonald, Modeling online reviews with multi-grain topic models, in: Proceedings of the 17th international conference on World Wide Web, ACM, 2008, pp. 111–120.

[31] C. Lin, Y. He, Joint sentiment/topic model for sentiment analysis, in: Proceedings of the 18th ACM conference on Information and knowledge management, ACM, 2009, pp. 375–384.

[32] C. Sauper, A. Haghighi, R. Barzilay, Content models with attitude, in: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, 1, Association for Computational Linguistics, 2011, pp. 350–358.

[33] S. Moghaddam, M. Ester, On the design of lda models for aspect-based opinion mining, in: Proceedings of the 21st ACM International Conference on Information and Knowledge Management, ACM, 2012, pp. 803–812.

[34] Z.C.A.M.B. Liu, Aspect extraction with automated prior knowledge learning, in: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, 2014, pp. 347–358.

[35] M. Shams, A. Baraani-Dastjerdi, Enriched lda (elda): combination of latent dirichlet allocation with word co-occurrence analysis for aspect extraction, Expert Syst. Appl. 80 (2017) 136–146.

[36] D.M. Blei, A.Y. Ng, M.I. Jordan, Latent dirichlet allocation, J.Mach. Learn.Res. 3 (Jan) (2003) 993–1022.

[37] A. Mukherjee, B. Liu, Aspect extraction through semi-supervised modeling, in: Meeting of the Association for Computational Linguistics: Long Papers, 2012, pp. 339–348.

[38] T. Wang, Y. Cai, H.F. Leung, R.Y.K. Lau, Q. Li, H. Min, Product aspect extraction supervised with online domain knowledge, Knowl. Based Syst. 71 (2014) 86–100.

[39] D. Chen, C. Manning, A fast and accurate dependency parser using neural networks, in: Conference on Empirical Methods in Natural Language Processing, 2014, pp. 740–750.

[40] T. Mikolov, K. Chen, G. Corrado, J. Dean, Efficient estimation of word representations in vector space, arXiv preprint arXiv:1301.3781(2013).

[41] K. Cho, B. van Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, Y. Bengio, Learning phrase representations using rnn encoder–decoder for statistical machine translation, in: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Association for Computational Linguistics, Doha, Qatar, 2014, pp. 1724–1734. URL http://www.aclweb.org/anthology/D14-1179.

[42] M. Pontiki, D. Galanis, H. Papageorgiou, I. Androutsopoulos, S. Manandhar, M. Al-Smadi, M. Al-Ayyoub, Y. Zhao, B. Qin, O.D. Clercq, Semeval-2016 task 5: aspect based sentiment analysis, in: International Workshop on Semantic Evaluation, 2016, pp. 19–30.