2014 IEEE International Conference on Data Mining Workshop

Joint Propagation and Refinement for Mining Opinion Words and Targets

Qiyun Zhao[1], Hao Wang[1,2], Pin Lv[1]
[1]Science and Technology on Integrated Information System Laboratory
[2]State Key Laboratory of Computer Science
Institute of Software, Chinese Academy of Sciences
Beijing, China
{qiyun12, wanghao, lvpin}@iscas.ac.cn

***Abstract*—This paper proposes a novel Joint Propagation and Refinement (JPR) method to extract opinion words and targets. We adopt a growing heuristic method to extract new opinion words and targets in two parallel processes: propagation and refinement. In the propagation process, we generate the candidate sets of opinion words and targets and construct Sentiment Graph Model (SGM) to evaluate the relations between opinion words and targets. We employ statistical word co-occurrence and dependency patterns to identify these relations. In addition, we discover new patterns by the newly extracted opinion words and targets, which can capture opinion relations more precisely in the case of informal texts. In the refinement process, we prune false results and update model iteratively. We employ Automatic Rule Refinement (ARR) to refine the rules of extraction, which means to refine the rule to extract false results. By using false results pruning and ARR process, we can efficiently alleviate the error propagation problem in traditional bootstrapping based methods. Experimental results on both English and Chinese datasets demonstrate the effectiveness of our method.**

Keywords—Opinion Mining; Sentiment analysis; Extraction; Bootstrapping; Refinement;

I. INTRODUCTION

Opinion mining, or sentiment analysis, has attracted a great deal of attention as its widespread application in public opinions detection, customer review summary and other systems which is required to extract people's opinions and sentiments. It not only assists buyers to make informed decisions, but also helps enterprises understand public opinions on their products or services. Extracting opinion words and opinion targets from opinion and review data is a key task in opinion mining. Opinion targets are aspects or features of objects which the opinions are expressed towards. Opinion words indicate the sentiment orientation, i.e. pos or neg.

In opinion words and targets extraction, identifying the relations between opinion words and targets plays an important role. Syntactic dependency structures are often used to understand grammatical modification relation between opinion words and their targets in [1][2]. Recent researches on opinion targets extraction have shown the effectiveness of syntactic patterns for opinion words and targets extraction in [3][4][5][6]. Similarly, in this paper, we utilize the dependency tree to discover the potential relations between opinion words and targets.

Existing approaches on opinion words and targets extraction have two types of framework: one is pipeline framework; the other is propagation (or bootstrapping-based) framework. In the pipeline framework, candidates of opinion expressions and opinion targets are generated first, and then they filter false results with refinement methods in [7][8][9]. In the refinement process, they took rule-based or machine learning approaches to identify potential relations between opinions and targets. The main challenge is the effectiveness of the refinement methods, because it decides the extraction result. In addition to the pipeline framework, researchers try to identify opinion words and targets iteratively in the propagation framework in [3][5][10]. The extraction result extends with heuristic rules in the iterative propagation process, but it could be affected by the error propagation.

Based on previous researchers, we point out some major challenges in the opinion words and targets extraction:

- **False opinion targets pruning.** Error propagation problem increases the probability to extract false results
- **Long-tail opinion targets discovery.** Pre-defined syntactic rules are difficult to cover all real-world cases because most of the reviews are informal and they contain a lot of grammatically incorrect sentences.
- **Domain adaptation problem.** Vocabulary frequency changes from one domain to another. Instance adaption models the changes of instance probability [21].

In this paper, we propose a Joint Propagation and Refinement (JPR) method to extract opinion words and targets. The basic idea of our method is to adopt refinement methods jointed with a propagation framework. Our contributions in this paper are summarized as follows:

- We propose a novel JPR method that combines the refinement process based on bootstrapping in a jointed framework. By using this method we can alleviate the problems of error propagation and long-tail results discovery in previous propagation or pipeline methods.

978-1-4799-4274-9/14 $31.00 © 2014 IEEE
DOI 10.1109/ICDMW.2014.66

- We identify potential opinion relations to extract more latent opinion words and targets in the case of informal texts and error parsing in real world. Meanwhile, we employ Automatic Rule Refinement (ARR) to pruning false results and update rules of extraction iteratively to improve the extraction performance.
- We evaluate our method using real-world datasets both in English and Chinese, and experimental results show the effectiveness of our approach compared with the statof-art methods.

II. RELATED WORK

In earlier research, researchers usually extracted high-frequency noun phrases as opinion targets in [7]. They held an intuitive idea that users may mention the targets many times in the review text. However, not all frequent nouns are opinion targets. That is, some opinion targets are low frequent.

Then some researchers utilized the relations between opinion words and targets to extract targets in [11][12]. Since the opinion words are easier to identify than targets with the help of opinion lexicon, the relations between opinion words and targets can be used to extract new targets in [3]. These opinion relations were usually shaped by the position span, POS tags and syntactic dependency structures. Due to fewer false opinion targets are extracted, relation-based methods gain higher accuracy than frequency-based methods.

Recently, some hybrid methods used opinion relations to extract opinion targets while filtering frequent noun phrases to increase accuracy in [13][14].

In addition, some researchers adopted model-based methods to handle the opinion targets extraction. They treated the extraction task as a labeling problem and employed supervised learning techniques [15][16][17][18]. However, it was difficult to obtain the annotated training data, and the trained models had limited uses in certain domains.

A typical bootstrapping framework is Double Propagation (DP) in [3]. DP is a bootstrapping based method, which propagates information between opinion words and targets. DP expands new opinion words and targets iteratively with predefined propagation rules based on syntactic patterns. The advantages of this method are straightforward, scalable and domain-independent. However, error propagation increases the probability to extract false results. In order to gain more true targets, we should apply effective pruning methods to refine the results.

Though DP has good domain-independency, it cannot be effectively applied to online reviews in real world. The main reason is that most of the reviews are informal and consist of a lot of grammatically incorrect sentences. As matter of fact, it is difficult to construct a comprehensive set of dependency relations between features and opinions to cover all cases. Some further work has been taken to improve the performance of the DP method, such as DPHITS with hyperlink-induced topic search algorithm in [5], and LSTBOOT using likelihood ratio tests for bootstrapping in [20].

There are also many refinement methods introduced in pipeline methods. A typical pipeline framework is Two-Sage Framework (TSF) in [6]. TSF first generated candidates of opinion words and targets, and then used well-designed models to refine the result. However, only adjectives were labeled as opinion candidates, so that opinion words such as verbs or nouns couldn't be identified. Furthermore, labeling all nouns as target candidates could induce much noise information, resulting in increasing the difficulty of refinement.

A recent knowledge-based approach is sentic computing [22][23], which relies on the ensemble application of common-sense computing and the psychology of emotions to infer the conceptual and affective information associated with natural language. In [24], they introduced a novel paradigm to concept-level sentiment analysis that merges linguistics, common-sense computing, and machine learning for improving the accuracy of this task. This approach achieved a better understanding of the contextual role of each concept within the sentence by allowing sentiments to flow from concept to concept based on the dependency relation of the input sentence. More and more opinion mining systems need broader and deeper common and commonsense knowledge bases [25]. Sentiment analysis research is moving toward content, concept, and context based analysis of natural language text, supported by time-efficient parsing techniques suitable for big social data analysis [26].

III. JOINT PROPAGATION AND REFINEMENT

Joint Propagation and Refinement (JPR) method is a bootstrapping based method. Figure 1 introduces the detailed process of JPR method, where we take opinion seed words set $\{O_{seed}\}$, dependency patterns $\{P\}$ and review data $\{D\}$ as the input. We scan all the sentences in the dataset, and we adopt a syntactic parsing method to capture the dependency structure on each sentence. Except the initial choice of opinion lexicon and extraction rules, JPR is carried out without any manual intervention.

At the beginning, we generate two candidate sets of opinion words and targets by employing the pre-defined rules. Then we iteratively extract opinion words and targets using predefined extraction rules and existing result set. There is a rule set containing several rules to identify the conditions for extraction. Most of the rules describe the latent relations between opinion words and targets, i.e. word co-occurrence or dependency patterns. The details of the rules will be described later.

In the propagation process, existing opinion words are used to find new opinion targets, which satisfy the rules of extraction. At the same time, the relations between opinion words and targets are identified during the extraction. We apply the structure of Sentiment Graph Model (SGM) to measure the relations between opinion words and targets, and quantize the relations by computing the weight on each edge on the graph. After the extraction, we employ several refinement methods to prune false results.

In the refinement process, we check the conditions of the rules to prune false opinion words and targets in the candidate sets $\{OC\}$ and $\{TC\}$. After pruning and refining, the remained extracted opinion words and targets are added to the refined result set $\{O\}$ and $\{T\}$, and pruned words generate a false result set $\{O_{false}\}$ and $\{T_{false}\}$. Then we apply these refined result set $\{O\}$ and $\{T\}$ to update SGM by adjusting the model parameters. We also take the rule refinement by the false result set $\{O_{false}\}$ and $\{T_{false}\}$ to update or remove rules of extraction. The refined opinion words and targets $\{O\}$ and $\{T\}$, updated SGM, as well as the refined rules of extraction are all applied for further opinion words and targets extraction. Repeat the propagation of extraction until no new opinion words and targets are identified.

Joint Propagation and Refinement

```
Input:    Review Data {D},
          Opinion seed words {O_seed},
          Rules of extraction {R},
Output:   Opinion targets set {T},
          Opinion words set {O},
Utility:  Dependency patterns {P},
          Sentiment Graph Model: G,
Algorithm:
Add {O_seed} to {O};
Extract a candidate opinion set {CO} by {R};
Extract a candidate target set {CT} by {R};
Build G and compute parameters;
repeat
    for each opinion words o in {O} do
        for each candidate target t in {CT} do
            apply {R_O-T} to identify t as a target,
            add it to {T}, remove it from {CT};
        end for
    end for
    for each opinion target t in {T} do
        for each candidate opinion o in {CO} do
            apply {R_O-T} to identify t as a target,
            add it to {O}, remove it from {CO};
        end for
    end for
    for each opinion target t in {T} do
        for each candidate target t in {CT} do
            apply {R_T-T} to identify t as a target,
            add it to {T}, remove it from {CT};
        end for
    end for
    for each opinion word o in {T} do
        for each opinion target t linked to o do
            If dependency path p between o and t
            is not in {P}, add p to {P};
        end for
    end for
    Pruning false results {O_false}, {T_false};
    for each opinion word o in { O_false } do
        for each opinion target t linked to o in {T_false} do
            Refine the r in {R} which is used to extract o, t;
            Update Conf(p) which is used to extract o and t;
        end for
    end for
    Remove {O_false}{T_false} from {O}{T}
    Update parameters on G;
until No new opinion words and targets are extracted;
return {O}{T}
```

Fig. 1. Joint Propagation and Refinement method

IV. Sentiment Graph Model

In this section, we will introduce some concepts in JPR method. To model the relations between opinion words and targets, we define the construction of Sentiment Graph Model.

A. *Sentiment Graph with Opinion Words and Targets*

Sentiment Graph Model: is a weighted, directed graph. Opinion words, targets are represented as vertices in the graph model.

First, we need to generate two candidate sets of opinion words and targets. Then we connect pairs of co-occurrence candidates in these sets. Figure 2 shows the structure of the sentiment graph of opinion words and targets.

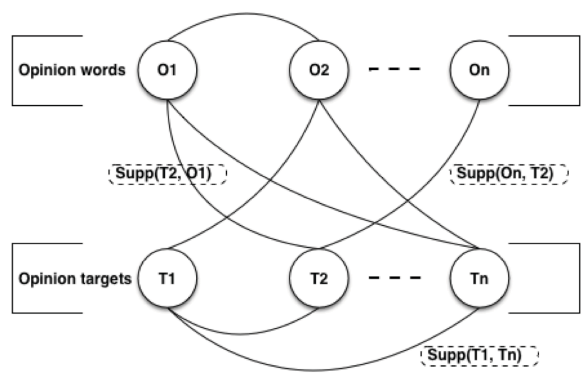

Fig. 2. Sentiment Graph of opinion words and targets

Now we introduce the estimation of the weight on the Sentiment Graph Model. First, we calculate a frequency table of two terms i and j, which represent opinion word or target candidates. As shown in Table 1, $C_{i,j}$ is the number of reviews containing term i and j; $P_{i,\bar{j}}$ is the number of reviews containing term i but not j; $Q_{\bar{i},j}$ is the number of reviews containing term j but not i; $R_{\bar{i},\bar{j}}$ is the number of reviews containing neither i nor j.

TABLE I. Table Type Styles

Frequency	j	$\bar{j}$
i	$C_{i,j}$	$P_{i,\bar{j}}$
$\bar{i}$	$Q_{\bar{i},j}$	$R_{\bar{i},\bar{j}}$

Then we measure the association of pair-wise terms as the support level for each pair of candidates. Here is the function to calculate the support level:

$$Supp(i,j) = 2[log\, L(P(i|j), C_{i,j}, N_j) + log\, L(P(i|\bar{j}), P_{i,\bar{j}}, N_{\bar{j}}) - log\, L(P(i), N_i, N] \quad (1)$$

where,

$$P(i|j) = \frac{C_{i,j}}{N_j} \quad (2)$$

$$L(p,k,n) = p^k(1-p)^{n-k} \quad (3)$$

N is the number of reviews. N_i is the frequency of term i in the corpus. $P(i|j)$ is the conditional probability of term i when term j occurs. We define $Supp(i, j)$ as the support level of term i to j. Then we use $Supp(o, t)$ as the weight of the edge $(e: v_o \rightarrow v_t)$ on the Sentiment Graph. Similarly, we use $Supp(t, o)$ and $Supp(t, t)$ as the weight of edges $(e: v_t \rightarrow v_o)$ and $(e: v_t \rightarrow v_t)$ representatively.

For example, the term "clear" appears 374 times in the corpus, as well as "voice" and "screen" appear 165 and 262 times respectively. The number of reviews (N) is 1000. There are 85 reviews contains both "voice" and "clear", such as in the sentence "The voice of my headphone is not clear". There are 135 reviews contains both "screen" and "clear", such as in the sentence "This cellphone got a clear and bright screen". In this case, we get the frequency of: N(clear)=374, N(voice)=165, N(screen)=262, C(clear, voice)=165, C(clear, screen)=262. So we can apply these parameters to compute P(clear | voice) = C(clear, voice) / N(clear) = 165 / 374 = 0.441, P(clear | screen) = C(clear, screen) / N(clear) = 262 / 374 = 0.700. And then, we are able to compute the likelihood ratio and Supp(clear, voice) and Supp(clear, screen).

We take the weight of each edge on the Sentiment Graph to describe the associated relations between different opinion words and targets.

B. Dependency Patterns

We employ the dependency parsing on sentences to identify the potential relations between opinion words and targets. The dependency structure has been turned out to be very useful in extracting opinion words and targets in [19]. Dependency grammar describes the dependency relations between words in a sentence. In the dependency structure, words are linked to each other by dependency relation. In [3], it describes both direct relations and indirect relation of dependency between two words. In [24][28], they employ a concept-based semantic parser for deconstructing text into natural language concepts based on the dependency relation.

We take the shortest path between opinion word and target on the dependency structure in a sentence. Such dependency path is used as the dependency patterns to measure the relation between opinion words and targets. Opinion word and target are replaced by wildcards "<O>" and "<T>" on the path, while other words are replaced by their POS tags. For example, in the sentence "The hotel has a clean room", "clean" is the opinion word and "room" is the opinion target, and "room" is also a feature of "hotel". In dependency structure, "clean" depends on "room" with relation "{mod}". So the dependency pattern is "<O>{mod}<T>". For another example of indirect relation, "The bed is comfortable", opinion target "bed" depends on the verb "is" with the relation of "{s}", which means that "bed" is the surface subject of "is", and opinion word "comfortable" depends on the verb "is" with the relation of "{pred}", which means that "comfortable" is the predicate of the "is" clause. So the dependency pattern between opinion word and target in this sentence is "<O>{pred}<VBE>{s}<T>".

There are three kinds of dependency patterns: "O-T", "O-O" and "T-T". "O-T" represents the pattern between opinion words and targets. Similarly, "O-O" and "T-T" represent the pattern within opinion words or targets representatively.

In this paper, we utilize the Stanford Parser[1] to gain dependency tree of sentences and use the Stanford Tagger[2] for POS tagging.

C. Sentiment Graph with Dependency Patterns

As dependency patterns are useful to identify relations between opinions and targets, we add them as vertices to the Sentiment Graph. Each dependency represents a syntactic relation between opinion words and targets. New edges that connect the patterns and opinion words or targets would be also added to the graph. The construction of the graph is shown in Figure 3.

In JPR method, we keep a dependency pattern set to record patterns that contain opinion words and targets. Though it is difficult to construct a comprehensive set of dependency relations between targets and opinions to cover all real-world cases, we discover potential dependency patterns and measure its confidence to discover new opinion words and targets.

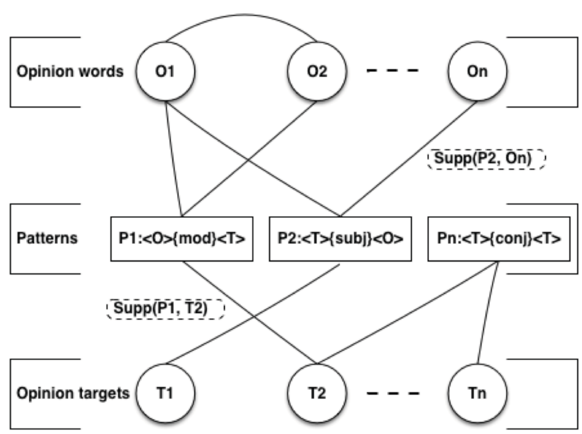

Fig. 3. Sentiment Graph with dependency patterns

We compute the support level for each associated pattern and opinion word, or target, which are used to evaluate the confidence of the dependency pattern used to extract new opinion words and targets. We take the support level as the weights on the Sentiment Graph Model. We evaluate the confidence of each dependency pattern by employing the following function:

$$Conf(p) = \sum_{i=1}^{N_o} Supp(p, o_i) + \sum_{j=1}^{N_t} Supp(p, t_j) \quad (4)$$

$Conf(p)$ is the sum of weights on all associated opinion words and targets with the pattern. Only with high confidence can the patterns be used to extract new opinion words and targets. The confidence score is used in rules of extraction. After refinement process, the confidence score need to be updated as some false results have been pruned.

---

[1] http://nlp.stanford.edu/software/lex-parser.shtml
[2] http://nlp.stanford.edu/software/tagger.shtml

V. AUTOMATIC RULE REFINEMENT

Rule-based methods are often used in opinion extraction tasks. In [3] they define rules for targets and opinion words extraction. For example, the rule of "O->O-Dep->T, O∈{O}, O-Dep∈{MR}, POS(T)∈{NN}" is used to extract new target. In the sentence of "The phone has a good screen", we find the dependency pattern of (good->mod->screen). Then we are able to extract screen as a new target, as "good" is known as opinion words. In [27] they employ bag-of-words and rule-based algorithm to process the special characteristics of social media texts, such as emoticons. In [29] they propose a novel rule-based approach that exploits common-sense knowledge and dependency trees to detect both explicit and implicit aspects.

In this work, we also define several rules of extraction to discover new opinion words and targets. The initial rules are shown in the Table 2 below:

TABLE II. RULES OF EXTRACTION

No.	Rules of Extraction
R1	Extract *adj.* as opinion word.
R2	Extract word in $\{O_{seed}\}$ as opinion word.
R3	If o is in $\{O\}$ and $Supp(o, t) > threshold$, extract t as target.
R4	If t is in $\{T\}$ and $Supp(t, o) > threshold$, extract o as target.
R5	If t is in $\{T\}$ and $Supp(t, t) > threshold$, extract t as target.
R6	If o is in $\{O\}$ and pattern "O-T" is in $\{P\}$ in dependency structure, extract t as target.
R7	If t is in $\{T\}$ and pattern "O-T" is in $\{P\}$ in dependency structure, extract o as target.
R8	If t is in $\{T\}$ and pattern "T-T" is in $\{P\}$ in dependency structure, extract t as target.
R9	If o is in $\{O\}$ and t is in $\{T\}$ in dependency structure, add p to $\{P\}$.

The first two rules are used to generate the initial candidate sets of opinion words and targets. It is intuitive that an adjective word is likely to present sentiment or opinion. However, many opinion words can be nouns or verbs, such as "atrocity" and "enjoy". So we take opinion seed words as input to extract different kind of opinion words. We label adjective and words in opinion seed words set as opinion candidates. Then our approch will expand the extraction in propagation process to dicover new opinion words of not only adjectives but also nouns and verbs. The rest of the rules contain different restriction to specify the conditions for extraction. In the propagation process, these rules are applied to filter the candidate set and identify opinion words and targets that satisfy the words co-occurrence threshold or match the dependency patterns with high confidence score.

We construct the Sentiment Graph Model on the candidate result set before propagation. Then we apply the rules of extraction to filter the candidate set. Taking R3 as example to demonstrate the process of extraction, we find an opinion word o existing in the result set in a sentence. If there is a target candidate t in the same sentence, we check the support level of o and t on the Sentiment Graph Model. Then we identify t as a target if the support level is higher then the threshold. After dependency parsing on one sentence, we check the rules with dependency patterns. If one pattern matches the dependency path on the sentence, we apply the corresponding rules to extract new opinion words or targets. We also discover potential dependency patterns with the help of extracted opinion words and targets.

In the Sentiment Graph Model, the similarity between different opinion words or opinion targets can be describe by the associated patterns and edges. If opinion words are linked to many coincident patterns, we take these similar opinion words to identify new targets. In this way we discover new rules of extraction during the propagation process besides the intuitive ones. For example, we find "powerful" and "durable" are similar in the Sentiment Graph Model, and we identify "battery" as an opinion targets in the sentence of "The battery of this mobile phone is powerful". In another sentence, "This cellphone has a durable and long battery life", we can apply this similarity between "powerful" and "durable" to extract "battery" as opinion target in this case.

In addition, we also update our rules of extraction by considering the pruned false results. This process is called automatic rule refinement. We collect the pruned candidates to a false result set. The false results are examples of incorrect extraction. Then we find out the rules used to extract these results, and adjust parameters in these rules, such as threshold, support level of opinion relation and confidence score of patterns. We expect to learn and update the rules with the help of the extraction results automatically, which is self-adaption in different domain and application. Modifying a rule to remove false result can simultaneously remove other false results. However, this action may also remove some correct results. Only when the false results are extracted by specific rule frequently, can the rule be refined seriously in order to reduce the side effect. By discovering the similarity between opinion words and targets and updating the threshold of false result pruning, we automatically refine our rules of extraction.

VI. DISCUSSION

In practice, JPR is very flexible and easy to optimize in diverse kinds of application. As the initial opinion lexicon and Sentiment Patterns are domain-independent, JPR is available for cross-domain datasets. In the refinement process, newly extracted opinion words and targets are added into the refined sets after pruning false results. The refinement classifiers are re-trained on the refined sets instead of the whole result sets. In this way, we reduce the workload of training. Furthermore, the classifiers and model are able to be applied on the newly added data without scanning the whole dataset. Stochastic gradient descent (SDG) method is also used to reduce the training workload by retraining the refinement classifiers on newly added data only. The advantages of JPR are listed below:

- Domain-independence and scalability. Our method employ heuristic extraction rules and self-adapted learning strategy.
- Automatic rule refinement method. We update the refinement classifiers after extraction in iteration. The refinement method is improved iteratively to reduce error propagation.
- Potential dependency patterns discovery. Different types of opinion words and long-tail targets are identified by discovering potential patterns.

TABLE III. RESULTS FOR OPINION TARGETS EXTRACTION

Methods	D1			D2			D3			D4			Avg.
	P	R	F	P	R	F	P	R	F	P	R	F	F
DP	0.54	0.50	0.52	0.57	0.52	0.54	0.60	0.56	0.58	0.58	0.53	0.55	0.55
DPHITS	0.65	0.59	0.62	0.66	0.62	0.64	0.68	0.64	0.66	0.69	0.66	0.67	0.65
LRTBOOT	0.65	0.70	0.67	0.68	0.71	0.69	0.68	0.72	0.70	0.70	0.73	0.71	0.70
TSF	0.69	0.71	0.70	0.72	**0.73**	0.72	0.74	**0.76**	0.75	0.76	0.73	0.74	0.73
JPR	**0.75**	**0.72**	**0.73**	**0.78**	0.72	**0.75**	**0.80**	**0.76**	**0.80**	**0.80**	**0.75**	**0.77**	**0.76**

TABLE IV. RESULTS FOR OPINION WORDS EXTRACTION

Methods	D1			D2			D3			D4			Avg.
	P	R	F	P	R	F	P	R	F	P	R	F	F
DP	0.58	0.60	0.59	0.60	0.64	0.62	0.66	0.65	0.65	0.67	0.69	0.68	0.64
TSF	0.64	0.69	0.66	0.68	**0.72**	0.70	0.78	0.76	0.77	0.77	0.73	0.75	0.72
JPR	**0.72**	**0.73**	**0.72**	**0.70**	**0.72**	**0.71**	**0.80**	**0.78**	**0.79**	**0.82**	**0.78**	**0.80**	**0.75**

VII. EXPERIMENTS

A. Datasets and Evaluation Metrics

We select two real world datasets in English and Chinese to evaluate our method. The datasets contain reviews on different products, so we test the performance of our method on cross-domain data. The first dataset in (Wang et al., 2011) has about 20000 sentences in English, which contains reviews on hotel (D1) and MP3 (D2).

The second dataset published in COAE2008[3] has about 5000 sentences, which contains Chinese review data on camera (D3) and car (D4). Besides some pre-annotated opinion targets in the datasets, we manually annotate opinion words and targets in sentences.

Evaluation Metrics: We evaluate our method by precision (P), recall (R) and F-measure (F).

B. Our Method v.s. State-of-the-art

We evaluate our opinion words and targets extraction performance of the proposed JPR against four state-of-art competitors listed below:

- **DP**: Double Propagation in [3].
- **DPHITS**: DP with hyperlink-induced topic search algorithm in [5].
- **LSTBOOT**: likelihood ratio tests for bootstrapping in [20].
- **TSF**: Two-Stage Framework in [6].

All of the above competitors are unsupervised methods. The first three methods are based on bootstrapping framework. DP extracts opinion words and targets using syntactic dependency relations. Some syntactic rules are manually defined for extraction. DPHITS uses hyperlink-induced topic search algorithm (HITS) to validate potential targets recognized by DP plus two additional syntactic patterns of “part-whole” and “no”. The last competitor TSF is a typical pipeline framework. TSF first generates candidates of opinion words and targets, then uses well-designed models to refine the result.

All of the above approaches use same five common opinion word seeds as {good, bad, nice, poor, perfect}. The choice of opinion seeds seems reasonable, as most people can easily use this opinion words to express their basic opinion and sentiment orientation.

We take the task of the opinion target extraction on these five methods. As DPHITS and LRTBOOT don’t extract opinion words, we only show the performance of opinion words extraction on DP, TSF and our JPR method.

Table 3 shows the experimental results for opinion targets extraction and Table 4 shows the results for opinion words extraction. We have the following analysis from the results table:

- JPR outperforms the four competitors in terms of F-measure on all domains with the highest average improvement of 0.21 in opinion words extraction and 0.09 in opinion targets extraction, and it outperforms the method ranked only second to our method at 0.03 in these two tasks. It indicates the effectiveness and domain-independence of our method.
- JPR achieves about 0.21 improvement in F-measure compared with DP. JPR also outperforms the other two bootstrapping methods at 0.05-0.10 in F-measure. We believe the improvement on recall score benefits from our pattern discovery, as new patterns can identify more opinion words and targets. The improvement in precision indicates the effectiveness of our iterative refinement process, which reduces the error propagation. In addition, the automatic rule refinement also makes contributions to extract more correct opinion words and targets compared with static and rules chosen manually.
- JPR outperforms pipeline methods in terms of precision at 0.03-0.06 and has comparable recall. It indicates that the process of updating refinement method during the iterations plays an important role to reduce false extraction. The reason for the close score of recall is the false drop in the refinement process. The false drop

[3] http://ir-china.org.cn/coae2008.html

error is also reinforced this in the propagation process. The scale of test datasets and the limited types of annotated data also have some negative impacts to the result.

C. *The Effect of Refinemnet*

In this section, we test two variants of our method to analyze the effect of refinement.

We adjust our method by removing some steps in the whole process to generate the variants methods. The description of these methods is listed below:

- **Ours-Full**: is the full implementation of our JPR method.
- **Ours-No-Auto**: is the implementation of our method with no automatic rule refinement process, which means that it doesn't update rule of extraction in propagation.
- **Ours-No-RF**: is the implementation of our method without any refinement process, such as pruning false result and automatic rule refinement.

In order to show the performance more clearly, we also add two competitors methods introduced before (LRTBOOT and TSF).

Then we employ these methods on the same datasets and check the performance. Results for opinion words targets extraction are shown in Figure 4 and Figure 5.

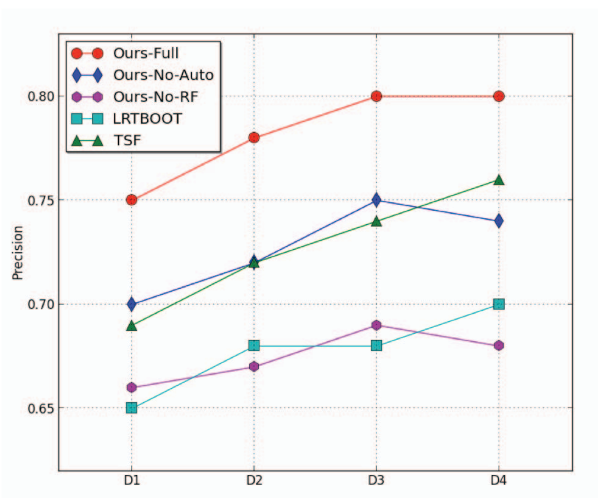

Fig. 4. Precision of opinion targets extraction

We see that Ours-Full outperforms Ours-No-Auto and Ours-No-RF at 0.05 and 0.10 in precision representatively. With the refinement on the rules of extraction automatically, the rules used for extraction have higher confidence. The gap between Ours-No-Auto and Ours-No-RF indicates that to pruning false results has significant effect. We believe it is because our method estimated confidence of patterns so false opinion relations are reduced. Therefore, the consideration of pattern confidence is beneficial as expected, which alleviates the false results extraction problem. Another interesting phenomenon is that Ours-No-Auto has comparable result with TSF method, which indicates the efficiency of our method.

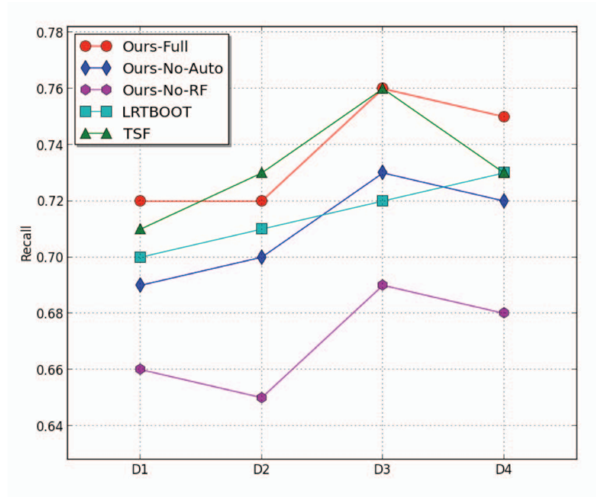

Fig. 5. Recall of opinion targets extraction

As showed in Figure 5, we find the improvement on recall is not significant as on precision.

Ours-Full outperforms Ours-No-RF about 0.06 in recall. It proves that the refinement process is helpful to discover more correct results. And we find that Ours-No-Auto method has close recall score compared with other competitor methods.

VIII. Conclusions and Future Work

This paper proposes a novel Joint Propagation and Refinement method for opinion words and targets extraction. Unlike the existing propagation framework or pipeline framework, JPR method combines the refinement process based on bootstrapping. We employ a Sentiment Graph Model containing dependency patterns to evaluate the relations between opinion words and targets. We also adopt an automatic rule refinement to pruning the false results and update the rules for extraction to improve performance. The experimental results show that JPR achieves higher performance over current state of the art unsupervised methods.

In the future, we plan to focus on improving the precision of opinion words and targets extraction by working on the refinement methods. We also plan to reduce the false drop to increase recall in the process of refinement, which is a side effect of rule refinement. Then we will also try to design new models to improve the Sentiment Graph Model to discover the potential relations between opinion words and targets.

Acknowledgment

This work is supported by National Natural Science Foundation of China (61303164), Beijing Natural Science Foundation (9144037), National High Technology Research and Development Program of China (863 Program) (2012AA011206) and National Basic Research Program of China (973 Program)(2013CB329305).

References

[1] A.M. Popescu, and O. Etzioni, "Extracting product features and opinions from reviews," In *Natural language processing and text mining*, Springer London, 2007, pp. 9-28.

[2] N. Kobayashi, K. Inui, and Y. Matsumoto, "Opinion mining from web documents: Extraction and structurization," *Information and Media Technologies 2*, no. 1, 2007, pp. 326-337.

[3] G. Qiu, B. Liu, J. Bu, and C. Chen, "Expanding domain sentiment lexicon through double propagation," In *IJCAI'09*, vol.9, 2009, pp. 1199–1204.

[4] G. Qiu, B. Liu, J. Bu, and C. Chen, "Opinion word expansion and target extraction through double propagation," *Computational Linguistics 37*, no.1, 2011, pp. 9–27.

[5] L. Zhang, B. Liu, S. H. Lim, and E. O'Brien-Strain, "Extracting and ranking product features in opinion documents," In *Proc. of the 23rd Int. Conf. on Computational Linguistics: Posters*, pp. 1462-1470. Association for Computational Linguistics, 2010.

[6] L. Xu, K. Liu, S. Lai, Y. Chen and J. Zhao, "Mining Opinion Words and Opinion Targets in a Two-Stage Framework," In *ACL'13*, 2013, pp. 1764–1773.

[7] M. Hu and B. Liu, "Mining opinion features in customer reviews," In *AAAI*, vol.4, no.4, 2004, pp. 755–760.

[8] Y. Wu, Q. Zhang, X. Huang, and L. Wu, "Phrase dependency parsing for opinion mining,". In *EMNLP'09*, Volume 3-Volume 3, pp. 1533-1541. Association for Computational Linguistics, 2009.

[9] B. Liu, "Sentiment analysis and opinion mining," Synthesis Lectures on Human Language Technologies 5, no.1, pp. 1–167.

[10] E. Riloff, and J. Wiebe, "Learning extraction patterns for subjective expressions," In *EMNLP'03*, pp. 105-112. Association for Computational Linguistics, 2003.

[11] S Baccianella, A Esuli, and F. Sebastiani, "Multi-facet rating of product reviews," In *Advances in Information Retrieval*. Springer Berlin Heidelberg, 2009, pp. 461-472.

[12] L. Jiang, M. Yu, M. Zhou, X. Liu, and T. Zhao, "Target-dependent twitter sentiment classification," In *Proc. of the 49th Ann. Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pp. 151-160. Association for Computational Linguistics, 2011.

[13] S. Blair-Goldensohn, K. Hannan, R. McDonald, T. Neylon, G. A. Reis, and J. Reynar, "Building a sentiment summarizer for local service reviews," In *Proc. of WWW workshop on NLP in the Information Explosion Era*, 2008, p. 14.

[14] S. Moghaddam and M. Ester, "Opinion digger: an unsupervised opinion miner from unstructured product reviews," In *CIKM'10*, ACM, 2010, pp. 1825-1828.

[15] W. Jin, H. H. Ho, and R. K. Srihari, "OpinionMiner: a novel machine learning system for web opinion mining and extraction," In *SIGKDD'09*, ACM, 2009, pp. 1195-1204.

[16] F. Li, C. Han, M. Huang, X. Zhu, Y. Xia, S. Zhang, and H. Yu, "Structure-aware review mining and summarization," In *COLING '10*, pp. 653–661. Association for Computational Linguistics, 2010.

[17] H. Lakkaraju, C. Bhattacharyya, I. Bhattacharya, and S. Merugu, "Exploiting Coherence for the Simultaneous Discovery of Latent Facets and associated Sentiments", In *SDM* , 2011, pp. 498-509.

[18] B. Yang, and C. Cardie, "Joint inference for fine-grained opinion extraction," In *ACL'13*, 2013, pp. 1640-1649.

[19] R. Johansson and A. Moschitti, "Syntactic and semantic structure for opinion expression detection," In *Proc. of the 14th Conf. on Computational Natural Language Learning*, pp. 67–76. Association for Computational Linguistics, 2010.

[20] Z. Hai, K. Chang, and G. Cong, "One seed to find them all: mining opinion features via association," In *CIKM'12*, 2012, pp. 255-264.

[21] R. Xia, C. Zong, X. Hu, E. Cambria, "Feature Ensemble Plus Sample Selection: Domain Adaptation for Sentiment Classification," *Intelligent Systems*, IEEE, vol. 28, no. 3, pp. 10-18, 2013.

[22] E. Cambria, A. Hussain. *Sentic Computing: Techniques, Tools, and Applications*. Dordrecht, Netherlands: Springer, 2012.

[23] E. Cambria, B. Schuller, B. Liu, H. Wang, C. Havasi, "Knowledge-Based Approaches to Concept-Level Sentiment Analysis," *Intelligent Systems*, IEEE, vol. 28, no. 2, pp. 12-14, 2013.

[24] S. Poria, E. Cambria, G. Winterstein, G. B. Huang, "Sentic patterns: Dependency-based rules for concept-level sentiment analysis,". *Knowledge-Based Systems*, vol.69, pp. 45-63, 2014.

[25] E. Cambria, B. Schuller, Y. Xia, C. Havasi, "New avenues in opinion mining and sentiment analysis," *Intelligent Systems*, IEEE, vol.28, no. 2, pp. 15-21, 2013.

[26] E Cambria, H Wang, B White, "Guest Editorial Introduction: Big Social Data Analysis", *Knowledge-Based Systems*, vol.69, pp. 1-2, 2014.

[27] Y. Xie, Z. Chen, K. Zhang, Y, Cheng, D. K. Honbo, A. Agrawal, A. N. Choudhary, "MuSES: Multilingual Sentiment Elicitation System for Social Media Data," *Intelligent Systems*, IEEE , vol.29, no.4, pp. 34-42, 2014.

[28] S. Poria, B. Agarwal, Gelbukh, A. Hussain, N. Howard, "Dependency-based semantic parsing for concept-level text analysis"," In Gelbukh, A. (ed.) *Computational Linguistics and Intelligent Text Proceccing*, Part I. LNCS, vol. 8403, pp. 113–127. Springer, Heidelberg, 2014.

[29] S. Poria, E. Cambria, L.W. Ku, C. Gui, A. Gelbukh, "A rule-based approach to aspect extraction from product reviews," In *Workshop Proc. of the 25th International Conference on Computational Linguistics, COLING'14*, pp. 28-37, 2014.