Large-scale Opinion Relation Extraction with Distantly Supervised Neural Network

Changzhi Sun[2], **Yuanbin Wu**[1, 2], **Man Lan**[1, 2],
Shiliang Sun[1, 2], **and Qi Zhang**[3]

[1]Shanghai Key Laboratory of Multidimensional Information Processing
[2]Department of Computer Science and Technology, East China Normal University
[3]School of Computer Science, Fudan University
{changzhisun}@stu.ecnu.edu.cn
{ybwu,mlan,slsun}@cs.ecnu.edu.cn
{qz}@fudan.edu.cn

Abstract

We investigate the task of open domain opinion relation extraction. Given a large number of unlabelled texts, we propose an efficient distantly supervised framework based on pattern matching and neural network classifiers. The patterns are designed to automatically generate training data, and the deep learning model is designed to capture various lexical and syntactic features. The result algorithm is fast and scalable on large-scale corpus. We test the system on the Amazon online review dataset, and show that the proposed model is able to achieve promising performances without any human annotations.

1 Introduction

Opinion mining systems aim to detect and extract opinion-related information from texts. With the help of natural language processing algorithms and large-scale user generated contents, researchers could take a closer look at how people express their opinions on various objects and topics. Such observations are important both for applications (e.g., recommendation and retrieval) and linguistic studies.

In this paper, we address the task of opinion relation extraction. The task tries to identify opinion expressions (words indicating sentiments, emotions and comments), opinion targets (objects of opinions) and their relations (what opinion on which target). The following are two examples.

> 1. The unit is [well designed] and [perfect reception].
> 2. The Passion of The Christ will [touch your heart].

Opinion relations in the two sentences are ("well designed", "unit"), ("perfect reception", "unit"), and ("touch your heart", "The Passion of The Christ"). Extracting opinion bearing relations is usually the first step towards fine-gained analysis of opinion in texts, and plays an important role in other sentiment related applications (e.g., sentiment summarization). The goal of this paper is to extract opinion relations from open domain large-scale opinion bearing texts.

Previous works on fine-grained opinion information extraction have achieved notable success on many aspects: various domains were examined (Pontiki et al., 2015), different types of relations were studied (Ganapathibhotla and Liu, 2008; Narayanan et al., 2009; Wu et al., 2011), and both supervised and unsupervised (pattern-based) algorithms were applied. But we have observed some difficulties when trying to use existing methods. The pattern based methods (both lexical patterns and syntactic patterns) are simple, fast, and scalable on large-scale datasets. However, the robustness of patterns is usually questionable in practice. For example, syntactic patterns are sensitive to errors in parse trees, which are common in user generated contents. Lexical patterns either have limited coverage (e.g., fixed set of patterns (Riloff and Wiebe, 2003)), or hard-to-control noise (e.g., bootstrapping approaches (Qiu et al., 2011)). On the other hand, supervised models can achieve better performances than patterns on manually labeled datasets, but it is often difficult to obtain large number of annotations for the relation extraction task, and the trained models are

Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers, pages 1033–1043,
Valencia, Spain, April 3-7, 2017.

also limited to specified domains. Thus, we still need an algorithm to better combine the power of syntactic and lexical patterns, while reduce manual annotations.

Another problem is that many existing systems rely on general purpose opinion lexicons to select candidate relations. If an opinion expression is not recognized by the lexicon, the systems are unable to extract the related relations. As an example, one weakness of many existing lexicons is the lack of support on multiword expressions (e.g., "more than what I expected", "honest to the book" and "adrenaline pumping"), which are common in opinion bearing texts. Another example is that some true expressions could be ignored either due to errors from POS taggers and syntactic parsers. Thus, to enlarge the coverage of opinion relation extraction, we need some method to better detect opinion expressions.

Regarding above problems, we make efforts to contribute to following aspects.

First, we propose a distantly supervised algorithm for open domain opinion relation extraction. The algorithm first applies domain independent patterns to get a set of opinion relations, then trains a classifier on them. We show that, although the relations from pattern matching are not as accurate as gold standard annotations, the distantly supervised classifier still improves performances. Our algorithm significantly outperforms the double propagation algorithm (Qiu et al., 2011), which is the state-of-the-art unsupervised opinion relation extraction system.

Second, we develop a neural network model to learn representations for lexical and syntactic contexts. The model uses bidirectional LSTMs to capture global information and convolutional neural networks to get local low dimensional feature embeddings. Comparing with other neural network models on relation extraction, we learn representations for different contexts explicitly, which is inspired by features in traditional relation classifiers. Empirical results show that the proposed model outperforms a strong logistic regression baseline, which uses handcrafted features as state-of-the-art supervised relation classifiers.

Third, we explore an unsupervised classifier to detect multiword opinion expressions. Given an expression, the classifier looks adjacent words and predicts whether it is an opinion expression. The new classifier helps us to discover opinion expressions which are not in general purpose opinion lexicons, and benefits the relation extractor.

We aim to make all algorithms simple, fast and scalable for large-scale corpus. Our system is tested on Amazon review data which contains 15 different domains and 33 million reviews. The output database contains 72.5 million pairs of opinion relations. Extensive experiments have been conducted on various aspects of the algorithm, and the performances of the proposed unsupervised models are even competitive with previous supervised models.

2 Related Works

Opinion relation extraction is an important task for fine-grained sentiment analysis. If human annotations are provided (e.g. MPQA corpus (Deng and Wiebe, 2015)), we could formulate the task into a supervised relation extraction problem as (Kobayashi et al., 2007; Johansson and Moschitti, 2013). Two types of models have been applied: pipeline models which first extract candidates of opinion expressions and targets then identify correct relations (Wu et al., 2009; Li et al., 2010; Yang and Cardie, 2012), and joint models which extract opinion expressions, targets and relations using a unified joint model (Yang and Cardie, 2013; Yang and Cardie, 2014). One consideration of applying supervised methods is their dependencies on the domains and human annotations.

Semi-supervised and unsupervised models are also applied for extracting opinion relations. Approaches include rule-based bootstrapping (Qiu et al., 2011), graph propagation algorithms (Xu et al., 2013; Liu et al., 2014; Brody and Elhadad, 2010), integer programming (Lu et al., 2011), and probabilistic topic models (Titov and McDonald, 2008; Mukherjee and Liu, 2012).

Our model is inspired by previous distantly supervised algorithms (Snow et al., 2004; Mintz et al., 2009). They use relations from WordNet or knowledge bases as distant supervision. Since we don't have similar resources for opinion relation extraction, we use patterns to generate relations. Neural network classifiers are popular for relation extraction recently. Many of them focus on fully supervised settings, recurrent neural networks (RNN) and convolutional neural networks (CNN) (Vu et al., 2016; Zeng et al., 2015; Xu et al., 2015a; Xu et al., 2015b; Zhang and Wang, 2015), sequence models and tree models

are investigated (Li et al., 2015; dos Santos et al., 2015). One similar network structure to our model is proposed in (Miwa and Bansal, 2016). They jointly extract entities and relations using two LSTM models. Another recent work (Jebbara and Cimiano, 2016) uses stacked RNNs and CNNs for aspect and opinion detection. Different from models there, we will learn representations for different lexical and syntactic features explicitly. Our formulation follows the features in traditional relation classifiers, which helps to interpret the learned vectors.

A closely related task is aspect-based opinion mining (Zhao et al., 2010; Yu et al., 2011; Wang et al., 2015). Instead of locating the opinion expressions, aspect-based opinion mining directly analyzes polarities of different opinion targets. The targets are usually constrained to be some predefined set. Shared tasks (SemEval2014, SemEval2015) have been held on the task, and various systems are proposed and evaluated (Pontiki et al., 2014; Pontiki et al., 2015). Comparing with aspect-based opinion mining, we will extract opinion expressions which are more informative, and we won't constrain opinion target types which helps us to handle open domain texts.

3 The Approach

Given an input sentence $s = w_1, w_2, \cdots, w_n$, where w_i is a word, the opinion relation extraction task outputs (O, T) pairs, where $O = w_i, w_{i+1}, \ldots, w_j$ is an opinion expression, $T = w_k, w_{k+1}, \ldots, w_l$ is an opinion target and the pair (O, T) is an opinion relation which asserts that opinion O is directed to target T [1]. Both O and T could be multiword expressions.

3.1 Patterns

Syntactic patterns have been shown to be effective for relation extraction. They are fast and can generalize well across domains, which are highly desirable for the open domain large-scale relation extraction task. However, despite of their advantages, two concerns are often raised: syntactic trees could be unreliable due to noise in texts and parsing errors, and the coverage of patterns is limited. To tackle the first problem, we deploy strong constraints on patterns in order to guarantee the quality of output. For the second problem, we enlarge the coverage by using a distantly supervised classifier (Section 3.2) and an opinion expression classifier (Section 3.3)

Table 1 lists the patterns used in our system. Like (Qiu et al., 2011), patterns are based on the dependency tree of input sentences, which basically capture adjective-noun, verb-complement and adverb-verb relations. The notation $w_1 \xrightarrow{l} w_2$ denotes that there is a dependency relation between word w_1 and w_2 with dependency relation type l. For example, the pattern P1 is activated if w_1 is the parent of w_2 in the dependency tree, and the dependency type is `amod` or `dep`.

In order to overcome noise and errors in dependency trees, we constrain all patterns by predefined part-of-speech (POS) tag sets and a general purpose opinion lexicon L. For example, the pattern P1 only accepts nouns and adjectives as arguments, and the adjectives are required to be an opinion word in L.

We also design the patterns to be able to handle multiword opinion expressions and targets (about 30% of all annotated expressions). Two cases are considered here. First, when two words match a pattern, we expand them to the smallest phrases containing them. It helps to collect some local contexts of opinion relations. For example, in pattern P4, the matching words "case" and "choice" are expanded to "the case" and "an excellent choice". Second, a relation pair could be compiled to a new opinion expression, which may have relations with other opinion targets. For example, in pattern C2, ("perfectly", "fit") is a relation, and it can be compiled into "fit perfectly" which appears in a new relation ("fit perfectly", "the case"). The compiled expressions can bring more informative relations which are ignored in previous works.

As an alternative of pattern matching, we also investigate the bootstrapping setting like (Qiu et al., 2011). In this setting, the algorithm is allowed to add new words to the opinion lexicon, and use the updated lexicon for successive pattern matching. While the bootstrapping could discover new opinion words which are not in the original lexicon, we find that the errors caused by newly added words are hard to control, and the advantages of bootstrapping are suppressed by the noise as the corpus becomes large. We will show (in the experiment section) that the accuracy drops 30% comparing with the direct pattern matching.

[1] We assume O, T are non-overlapped, and their distance in s is less than a threshold (10 in all experiments).

Name	Pattern	Output	Example
P1	$w_1 \xrightarrow{\text{amod}}_{\text{dep}} w_2$ w_1.pos $\in$ NOUN, w_2.pos $\in$ ADJ	$T = w_1$.np $O = w_2$	It is a [cool] case. case $\xrightarrow{\text{amod}}$ cool
P2	$w_1 \xrightarrow{\text{acomp}}_{\text{xcomp}} w_2$ w_1.pos $\in$ VERB, w_2.pos $\in$ ADJ	$T = w_1$.vp $O = w_2$	The case looks [great]. looks $\xrightarrow{\text{xcomp}}$ great
P3	$w_1 \xrightarrow{\text{advmod}} w_2$ w_1.pos $\in$ VERB, w_2.pos $\in$ADV	$T = w_1$ $O = w_2$	The cover matches [perfectly]. matches $\xrightarrow{\text{advmod}}$ perfectly
P4	$w_1 \xrightarrow{\text{nsubj}} w_2$ w_1.pos $\in$ NOUN, w_2.pos $\in$ NOUN or VERB or ADJ has a coplua verb between w_1 and w_2	$T = w_1$.np ($O = w_2$.np $O = w_2$.vp $O = w_2$.adjp)	This case is [an excellent choice]. case $\xleftarrow{\text{nsubj}}$ choice
C1	$w_1 \xleftarrow{\text{conj}} (O', T')$ O'.pos. $\in$ ADJ or ADV w_1.pos $\in$ ADJ or ADV	$T = T'$ $O = w_1$	The case looks [great] and very [cute]. cute $\xleftarrow{\text{conj}}$ (great, looks)
C2	$(O', T') \xrightarrow{\text{nsubj}} w_1$ w_1.pos $\in$ NOUN, T'.pos $\in$ VERB O'.pos $\in$ ADV, T' and O' are adjacent	$T = w_1$ $O = T'O'$	The case [fits perfectly]. (perfectly, fits) $\xrightarrow{\text{nsubj}}$ the case

Table 1: Syntactic patterns in the system. We denote POS tag sets: NOUN ={NN, NNS}, VERB = {VB, VBD, VBN, VBP, VBZ}, ADJ = {JJ, JJR, JJS}, ADV = {RB, RBR, RBS}. w_i.pos is the POS tag of w_i. w_i.np (w_i.vp, w_i.adjp) is the smallest noun (verb, adjective) phrase containing w_i (return w_i if no such phrase exists). T, T' are opinion targets, O, O' are opinion expressions. "$(O', T') \rightarrow$" and "$\leftarrow (O', T')$" represent dependency relations on words O' and T' respectively.

3.2 Distant Supervision

Despite of the high precision, one well-known disadvantage of pattern-based methods is the low coverage. Consider the following example,

> Ordered the k9ballistics Crate Pad and I am [so pleased].

No pattern in Table 1 is applicable on relation ("so pleased", "Ordered the k9ballistics Crate Pad"), although it could be inferred from the context. In fact, the two expressions are close in distance, and "Ordered the k9ballistics Crate Pad " is the only possible object of "please" in the sentence. Many similar cases appear in online review corpus which downgrade the performance of patterns. To further explore those relations, we develop distantly supervised classifiers to integrate various lexical and syntactic contexts. Our experiments show that classifiers help to increase coverage of patterns by 20%.

Given a candidate relation $x = (O, T)$ in sentence s, the classifier outputs probability $p(y|x), y \in \{1, -1\}$ telling whether x is a valid opinion relation. Since manually labelled corpus are costly and difficult to obtain for open domains, we would prefer unsupervised classifiers. On the other side, the pattern matching can generate a set of opinion relations without any human annotations. Although the relations are not completely correct, they are almost free to collect and easily amount to a large set. Thus, we can take the relations from the pattern matching as the distant supervision, and hope the broad coverage could overcome the noise.

Formally, we take all relations extracted by patterns as positive samples. For each positive sample (O, T), we add a negative sample (O, T') for $T' \neq T$ (T' is NP, VP or ADJP). Similarly, we add negative samples (O', T) for all $O' \neq O$. At test time, we consider all VP and ADJP in s which contain at least one word in the general opinion lexicon L as candidate opinion expressions, all NP and VP as candidate opinion targets, and all possible pairs between them are candidate relations.

Our distantly supervised classifier is based on a neural network. Different from most previous deep learning models, the classifier learns representations for different lexical and syntactic contexts explicitly, which is inspired by features in traditional (non-neural-network-based) relation classifiers. We would observe from experiments that knowledge form previous feature engineering works can help neural network models to achieve better performances. Before explaining the model, we refresh some notations first. For $x = (O, T)$ in sentence s, where

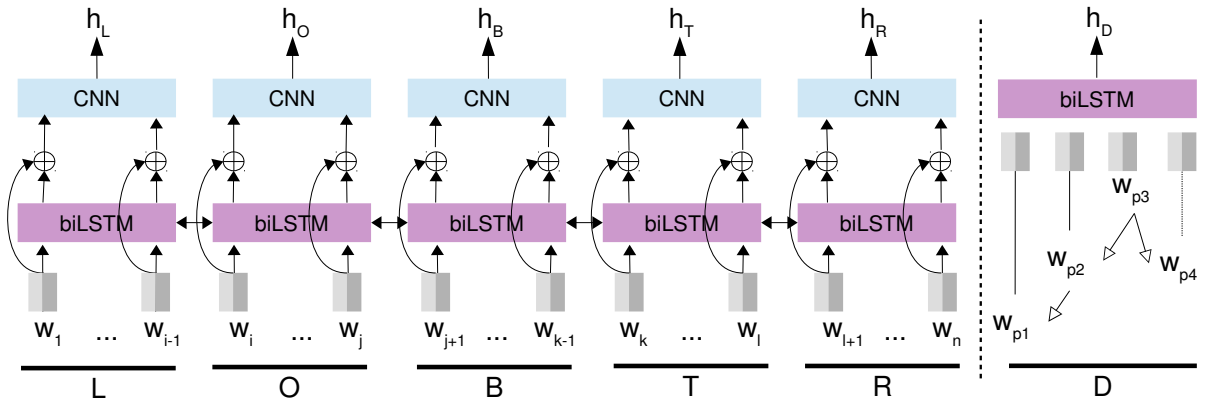

Figure 1: Representation learning of lexical and syntactic contexts. The left hand side is the five CNNs and the sentence level biLSTM for lexical contexts. For a word w, operator "$\oplus$" concatenates word/POS embeddings of w and the output vectors of the biLSTM on w. The right hand side is the biLSTM for syntactic contexts. $w_{p1}, \ldots, w_{p4}$ is the dependency path D, w_{p3} is the lowest common ancestor of head words of O and T.

$O = w_i, \ldots, w_j, T = w_k, \ldots, w_l$, denote other parts of s to be $L = w_1, \ldots, w_{i-1}, B = w_{i+1}, \ldots, w_{k-1}, R = w_{l+1}, \ldots, w_n$, where L, R are the left and right contexts of x, B is the context between O and T [2]. Let $D = w_{p1}, \ldots, w_{pm}$ be the path connecting head words of O and T in the dependency tree.

To capture lexical contexts of x, we use five convolutional neural networks (CNN) to learn representations for L, O, B, T, R. Take B as an example, the output $\mathbf{h}_B \in \mathbf{R}^d$ is the result of a single layer convolution of inputs with max-pooling. The input of the CNN includes word and POS tag embeddings of $w_{i+1}, \ldots, w_{k-1}$. The five local CNN models can be independently trained before making the final predictions, however, it ignores global information of the sentence, and also the potential sharing of features among local models. In order to incorporate global structures, we build the five local models on top of a sentence level bidirectional long short term memory network (biLSTM). The recurrent structure and memory mechanism of biLSTM can propagate and share long distance features of s. We take outputs of memory cells as inputs of the CNN models (in addition to the embeddings). All representations $\mathbf{h}_L, \mathbf{h}_O, \mathbf{h}_B, \mathbf{h}_T, \mathbf{h}_R$ use the same network structure in our experiments. Finally, to make a prediction on y, we use a softmax function $p(y|x) = \frac{1}{Z} \exp\{\theta^\intercal \Phi(x, y)\}$ on the weighted averaged feature vector $\Phi(x, y)$,

$$\Phi(x, y) = a_L \mathbf{h}_L + a_O \mathbf{h}_O + a_B \mathbf{h}_B + a_T \mathbf{h}_T + a_R \mathbf{h}_R,$$

where $a_L, a_O, a_B, a_T, a_R \in \mathbf{R}, \theta \in \mathbf{R}^d$ are parameters of the model.

We also try to incorporate dependency path D into the model as suggested by (Xu et al., 2015a; Xu et al., 2015b). We use a similar bidirectional LSTM like (Xu et al., 2015b), and concatenate the final outputs of the forward LSTM and the backward LSTM to get feature representation $\mathbf{h}_D \in \mathbf{R}^d$. The experiments show that, however, $\mathbf{h}_D$ can not get further performance gains. We suspect that the errors from dependency parsing limited the contribution of this feature.

3.3 Opinion Expression Classifier

In above pattern matching and distant supervision algorithms, a candidate opinion expression is extracted if it contains at least one opinion word in the general purpose opinion lexicon L. Although the simple approach helps to handle multiword expressions, some expressions could also be ignored since they have no opinion words in L or their POS tags are wrongly assigned. In this section, we introduce our unsupervised opinion expression classifier, which predicts whether a phrase is an opinion expression based on its contexts.

Formally, for a candidate expression $O = w_i, \ldots, w_j$ in sentence s, we use context words $w_{i-c}, \ldots, w_{i-1}$ and $w_{j+1}, \ldots, w_{j+c}$ as inputs of a CNN classifier (c is the context window size, and

[2] For simplicity, we assume O appears before T. In the implementation, an indicator dimension is set to identify whether O appears first.

we set it to 5 in all experiments). After the convolution layer, max-pooling and softmax (similar to the CNNs in Section 3.2), the classifier outputs probability $p(z|O)$ where $z \in \{-1, 1\}$ indicates whether O is a valid opinion expression. In order to get the training set, we rely on the lexicon L. Given the unlabelled corpus and $w \in L$, we consider each appearance of w in the corpus as a positive example, and other randomly chosen words as negative examples.

To apply the opinion expression classifier in the opinion relation classifier, we add expressions which have $p(z|O)$ greater than some threshold γ to the relation classifier.

4 Experiments

4.1 Configurations

We extract opinion relations on a subset of Amazon product review corpus provided by (McAuley et al., 2015), which contains 15 domains and 33 million reviews. The statistics of extracted relations are in Table 2.

For quantitative evaluation, we select four domains (Cell Phones, Movie and TV, Food, Pet Supplies) for detailed analyses. We manually label all correct opinion relations in 1000 sentences, and select 200 sentences as the development set, the rest 800 as the test set [3]. Furthermore, to compare with previous supervised methods, we also conduct experiments on USAGE corpus (Klinger and Cimiano, 2014) which annotates 4481 opinion relations for 8 products.

We use NLTK (Bird et al., 2009) for sentence splitting and word segmentation, Stanford parser [4] for getting POS tags, phrase chunks and dependency trees, and scikit-learn toolkit (Pedregosa et al., 2011) and TensorFlow [5] for machine learning algorithms. The general purpose opinion lexicon is from (Wilson et al., 2005).

4.2 Main Results

Table 4 shows results on four domains. The methods for comparison are:

- **Adjacent** is a simple baseline system from (Hu and Liu, 2004). It first identifies words in the general purpose opinion lexicon, then finds the nearest noun or verb phrase to them as their opinion targets.

[3]https://github.com/AntNLP/OpinionRelationCorpus
[4]http://nlp.stanford.edu/software/lex-parser.shtml
[5]https://www.tensorflow.org/

Domain	#Reviews	#Sents	#Relations
Cell Phones	3.4	19.1	6.7
Movie and TV	4.6	47.6	15.3
Food	1.3	7.6	2.5
Pet Supplies	1.2	8.0	2.4
Automotive	1.4	9.5	2.6
Digital Music	0.8	7.3	2.4
Beauty	2.2	6.4	3.8
Toys and Games	2.3	11.7	3.8
Instruments	0.5	13.5	1.4
Office Products	1.2	4.1	2.6
Patio	1.0	8.3	2.0
Baby	0.9	6.5	1.8
Clothing	5.7	29.1	10.2
Sports	3.3	20.2	7.1
Kindle	3.2	25.0	7.9
All	33	223.9	72.5

Table 2: Statistics of the opinion relation database. The relations are extracted by patterns in Table 1 and normalized by removing leading articles, pronouns and copulas of opinion expressions and targets. All numbers are in 10^6.

- **Bootstrapping** reimplements the double propagation in (Qiu et al., 2011), which is the state-of-the-art unsupervised opinion relation extraction algorithm. It also uses a set of patterns, but adds new opinion words discovered by the patterns to the existing lexicon on the fly. The updated lexicon is then used in following bootstrapping iterations.

- **Pattern** is the pattern matching method in Section 3.1.

- **LR** is a logistic regression trained with the same distant supervision as Section 3.2. We use standard relation extraction features (Table 3), which are used in state-of-the-art supervised relation classifiers (Mintz et al., 2009) [6].

- **NN** is our neural network model in Section 3.2. We set the dimension d of outputs (e.g., $\mathbf{h}_B$) be 128, the output dimension of biLSTM be 128, the dimension of word/POS tag embeddings be 300. We use three convolution kernels with window size 1, 2, 3, and initialize word embeddings with pre-trained word vectors from word2vec tools [7]. By default, we use the five CNNs on the sentence level biLSTM and not include dependency path $\mathbf{h}_D$ and the opinion expression classifier (the configuration of "NN" equals "biLSTM+LOBTR" in Table 5). In order to build

[6]We select the features on the development set.
[7]https://code.google.com/archive/p/word2vec/

Lexical Features
① POS tag sequences of O and T.
② The length of O and T.
③ The distance between O and T.
④ The word sequence between O and T in s.
⑤ POS tags of words between O and T in s.
⑥ Words, POS tags of w_{i-1}, w_{i-2}, w_{j+1}, w_{j+2}.
⑦ Words, POS tags of w_{k-1}, w_{k-2}, w_{l+1}, w_{l+2}.
⑧ Combined POS tags of O and T.
Syntactic Features
⑨ Does a dependency relation exist between O and T.
⑩ The dependency path between O and T.
⑪ The length of the dependency path.

Table 3: Features in logistic regression.

the training set, we run the pattern matching on 6×10^4 unlabelled sentences.

- **NN+Pattern** stacks results of "Pattern" and "NN".

We have several observations on Table 4. First, performances of "Adjacent" are poor, which means that we do need some advanced linguistic features for the task. Second, "Bootstrapping" underperforms "Pattern" in four domains. We examine the outputs of "Bootstrapping" and find that the newly added words bring a lot of noise into the opinion lexicon, which affect the accuracy negatively. Third, while "Pattern" has the highest precision in all systems, distantly supervised methods ("LR" and "NN") help to improve recall and achieve better F1 values (except on the Pet domain). Hence, based on the distant supervision from patterns, classifiers cover more correct relations. Regarding the Pet domain, the precision of "Pattern" is low, so the number of errors in training set of "LR" and "NN" is large, and one could fail to learn reliable models. Fourth, the neural network model "NN" outperforms traditional classifier "LR" on all domains, which shows that learning feature representations has some advantages than handcrafting features on our experiment settings. Finally, simply stacking the results of "Pattern" and "NN" can improve the overall scores.

Next, we test our neural network model with various settings in Table 5. First, we compare models with different configurations on CNNs in row 1 to row 3. The setting "biLSTM+B" only uses the CNN corresponding to the words in B (i.e., the words between O and T); "biLSTM+OBT" uses three CNN on words in B, O, T; "biLSTM+LOBTR" involves all five CNNs (equals to "NN" in Table 4). We see that, in general, the performances (especially recalls) increase as we introduce more CNNs. However, the dependency path feature $\mathbf{h}_D$ ("biLSTM+LOBTR+D" in row 4) won't help to get further improvements. Second, in row 5, we drop the sentence level biLSTM and only use the five CNNs, and observe some loss on performances compared with row "biLSTM+LOBTR". Hence, the long distance information provided by the biLSTM is also helpful. Third, we test the opinion expression classifier in the last two rows. Recall that γ is the threshold that controls the output of the classifier. The result shows that when $\gamma = 0.8$, new opinion expressions added by the classifier can improve the scores, but when $\gamma = 0.5$, the noise can overwhelm the gains. We futher show the precision-recall curves of "NN" and "LR" in the case of $\gamma = 0.8$ in Figure 2. Some interesting opinion expressions added by the classifier are "not even enough", "became extremely hot", "*just* enough", "arrived damaged", "looks cool n cute", which shows that the classifier could both discover opinion expressions without words in general purpose lexicons, and have some tolerance to noise.

4.3 Results on USAGE Corpus

In order to compare with fully supervised methods, we evaluate our models on USAGE corpus in Table 6. To build the distantly supervised models, we use untagged reviews which are about the 8 products of USAGE. The baseline systems are (Klinger and Cimiano, 2014) and (Jebbara and Cimiano, 2016), which are state-of-the-art systems on the dataset.

Results show that, with the same setting ("gold") [8], our fully unsupervised models achieve competitive precision scores against previous fully supervised methods. By examining the outputs, one reason for the performance gaps on recall may be the differences of annotation guidelines between USAGE and our dataset. For example, we don't annotate pronouns as opinion targets while USAGE does (e.g., ("love", "it") is a proper annotation in USAGE). We can also observe that distant supervisions provide notable performances gains than direct pattern matching.

[8] Both baselines don't report end-to-end performances. As a reference, in (Jebbara and Cimiano, 2016), the F1 of opinion expression and target extraction are 50% and 67%.

System	Phone			Movie			Food			Pet		
	P	R	F	P	R	F	P	R	F	P	R	F
Adjacent	38.6	65.7	48.6	30.0	58.8	39.7	31.4	46.5	37.5	28.4	62.2	39.0
Bootstrapping	44.0	62.9	51.8	26.9	49.3	34.8	43.6	54.0	48.2	33.6	57.7	42.5
Pattern	**69.4***	64.4	61.1	**62.2***	42.4	50.4	**76.0***	41.9	54.1	**59.9***	51.3	55.3*
LR	60.1	64.7	62.4	55.6	57.0	55.3	65.5	49.2*	56.2	47.6	**59.3***	52.8
NN	63.4	67.9*	65.6*	56.8	58.2*	57.5*	70.5	47.7	56.9*	51.4	58.0	54.5
NN+Pattern	64.4	**70.5**	**67.3**	58.2	**59.9**	**59.1**	68.4	**50.8**	**58.3**	54.9	58.2	**56.5**

Table 4: Comparison with different baseline systems. The dark entries are the highest scores among all systems, and "*" indicates the highest scores excluding "NN+Pattern".

System	Phone			Movie			Food			Pet		
	P	R	F	P	R	F	P	R	F	P	R	F
biLSTM +B	59.8	**69.5**	64.3	54.7	**59.1**	56.8	64.8	**48.5**	55.5	47.4	57.2	51.8
biLSTM +OBT	63.4	66.7	65.3	59.8	58.2	58.9	68.6	46.8	55.6	56.2	56.9	56.5
biLSTM +LOBTR	63.4	67.9	65.6	56.8	58.2	57.5	66.5	47.7	55.5	51.4	**58.0**	54.5
biLSTM +LOBTR+D	65.5	61.3	63.4	59.5	55.8	57.6	69.1	46.8	55.8	54.9	57.7	56.3
LOBTR	64.0	65.3	64.7	57.5	56.7	57.1	**70.6**	43.1	53.5	54.3	57.7	55.9
$\gamma = 0.5$	64.2	64.7	64.5	56.2	57.9	57.0	65.5	45.5	53.7	**61.6**	54.3	**57.7**
$\gamma = 0.8$	**65.6**	66.1	**65.9**	**61.1**	58.2	**59.6**	67.1	48.2	**56.1**	58.6	50.8	54.4

Table 5: Different settings of the proposed model.

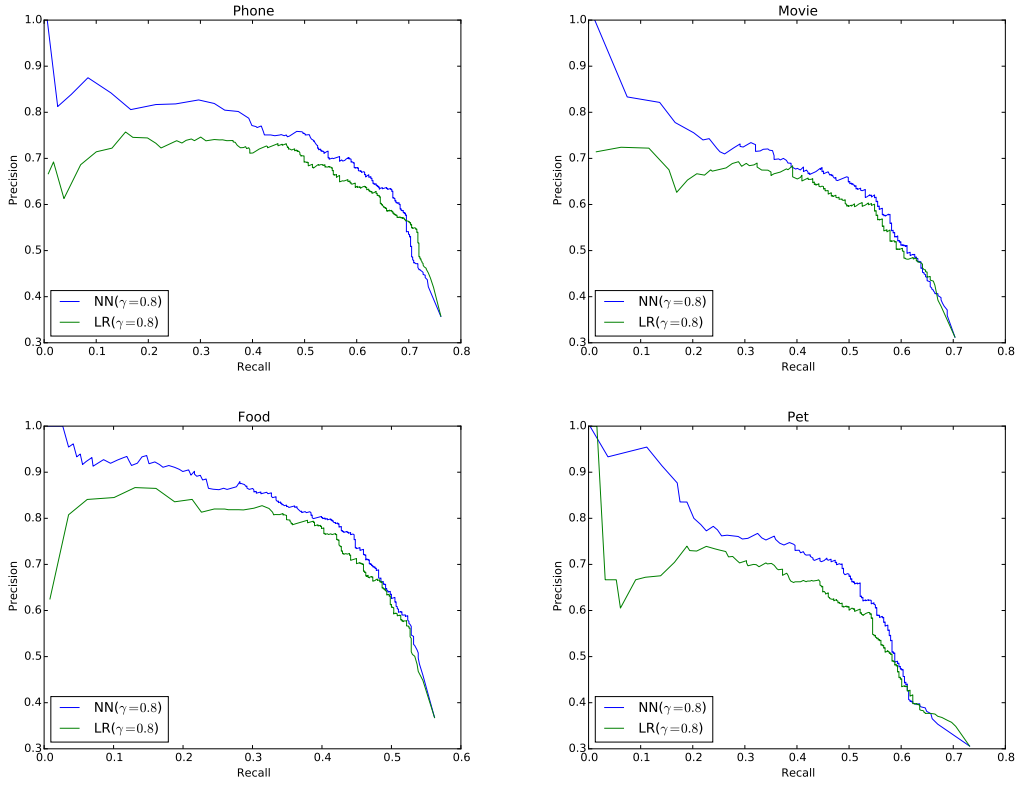

Figure 2: Precision recall curves for $\gamma = 0.8$.

4.4 Error Analysis

Finally, we do some error analyses on the extracted relations. Taking movie domain as example, we find that for movie reviews, the opinions from reviewers are mixed with plot and characters of movies, which makes it difficult to distinguish opinions and background topics with our simple opinion relation definition. For example,

Systems	P	R	F
Klinger et al. (2014)	-	-	65.0
Jebbara et al. (2016)	87.0	75.0	81.0
Pattern	51.4	20.7	29.5
LR (end-to-end)	49.5	27.8	35.6
NN (end-to-end)	43.3	40.1	41.6
LR (gold)	89.1	47.9	62.3
NN (gold)	81.4	62.8	70.9

Table 6: Results on USAGE corpus. First two rows are state-of-the-art systems on the dataset. Both of them assume gold annotations on opinion expressions and targets have been given. We report results with identical settings ("gold"), and also "end-to-end" results in which no gold annotations are provided.

in "Also said repeatedly how Tojo was [loyal to Emperor Hirohito]", the word "loyal" indicates a wrong opinion relation since it's a description of the story. A true comment from the reviewer is behind the word "repeatedly", which is hard to be expressed with our opinion relations. We plan to introduce both more background knowledge and more powerful relation types in future work.

5 Conclusion

We investigate the task of large-scale opinion relation extraction. Our algorithm first uses syntactic patterns to get a set of opinion relations, then builds a neural network classifier based on these relations. We also develop an opinion expression classifier to better extract opinion words. Extensive experiments on Amazon review data show the effectiveness of the proposed methods.

6 Acknowledgements

The authors wish to thank the reviewers for their helpful comments and suggestions. This research is supported by NSFC (61402175, 61473092) and STCSM (15ZR1410700, 14DZ2260800). Yuanbin Wu is supported by Microsoft Research Asia Collaborative Research Program. The corresponding authors are Yuanbin Wu, Man Lan and Shiliang Sun.

References

Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit*. O'Reilly.

Samuel Brody and Noemie Elhadad. 2010. An unsupervised aspect-sentiment model for online reviews. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 804–812, Los Angeles, California, June. Association for Computational Linguistics.

Lingjia Deng and Janyce Wiebe. 2015. Mpqa 3.0: An entity/event-level sentiment corpus. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1323–1328, Denver, Colorado, May–June. Association for Computational Linguistics.

Cicero dos Santos, Bing Xiang, and Bowen Zhou. 2015. Classifying relations by ranking with convolutional neural networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 626–634, Beijing, China, July. Association for Computational Linguistics.

Murthy Ganapathibhotla and Bing Liu. 2008. Mining opinions in comparative sentences. In *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, pages 241–248.

Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proc. of SIGKDD*, pages 168–177.

Soufian Jebbara and Philipp Cimiano. 2016. Aspect-based relational sentiment analysis using a stacked neural network architecture. In *ECAI*, pages 1123–1131.

Richard Johansson and Alessandro Moschitti. 2013. Relational features in fine-grained opinion analysis. *Computational Linguistics*, 39(3):473–509.

Roman Klinger and Philipp Cimiano. 2014. The USAGE review corpus for fine grained multi lingual opinion analysis. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC-2014)*, pages 2211–2218.

Nozomi Kobayashi, Kentaro Inui, and Yuji Matsumoto. 2007. Extracting aspect-evaluation and aspect-of relations in opinion mining. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 1065–1074, Prague, Czech Republic, June. Association for Computational Linguistics.

Fangtao Li, Chao Han, Minlie Huang, Xiaoyan Zhu, Yingju Xia, Shu Zhang, and Hao Yu. 2010. Structure-aware review mining and summarization. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 653–661.

Jiwei Li, Thang Luong, Dan Jurafsky, and Eduard Hovy. 2015. When are tree structures necessary for deep learning of representations? In *Proc. of EMNLP*, pages 2304–2314.

Kang Liu, Liheng Xu, and Jun Zhao. 2014. Extracting opinion targets and opinion words from online reviews with graph co-ranking. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 314–324, Baltimore, Maryland, June. Association for Computational Linguistics.

Yue Lu, Malú Castellanos, Umeshwar Dayal, and ChengXiang Zhai. 2011. Automatic construction of a context-aware sentiment lexicon: an optimization approach. In *Proc. of WWW*, pages 347–356.

Julian J. McAuley, Rahul Pandey, and Jure Leskovec. 2015. Inferring networks of substitutable and complementary products. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Sydney, NSW, Australia, August 10-13, 2015*, pages 785–794.

Mike Mintz, Steven Bills, Rion Snow, and Daniel Jurafsky. 2009. Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 1003–1011, Suntec, Singapore, August. Association for Computational Linguistics.

Makoto Miwa and Mohit Bansal. 2016. End-to-end relation extraction using lstms on sequences and tree structures. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1105–1116, Berlin, Germany, August. Association for Computational Linguistics.

Arjun Mukherjee and Bing Liu. 2012. Aspect extraction through semi-supervised modeling. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 339–348, Jeju Island, Korea, July. Association for Computational Linguistics.

Ramanathan Narayanan, Bing Liu, and Alok Choudhary. 2009. Sentiment analysis of conditional sentences. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 180–189, Singapore, August. Association for Computational Linguistics.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *JMLR*, 12:2825–2830.

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

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

Guang Qiu, Bing Liu, Jiajun Bu, and Chun Chen. 2011. Opinion word expansion and target extraction through double propagation. *Computational Linguistics*, 37(1):9–27.

Ellen Riloff and Janyce Wiebe. 2003. Learning extraction patterns for subjective expressions. In Michael Collins and Mark Steedman, editors, *Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing*, pages 105–112.

Rion Snow, Daniel Jurafsky, and Andrew Y. Ng. 2004. Learning syntactic patterns for automatic hypernym discovery. In *NIPS*, pages 1297–1304.

Ivan Titov and Ryan McDonald. 2008. A joint model of text and aspect ratings for sentiment summarization. In *Proceedings of ACL-08: HLT*, pages 308–316, Columbus, Ohio, June. Association for Computational Linguistics.

Ngoc Thang Vu, Heike Adel, Pankaj Gupta, and Hinrich Schütze. 2016. Combining recurrent and convolutional neural networks for relation classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 534–539, San Diego, California, June. Association for Computational Linguistics.

Linlin Wang, Kang Liu, Zhu Cao, Jun Zhao, and Gerard de Melo. 2015. Sentiment-aspect extraction based on restricted boltzmann machines. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 616–625, Beijing, China, July. Association for Computational Linguistics.

Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. 2005. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 347–354, Vancouver, British Columbia, Canada, October. Association for Computational Linguistics.

Yuanbin Wu, Qi Zhang, Xuangjing Huang, and Lide Wu. 2009. Phrase dependency parsing for opinion mining. In *Proceedings of the 2009 Conference on*

Empirical Methods in Natural Language Processing, pages 1533–1541, Singapore, August. Association for Computational Linguistics.

Yuanbin Wu, Qi Zhang, Xuanjing Huang, and Lide Wu. 2011. Structural opinion mining for graph-based sentiment representation. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 1332–1341, Edinburgh, Scotland, UK., July. Association for Computational Linguistics.

Liheng Xu, Kang Liu, Siwei Lai, Yubo Chen, and Jun Zhao. 2013. Mining opinion words and opinion targets in a two-stage framework. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1764–1773, Sofia, Bulgaria, August. Association for Computational Linguistics.

Kun Xu, Yansong Feng, Songfang Huang, and Dongyan Zhao. 2015a. Semantic relation classification via convolutional neural networks with simple negative sampling. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 536–540, Lisbon, Portugal, September. Association for Computational Linguistics.

Yan Xu, Lili Mou, Ge Li, Yunchuan Chen, Hao Peng, and Zhi Jin. 2015b. Classifying relations via long short term memory networks along shortest dependency paths. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1785–1794, Lisbon, Portugal, September. Association for Computational Linguistics.

Bishan Yang and Claire Cardie. 2012. 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*, pages 1335–1345, Jeju Island, Korea, July. Association for Computational Linguistics.

Bishan Yang and Claire Cardie. 2013. Joint inference for fine-grained opinion extraction. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1640–1649, Sofia, Bulgaria, August. Association for Computational Linguistics.

Bishan Yang and Claire Cardie. 2014. Joint modeling of opinion expression extraction and attribute classification. *Transactions of the Association of Computational Linguistics*, 2:505–516.

Jianxing Yu, Zheng-Jun Zha, Meng Wang, Kai Wang, and Tat-Seng Chua. 2011. Domain-assisted product aspect hierarchy generation: Towards hierarchical organization of unstructured consumer reviews. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 140–150, Edinburgh, Scotland, UK., July. Association for Computational Linguistics.

Daojian Zeng, Kang Liu, Yubo Chen, and Jun Zhao. 2015. Distant supervision for relation extraction via piecewise convolutional neural networks. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1753–1762, Lisbon, Portugal, September. Association for Computational Linguistics.

Dongxu Zhang and Dong Wang. 2015. Relation classification via recurrent neural network. *CoRR*.

Xin Zhao, Jing Jiang, Hongfei Yan, and Xiaoming Li. 2010. Jointly modeling aspects and opinions with a MaxEnt-LDA hybrid. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 56–65, Cambridge, MA, October. Association for Computational Linguistics.