Is "hot pizza" Positive or Negative? Mining Target-aware Sentiment Lexicons

Jie Zhou[1,2], Yuanbin Wu[2], Changzhi Sun[3], and Liang He[1,2]

[1]Shanghai Key Laboratory of Multidimensional Information Processing
East China Normal University, China
[2]School of Computer Science and Technology East China Normal University, China
[3]ByteDance AI Lab
jzhou@ica.stc.sh.cn, {ybwu,lhe}@cs.ecnu.edu.cn
sunchangzhi@bytedance.com

Abstract

Modelling a word's polarity in different contexts is a key task in sentiment analysis. Previous works mainly focus on domain dependencies, and assume words' sentiments are invariant within a specific domain. In this paper, we relax this assumption by binding a word's sentiment to its collocation words instead of domain labels. This finer view of sentiment contexts is particularly useful for identifying commonsense sentiments expressed in neutral words such as "big" and "long". Given a target (e.g., an aspect), we propose an effective "perturb-and-see" method to extract sentiment words modifying it from large-scale datasets. The reliability of the obtained target-aware sentiment lexicons is extensively evaluated both manually and automatically. We also show that a simple application of the lexicon is able to achieve highly competitive performances on the unsupervised opinion relation extraction task.

1 Introduction

Sentiments of words can be subtle. We are used to using the same word to express different emotions in different contexts. "Hot", for example, suggests a negative sentiment when commenting a computer hardware and a positive sentiment when commenting a pizza, even itself alone is identified without any general orientation. In these situations, it is the composition of a word, contexts, and commonsense carries an opinion. Automatically detecting such context dependent sentiments would strengthen both our understanding of implicit opinions in languages and improve existing sentiment analyses models, which is the main topic of this work.

To handle shifts of word sentiment, prior works studied how to adapt existing sentiment lexicons to new domains (Hamilton et al., 2016; Xing et al., 2019). By modeling differences and similarities of text topics, they can detect new sentiments of words as the domain changes. The basic assumption of those domain-level sentiment lexicons is that a word keeps a consistent sentiment within a domain. This assumption, however, might be strong for fine-granularity analyses of text sentiments: words (especially, neural words such as "long", "fast") could exhibit different orientations even in the same domain (Figure 1).

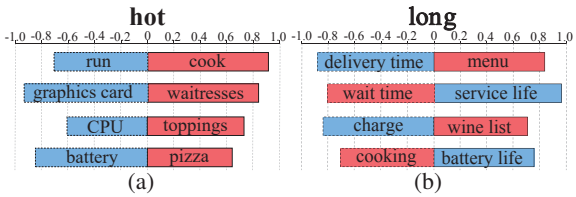


Figure 1: Visualization of real-world commonsense sentiment of "hot" and "long" extracted by our framework. Red and blue indicate the targets in restaurant and electronic domains, respectively.

To collect more detailed information of a sentiment, another branch of works (aspect-based sentiment analysis (Pontiki et al., 2014; Zhou et al., 2020a,b), opinion relation extraction (Sun et al., 2017)) attempt find answers of "who express what opinion on which target" for opinion bearing texts. Existing solutions heavily rely on manual annotations and linguistic rules, which are either hard to scale-up or hard to be complete.

In this work, we study the task of extracting *target-aware* sentiment lexicons. An entry of such lexicon is a pair of a sentiment word and a target word, and their collocation expresses a sentiment. It improves existing domain-dependent lexicons by being more concrete and accurate on describing opinions. Departing from approaches adopted in existing aspect-based analyses, we aim to build context-aware lexicons by minimizing the requirement of annotations (e.g., only document-

Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics, pages 608–618
April 19 - 23, 2021.

level sentiment labels) and errors from handcrafted patterns. Our method starts from a target word (e.g., an aspect in product reviews), and extract sentiment words from its local context. The main strategy is to perturb context words and see how the sentiment of the target word changes: words with high influence on the target's sentiment hold high probability of forming a collocation with the target. We accomplish this by observing the behaviour of a well-trained document-level sentiment classifier when we change the contexts of the target word. Two types of perturbations are examined, *discrete perturbation* which only requires a black-box classifier, and *continuous perturbation* which asks for network gradients. We collect evidences of each candidate pair on large datesets to ensure the reliability of the final lexicon. Finally, the polarities of a lexicon entry can also be obtained by querying the sentiment classifier.

On two online product review domains (electronic and restaurant), we evaluate the extracted target-aware lexicon both manually and automatically. Quantitative and qualitative results show that the lexicons are reasonable to reflect common sentiment usage in each domain. As an application, we apply the lexicons to the task of unsupervised opinion relation extraction. The model performs significantly better than the baseline extractor, and even competitive with a recent supervised model on restaurant reviews. We summarize main contributions as follows,

- We propose to extend general purpose opinion lexicons with target constraints which provides a finer view on word-level sentiments.
- We develop a scalable approach to automatically mine target-aware sentiment lexicon from texts without extensive annotations and elaborated linguistic rules.
- Besides manual evaluations, we propose an automatic way to evaluate the extracted lexicon with downstream tasks.
- We are able to achieve significant improvements on unsupervised opinion relation extraction task with the help of the new lexicons.

2 Definitions and the Task

Let d be a document with sentences $s_1, s_2, ..., s_{|d|}$ and $y \in Y$ be the sentiment label of d. [1] Given a corpus $\mathcal{D} = \{(d_i, y_i)\}_{i=1}^{|\mathcal{D}|}$ and a target word t (e.g., screen, pizza), [2] our task is to extract *target-aware opinion words* of t only using document-level sentiment labels. Precisely, we aim to output a set of triples $(t, o, \mathbf{p})$, where o is an opinion word commonly used to comment target t and $\mathbf{p} \in \mathbf{R}^{|Y|}$ is the distribution of its sentiment orientation.

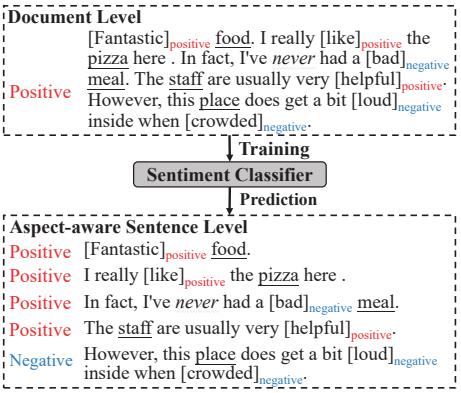


Figure 2: The process of distant supervision.

We develop the lexicon extractor in three steps. First, we build an approximate target-level sentiment classifier (Section 3) using document-level sentiment labels. Second, for each sentence s containing target t, we calculate how important a word $w \in s$ is on helping the classifier correctly predicting s's polarity (Section 4.1 and 4.2). We aggregate scores of w over all its occurrences to get its confidence of being an opinion word of t. Finally, we derive the polarity of w by querying the classifier with template sentences (Section 5).

3 Approximating Target-oriented Opinion

To identify target-aware opinion words, our key approach is to inspect how the opinion of a target changes when its context words change. Hence, it is crucial to know the polarity of a target in documents. However, annotations in D are document-level: for a document, its sentiment label expresses overall sentiments for all targets in the document, rather than a specific one. For example, the restaurant review in Figure 2 talks about 5 targets, each of them is commented by different opinion words with different polarities. In one of our datasets, 93% of documents contain multiple sentences (6 in average), and more than 82% contain multiple targets (7 in average). Therefore, directly using

[1] We use the 5-level label set $Y = \{1, 2, 3, 4, 5\}$ (the larger a number, the more positive it represents).

[2] Here we mainly focus on online reviews, but the methods could be applied to other sentiment-bearing texts.

document-level sentiment labels could be inappropriate for target-level analyses. On the other hand, it is quite expensive to annotate target-level sentiments, and existing datasets are far from enough for a robust commonsense opinion extractor.

To deal with this problem, we borrow the idea of distant supervision (Mintz et al., 2009): if a document is labelled as positive, at least one sentence (target) in it is positive. By seeing a large amount of positive documents, a classifier may be able to generalize patterns of their positive sentences, thus may help finding sentence-level (target-level) opinions. Here we simply build a document-level sentiment classifier, and apply it on sentences to get pseudo target-level sentiment labels (for simplicity, we assume one sentence contains one target). Advanced distant supervision models could also be applied, but we find this simple method preforms quite well in our experiments.

To build the sentiment classifier, we fine-tune BERT (Devlin et al., 2019) on $\mathcal{D}$ to encode domain specific semantics and augment it with a sentiment prediction task to encode sentiment information. For a document d, we feed its word sequence into BERT and obtain a vector representation $\mathbf{d} = \text{BERT}(d)$, then we apply a softmax operator on $\mathbf{d}$ to get the probability of its sentiment $P(y|d)$,

$$P(y|d) = \text{softmax}(W_c\mathbf{d} + b_c), \quad (1)$$

where W_c, b_c are new parameters for the sentiment classification task. The loss function is the cross-entropy between the predicted probability and the true label,

$$L = -\frac{1}{|\mathcal{D}|}\sum_{i=1}^{|\mathcal{D}|}\log P(y_i|d_i). \quad (2)$$

For each sentence s containing t, we apply above classifier to predict pseudo sentiment label y_p of s. In the following sections, we will rely on the set $S_t = \{(s, y_p)|t \in S\}$ to extract target-aware opinion words of t.

4 Importance Scores

We propose two score functions for measuring a context word w's influence on the target-oriented sentiment: one is *discrete perturbation* which only requires outputs of the sentiment classifier, another is *continuous perturbation* which needs network gradients. They are also called model-free and model-based methods, respectively. Both of them are simple and easy to compute given the trained model, and thus suitable for large-scale collective analyses.

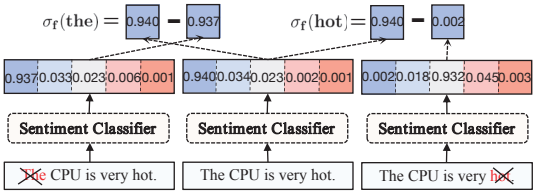


Figure 3: The possibility of the sentence is super negative changed from 0.940 to 0.002 when the word "hot" is deleted.

4.1 Discrete Perturbation

A well-trained sentiment classifier should correctly capture correlations between sentence words and sentence polarities. Intuitively, an opinion word (of the target) would have high influence on the sentiment distribution $P(y_p|s)$. For example, in Figure 3, "hot" is more informative than "The" for predicting the sentence's negative label.

In order to see whether a word w affects $P(y_p|s)$, we perturb the sentence s by removing w from it (denoted by s_{-w}) and examine the output differences,

$$\sigma_f(w, s) = P(y_p|s) - P(y_p|s_{-w}).$$

The larger $\sigma_f(w, s)$ is, the more $P(y_p|s_{-w})$ changes, and the more important w for getting the right sentiment label. We will use $\sigma_f(w, s)$ as an indicator of target-aware opinion words, and aggregate them on $\mathcal{D}$. Let $S_t^w \subseteq S_t$ be the set of sentences which t and w co-occur, we average $\sigma_f(w, s)$ on S_t^w to get the model-free importance score $\sigma_f(w)$,

$$\sigma_f(w) = \log P(w|t)\frac{1}{|S_t^w|}\sum_{s\in S_t^w}\sigma_f(w, s). \quad (3)$$

In order to reduce the affect of noise and rare language usage, we take co-occurrence statistic into account: a target-aware opinion word should co-occur with the target often. Therefore, the average score is empirically scaled with their co-occurrence probability $P(w|t) = \frac{|S_t^w|}{|D|}$.

The score $\sigma_f(w)$ is model-free in the sense that we don't need to know details of the sentiment classifier and only inquire the difference of outputs when the input sentence is perturbed. Hence,

though we use the BERT-based classifier here, we can use any other off-the-shelf sentiment classifiers (e.g., pre-trained models with different training objectives, multi-task learned classifiers, etc.) to further enrich (or constrain) the score.

4.2 Continuous Perturbation

Besides the discrete perturbation setting, we could also utilize the full classification model to identify target-aware opinion words. In this continuous perturbation setting, we ask the same question of how the sentiment prediction will change when we perturb sentence words. However, instead of perturbing them discretely (i.e., removing a word), we can perform continuous perturbations on word vectors (Goodfellow et al., 2015).

Let $L(y_p, s, \mathbf{w}) = -\log P(y_p|s)$ be the loss on sentence s and $\mathbf{w}$ is the word vector of w. If we slightly perturb $\mathbf{w}$ to $\mathbf{w}'$ with $\|\mathbf{w}' - \mathbf{w}\| \leq \varepsilon$, we can bound the absolute change of the loss function using the first-order approximation of $L(y_p, s, \mathbf{w})$,

$$\begin{aligned}
&|L(y_p, s, \mathbf{w}') - L(y_p, s, \mathbf{w})| \\
&\approx |\nabla_{\mathbf{w}} L(y_p, s, \mathbf{w})^T (\mathbf{w}' - \mathbf{w})| \\
&\leq \|\nabla_{\mathbf{w}} L(y_p, s, \mathbf{w})\| \|\mathbf{w}' - \mathbf{w}\| \\
&\leq \varepsilon \|\nabla_{\mathbf{w}} L(y_p, s, \mathbf{w})\|.
\end{aligned}$$

The magnitude of the gradient's norm $\|\nabla_{\mathbf{w}} L(y_p, s, \mathbf{w})\|$ could be a sign of how sensitive the sentiment label is with respect to w: to get the right prediction we will prefer not to perturb those words with large gradient norms. Therefore, a large gradient norm may also indicate an opinion words of the target. Define

$$\sigma_b(w, s) = \frac{1}{g^* - g_*} \left(\|\nabla_{\mathbf{w}} L(y_p, s, \mathbf{w})\| - g_* \right),$$

where $g^* = \max_{s \in S_t^w} \|\nabla_{\mathbf{w}} L(y_p, s, \mathbf{w})\|$, $g_* = \min_{s \in S_t^w} \|\nabla_{\mathbf{w}} L(y_p, s, \mathbf{w})\|$ are the maximum and minimum gradient norm in S_t^w, which help normalizing $\sigma_b(w, s)$ into $[0, 1]$.

Similar to Equation 3, we collect all $\sigma_b(w, s)$ in S_t^w and scale their average with co-occurrence probability. The model-based score of w is defined as,

$$\sigma_b(w) = \log P(w|t) \frac{1}{|S_t^w|} \sum_{s \in S_t^w} \sigma_b(w, s). \quad (4)$$

Finally, the computation of both discrete perturbation and continuous perturbation could be done efficiently using auto-gradient tools. The discrete perturbation setting requires a forward process of the network, while the continuous perturbation setting needs an additional backward computation. We also note that the "perturb-and-see" strategy behind both scores characterizes the relation between opinion words and the target only through the sentiment label, which is an indirect way. As a consequence, though the scores could recognize "big" implies a negative opinion on "battery", it could also identify "not" in "the battery is not big" as an important word for the positive opinion. In practice, we could filter out such cases by rules, but how to explicitly handle semantic composition in importance scores would be an important future work.

5 Polarity Inference

Given the importance scores of words with respect to t, we can rank them accordingly and take the top-k words as t's opinion lexicon. As the final step, we are left to determine the polarity of an opinion word o. We accomplish this by building template sentences which try to carry the semantic like "what opinion on which target". We call these sentences *template* which will be use to probe the sentiment classifier's knowledge on (t, o)'s polarity.

Formally, define $\mathcal{T}$ to be a set of templates, each template $\tau \in \mathcal{T}$ takes an opinion word and a target as input, outputs a natural language sentence $\tau(t, o)$. Here, we use the following two templates,

- $\tau(t, o) =$ "The t is o." (e.g., $\tau(\text{battery}, \text{big}) =$ "The battery is big.").
- $\tau(t, o) =$ "o t." (e.g., $\tau(\text{battery}, \text{big}) =$ "big battery.").

By feeding $\tau(t, o)$ into the sentiment classifier, we obtain $P(y_p|\tau(t, o))$, and the polarity distribution $\mathbf{p}$ of (t, o) is averaged over all templates,

$$\mathbf{p} = \frac{1}{|\mathcal{T}|} \sum_{t \in \mathcal{T}} P(y|\tau(t, o)) \quad (5)$$

6 Experimental Results and Analyses

We wish to evaluate the merit of our target-aware sentiment lexicon in this section. We first introduce the experimental setup in Section 6.1. Then, we design detail experiments to answer the following key questions.

	Electronic						Restaurant					
	P@5		P@10		P@20		P@5		P@10		P@20	
	$\mathcal{L}$	$\mathcal{L}_c$	$\mathcal{L}$	$\mathcal{L}_c$	$\mathcal{L}$	$\mathcal{L}_c$	$\mathcal{L}$	$\mathcal{L}_c$	$\mathcal{L}$	$\mathcal{L}_c$	$\mathcal{L}$	$\mathcal{L}_c$
PMI	0.292	0.176	0.284	0.190	0.287	0.178	0.260	0.132	0.260	0.138	0.273	0.149
DP	0.556	**0.352**	0.514	**0.296**	0.460	0.244	0.952	**0.772**	0.932	0.602	0.899	**0.597**
CP	0.608	0.212	0.620	0.210	0.596	0.206	0.828	0.676	0.814	0.536	0.816	0.429
DP+CP	**0.704**	0.296	**0.686**	0.288	**0.628**	**0.262**	**0.980**	0.748	**0.960**	**0.674**	**0.927**	0.537

Table 1: The results of human evaluation on $\mathcal{L}$ and $\mathcal{L}_c$ over electronic and restaurant. DP and CP mean discrete perturbation and continuous perturbation, respectively.

Q1 Can we trust our target-aware sentiment lexicon? To evaluate the quality of the extracted lexicon, we test the performance with both manual evaluation (Section 6.2) and automatic downstream task (Section 6.3).

Q2 Useful or not? As an application, we apply our lexicon into unsupervised opinion extraction task in Section 6.4.

Q3 Do we really understand our model? In Section 6.5, to investigate the insight of commonsense sentiment mined from the texts, we visualize several real-world examples.

6.1 Experimental Setup

We conduct experiments to validate the effectiveness of our approach on two widely different domains: electronic and restaurant, taken from Amazon dataset[3] and Yelp Challenge 2015[4]. We obtain the target set from SemEval'14, SemEval'15, and SemEval'16 for convenience [5].

The extracted target-aware sentiment lexicon ($\mathcal{L}$) can be divided into target-aware general sentiment lexicon ($\mathcal{L}_g$) and commonsense sentiment lexicon ($\mathcal{L}_c$). $\mathcal{L}_g$ means the opinion words in $\mathcal{L}$ that are in general lexicon and $\mathcal{L}_c$ means the opinion words in $\mathcal{L}$ that are not in general lexicon. Here, we use the general lexicon from (Hu and Liu, 2004) to filter the general sentiment words and obtain the commonsense lexicon. This general lexicon contains around 6800 positive and negative opinion words or sentiment words for the English language.

We adopt $\text{BERT}_{\text{base}}$ as the basis for all experiments. Adam (Kingma and Ba, 2015) is adopted as the optimizer with learning rate 5e-5 for fine-tuning and sentiment classification.

[3]http://jmcauley.ucsd.edu/data/amazon/

[4]https://www.yelp.com/dataset/challenge

[5]Here we use the targets from existing datasets, but the targets could be extracted automatically through existing work (Poria et al., 2014) or be inputted by users.

6.2 Human Evaluation

To evaluate the quality of the target-aware sentiment lexicon, we test its performance through human evaluation. For quantitative evaluation, we sample 50 targets with top-20 opinion words in each domain to investigate the performance of $\mathcal{L}$ and $\mathcal{L}_c$. Finally, we obtain 3122 and 2877 (t, o) pairs after filtering repetitive pairs for electronic and restaurant, respectively. We ask ten annotators to label them to make sure each pair is marked with three times. Then, we obtain the label through voting. We calculate the Krippendorff's alpha coefficient (Krippendorff, 2011) to measure the inter-annotator agreement of the manual annotation. The value is 0.850 and 0.702 for restaurant and electronic, which indicates the high agreement of the labeled data.

Table 1 reports the results of the human evaluation. The pointwise mutual information (PMI) measure (Hamilton et al., 2016; Church and Hanks, 1990) is adopted as the baseline to compare with, which applied to each target t w.r.t. each word w. We adopt the precision of top-k (e.g., 5, 10, and 20) to measure the performance of the methods across both $\mathcal{L}$ and $\mathcal{L}_c$. From this table, we observe that: **First**, both our discrete perturbation and continuous perturbation algorithms perform much better than PMI. Additionally, in the restaurant domain, our model obtains more than 90% precision for $\mathcal{L}$. These indicate the great effectiveness of capturing target-aware sentiment words and commonsense sentiment words. **Second**, the discrete perturbation method often has higher precision than continuous perturbation method, but the combination of them (Discrete+Continuous Perturbation) [6] obtains the best results in most cases. It suggests that the discrete perturbation and the continuous perturbation settings may focus on different types of opinion words.

[6]Here we simply calculate the average score of them, but different weights can be used.

	Electronic	Restaurant
Original	93.82	91.64
Random-based Deleting	90.49	88.13
Lexicon-based Deleting	$84.77^{\dagger\ddagger}$	$80.67^{\dagger\ddagger}$

Table 2: The results of downstream task: sentiment analysis via **Strategy 1**. The marker † and ‡ refer to p-values < 0.05 when "original" and "random-based deleting" compare with "lexicon-based deleting".

6.3 Downstream Tasks

Besides human evaluation, we also automatically evaluate our commonsense sentiment lexicon $\mathcal{L}_c$ with downstream tasks. Here we examine document-level sentiment analysis. In particular, for each domain, we sample 3500 documents which do not contain any general sentiment lexicon words but have obvious opinion orientations on electronic and restaurant ("Original"). Then we perform sentiment classification on the dataset with $\mathcal{L}_c$ using two strategies.

Strategy 1 For each sample in "Original", we remove opinion words which appear in our $\mathcal{L}_c$, and test the performance of sentiment classification using a well-trained sentiment classifier (Section 3). Note that we only use the top-100 opinion words to make sure only fewer than five words are being deleted. To show the effectiveness of our lexicon, we compare our model with removing words randomly with the same rate (Table 2). We find that removing the words in $\mathcal{L}_c$ performs significantly worse than both the original and random removing. It indicates that our method can capture the commonsense opinion words effectively.

Strategy 2 We apply our commonsense lexicon as extra knowledge to enhance a sentiment classification model. Here, we study the standard BiLSTM-based classifier: a BiLSTM is used to encode sentences, the last hidden vector of a sentence is adopt for classification. To inject our extracted lexicon $(t, o, \mathbf{p})$, we concatenate $\mathbf{p}$ to the input of BiLSTM if t and o occur. We sample 1000 and 500 instances from previous 3500 samples as the training and test set. To validate the effectiveness of each model components, we also show ablation test results. Table 3 shows the results. We have the following observations.

• Our commonsense lexicon $\mathcal{L}_c$ can significantly improve the performance of sentiment classification. $\mathcal{L}_c$ + BiLSTM outperforms basic BiLSTM, while the model with PMI is even worse than BiLSTM. We also find the results of the discrete perturbation and continuous perturbation method are similar, and both of them can improve the results of sentiment classification.

	Electronic	Restaurant
BiLSTM	78.84	80.89
$\mathcal{L}_c$ + BiLSTM	$80.77^{\triangle}$	$81.71^{\triangle}$
$\mathcal{L}_c$ + BiLSTM - $\mathbf{p}$	79.56	81.11
$\mathcal{L}_c$ + BiLSTM (PMI)	78.75	80.63
$\mathcal{L}_c$ + BiLSTM (w/o DP)	80.11	81.45
$\mathcal{L}_c$ + BiLSTM (w/o CP)	80.01	81.50
$\mathcal{L}_c$ + BiLSTM (w/o DS)	79.57	80.94

Table 3: The results of sentiment analysis via **Strategy 2**. The marker $^{\triangle}$ refers to p-values < 0.05 comparing with "BiLSTM". DP and CP mean discrete perturbation and continuous perturbation, respectively. DS represents distant supervision.

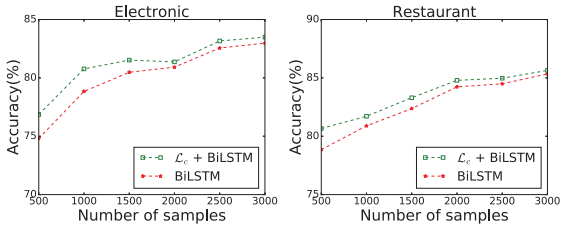


Figure 4: The results of sentiment analysis with different sample number.

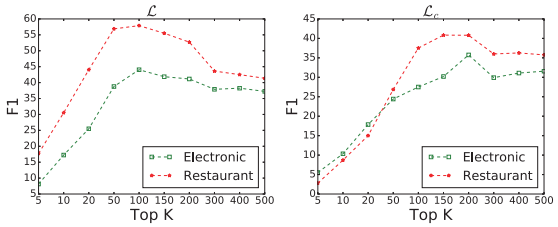
• $\mathcal{L}_c$ + BiLSTM performs better than the corresponding model without distant supervision, which indicates our distant supervision can capture the target information effectively. To further verify the effectiveness of distant supervision, we also randomly select 200 samples from the set S_t and evaluate them with three annotators by voting. The accuracy is 80.5% and 82% for 5-class classification over electronic and restaurant domains. Additionally, there are 71% and 65% of the samples have different polarities with their document-level label, and the accuracy of these samples is 80.99% and 81.54% in electronic and restaurant domains. These indicate our distant supervision can learn the target-oriented sentiment effectively.

• Compared with $\mathcal{L}_c$ + BiLSTM - $\mathbf{p}$ (which takes whether a word is an opinion word as feature), $\mathcal{L}_c$ + BiLSTM obtains better results. It suggest that polarity inference might be reasonable to infer the polarities of (t, o) pairs.

Additionally, to investigate the influence of sample numbers, we draw the results with different sample numbers in Figure 4. We can find that the

	Electronic			Restaurant		
	P	R	F1	P	R	F1
Rule-based	50.13	33.86	40.42	58.14	42.71	47.39
Ours	46.12	42.13	44.04	53.93	62.47	57.89
LSTM	55.71	57.53	56.52	57.46	64.96	60.87

Table 4: The results of opinion extraction. Note that LSTM is a supervised method.



(a) (b)

Figure 5: The results of unsupervised opinion extraction with different top-k via $\mathcal{L}$ and $\mathcal{L}_c$.

fewer samples, the more improvement by our commonsense lexicon.

6.4 Application (Unsupervised Opinion Extraction)

To answer **Q2**, we apply our lexicon into unsupervised opinion relation extraction. We test our lexicon on two datasets[7]: electronic and restaurant, which are released by (Fan et al., 2019), who labeled the opinion words towards the given target.

To investigate the performance of the target-aware sentiment lexicon $\mathcal{L}$, we perform unsupervised opinion extraction on the whole dataset. Table 4 reports the experimental results. We compare our method with two methods: 1) rule-based method (Hu and Liu, 2004) use the distance and POS tags to determine the opinion words; 2) supervised LSTM was proposed by (Liu et al., 2015). We use the results reported in (Fan et al., 2019) here. From this table, we observe: **First**, our $\mathcal{L}$ performs significantly better than the rule-based method even without using any rules or human annotations. **Second**, our unsupervised method is comparable with the supervised method (e.g., LSTM) in Restaurant. Additionally, we explore the influence of top-k in Figure 5 (a). We can find that top-100 is recommended for $\mathcal{L}$ in our experiments.

To verify our method can extract commonsense opinion words accurately, we also evaluate our $\mathcal{L}_c$ on the samples without general words. From Figure 5 (b), we can find that $\mathcal{L}_c$ achieves 40% F1 on the restaurant domain. Considering that we don't include any general sentiment words, we think the result is quite promising.

[7]https://github.com/NJUNLP/TOWE

6.5 Case Studies

To investigate the insight of commonsense sentiment mined from texts, we show several real-world examples in electronic and restaurant in this section. We present some interesting discoveries through in-depth analysis as follows.

We explore the sentiment polarity of different targets with the same opinion word here. As shown in Figure 1, we draw the targets w.r.t. opinion words "hot" and "long". We obtain the following interesting findings. **First**, our model can detect the commonsense sentiment in the corpus effectively. For example, our model can find that "hot" is a common-used collocation for "pizza", "CPU", and "battery", and it expresses a positive sentiment for "pizza", while it represents a negative sentiment for "CPU" and "battery". **Second**, domain-dependent sentiment words and their orientations are insufficient, and both the target and the opinion words are essential. For example, "long" has a positive polarity for "battery life" and negative sentiment for "charge" even both "battery life" and "charge" are in the electronic domain.

The opinion words most related to the given target (top-10) in $\mathcal{L}$ and $\mathcal{L}_c$ are shown in Table 5. From this table, we obtain the following discoveries. **First**, our method captures not only the general opinion words but also the commonsense opinion words. **Second**, as mentioned in Section 4.2, though the scores could recognize "fast" expresses a positive opinion on "response", it also identifies the words are important for sentiment but not opinion words, such as "no", "not" and "never". In practice, we could filter out such cases by rules, but how to explicitly handle semantic composition in importance scores would be an important future work.

From Table 5, we observe that $\mathcal{L}$ and $\mathcal{L}_c$ for different targets are quite different. To investigate whether the common-used opinion words for different targets are different, we measure it by,

$$\text{div} = \frac{1}{|T|(|T|-1)} \sum_{\substack{i,j=0 \\ i \neq j}}^{|T|} 1 - \frac{|\mathcal{L}^{t_i} \cap \mathcal{L}^{t_j}|}{|\mathcal{L}^{t_i} \cup \mathcal{L}^{t_j}|} \quad (6)$$

where T is the set of targets in our dataset, t_k is the k-th target in T and $\mathcal{L}^{t_k}$ means the sentiment lexicon of t_k. The value of div is 0.65 and 0.90 (0.89

	Electronic				Restaurant			
	Response		Memory		Workers		Service	
	$\mathcal{L}$	$\mathcal{L}_c$	$\mathcal{L}$	$\mathcal{L}_c$	$\mathcal{L}$	$\mathcal{L}_c$	$\mathcal{L}$	$\mathcal{L}_c$
1	fast$_+$	fast$_+$	hesitate$_*$	hesitate$_*$	rude$_*$	attitude	worst$_*$	never
2	quickly$_+$	quickly$_+$	perfectly$_+$	recent$_+$	attitude	extremely$_+$	horrible$_*$	zero$_*$
3	excellent$_+$	quite	recent$_+$	class$_+$	terrible$_*$	not	terrible$_*$	beat$_+$
4	quite	longer$_*$	class$_+$	crucial$_+$	extremely$_+$	greeted$_+$	amazing$_+$	extremely$_+$
5	longer$_*$	faster$_+$	crucial$_+$	suggest$_+$	super$_+$	professional$_+$	disappointed$_*$	5$_+$
6	faster$_+$	sometimes$_*$	suggest$_+$	frame	friendly$_+$	ignored$_*$	outstanding$_+$	above
7	sometimes$_*$	no	proprietary$_+$	fastest$_+$	helpful$_+$	fast$_+$	awful$_*$	average
8	no	appropriately$_+$	prefer$_+$	faster$_+$	nice$_+$	dressed$_+$	excellent$_+$	professional$_+$
9	appropriately$_+$	remote	limited$_*$	swap	efficient$_+$	welcoming$_+$	exceptional$_+$	five$_+$
10	softer$_+$	plugged$_*$	perfect$_+$	kingston	incompetent$_*$	friendliest$_+$	sucks$_*$	fast$_+$

Table 5: We list top-10 opinion words of several targets for two domains: electronic and restaurant. The marker $_+$ and $_*$ represent positive and negative sentiment respectively.

and 0.96) for $\mathcal{L}$ and $\mathcal{L}_c$ over restaurant (electronic). All these indicate that commonsense lexicon $\mathcal{L}_c$ is more diverse than general lexicon $\mathcal{L}_g$ over different targets. In addition, the commonly used general opinion words and commonsense sentiment words are different for different targets.

7 Related Work

Domain adaptation has been studied for a long time in the field of sentiment analysis (Wu et al., 2017; Choi and Cardie, 2009; Cambria et al., 2018; Zhou et al., 2020c). We mainly summarize the related work about lexicon domain adaptation that aims to build a domain-specific sentiment lexicon (Ofek et al., 2016; Vo and Zhang, 2016; Hamilton et al., 2016). In (Hamilton et al., 2016), authors inferred the orientation of words from general opinion words by building a graph for each domain. Xing et al. (2019) judged the word polarity via a document-level sentiment classifier. However, it is time-consuming for they have to retrain the model for each word after changing the polarity randomly. Moreover, these existing methods mainly focus on the domain-level, while the sentiment polarities of some words depend on their opinion targets (Liu and Zhang, 2012). It is essential to predict the sentiment in target-level by integrating both target and opinion words.

The most related work to us is (Zhao et al., 2012). Zhao et al. (2012) focused on inferring the polarity of a binary tuple of a polarity word and a target via search engine, while target-aware opinion words extraction is not fully explored. To take the target into account, Wu et al. (2019) proposed to construct a target-specific sentiment lexicon. However, both NLP preprocessing pipelines (e.g., parsing, POS tagging) and linguistic rules are integrated into their algorithm. Different from them, we first extract the target-aware commonsense opinion words via pre-trained models, which learned rich commonsense knowledge hidden in human languages. Then, we predict the sentiment polarity of target and opinion word pair through a probing strategy. We focus on building context-aware lexicons by minimizing the requirement of annotations and handcrafted external resources.

To take the target into account, Wu et al. (2019) proposed to construct a target-specific sentiment lexicon. However, both NLP preprocessing pipelines (e.g., parsing, POS tagging) and linguistic rules are integrated into their algorithm. Available resources like general sentiment lexicon and thesaurus are also made used. Since it is not easy to apply on different domains, we develop a framework to automatically mine aspect-aware commonsense sentiment from texts without extensive annotations and elaborated linguistic rules.

Pre-trained models (e.g., ELMo (Peters et al., 2018), GPT (Radford et al., 2019), BERT (Devlin et al., 2019)) have achieved great success in NLP recently. By exploring a large number of open domain texts, pre-trained models are able to encode rich semantic information hidden in human languages and thus provide new powerful tools for knowledge mining and extraction (Davison et al., 2019; Petroni et al., 2019). Since the commonsense opinions are closely related to human commonsense and background knowledge, we adopt pre-trained language models to mine the commonsense sentiment from texts automatically.

Gradient-based methods (Goodfellow et al., 2015) have been widely applied into computer version and NLP (Zeiler and Fergus, 2014; Liang et al., 2018). The gradient-based approach is also used to understand the decisions of the text classification models from the token level (Li et al., 2016; Alikaniotis et al., 2016). In addition, Rei et al. (2018)

adopted gradient-based approach to detect the important tokens in the sentence via the sentence-level label. In this paper, we design a continuous perturbation algorithm to discover the target-aware opinion words using the gradient.

8 Conclusion

In this paper, we propose a framework for automatic target-aware sentiment mining from texts without manual annotations or linguistic rules. We evaluate the proposed framework on two large-scale online review domains: restaurant and electronic with both manual checking and automatic downstream tasks. We also achieve significant improvements by applying the opinion lexicon to the task of unsupervised opinion relation extraction. To investigate the insight of commonsense sentiment mined from the texts, we visualize several real-world examples and analyze them in-depth. The extensive experimental results demonstrate the excellent performance in building a target-aware sentiment lexicon.

Acknowledgements

The authors wish to thank the reviewers for their helpful comments and suggestions. This research is (partially) supported by NSFC (62076097), STCSM (18ZR1411500). This work is also supported by National Key R&D Program of China (No.2018AAA0100503&2018AAA0100500). The computation is performed in ECNU Multifunctional Platform for Innovation(001). The corresponding authors are Yuanbin Wu and Liang He.

References

Dimitrios Alikaniotis, Helen Yannakoudakis, and Marek Rei. 2016. Automatic text scoring using neural networks. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*. The Association for Computer Linguistics.

Erik Cambria, Soujanya Poria, Devamanyu Hazarika, and Kenneth Kwok. 2018. Senticnet 5: Discovering conceptual primitives for sentiment analysis by means of context embeddings. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 1795–1802. AAAI Press.

Yejin Choi and Claire Cardie. 2009. Adapting a polarity lexicon using integer linear programming for domain-specific sentiment classification. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing, EMNLP 2009, 6-7 August 2009, Singapore, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 590–598. ACL.

Kenneth Ward Church and Patrick Hanks. 1990. Word association norms, mutual information, and lexicography. *Comput. Linguistics*, 16(1):22–29.

Joe Davison, Joshua Feldman, and Alexander M. Rush. 2019. Commonsense knowledge mining from pretrained models. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 1173–1178. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Zhifang Fan, Zhen Wu, Xin-Yu Dai, Shujian Huang, and Jiajun Chen. 2019. Target-oriented opinion words extraction with target-fused neural sequence labeling. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 2509–2518. Association for Computational Linguistics.

Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. 2015. Explaining and harnessing adversarial examples. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

William L. Hamilton, Kevin Clark, Jure Leskovec, and Dan Jurafsky. 2016. Inducing domain-specific sentiment lexicons from unlabeled corpora. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pages 595–605. The Association for Computational Linguistics.

Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Seattle, Washington, USA, August 22-25, 2004*, pages 168–177. ACM.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Klaus Krippendorff. 2011. Computing krippendorff's alpha-reliability.

Jiwei Li, Xinlei Chen, Eduard H. Hovy, and Dan Jurafsky. 2016. Visualizing and understanding neural models in NLP. In *NAACL HLT 2016, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego California, USA, June 12-17, 2016*, pages 681–691. The Association for Computational Linguistics.

Bin Liang, Hongcheng Li, Miaoqiang Su, Pan Bian, Xirong Li, and Wenchang Shi. 2018. Deep text classification can be fooled. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden*, pages 4208–4215. ijcai.org.

Bing Liu and Lei Zhang. 2012. A survey of opinion mining and sentiment analysis. In Charu C. Aggarwal and ChengXiang Zhai, editors, *Mining Text Data*, pages 415–463. Springer.

Pengfei Liu, Shafiq R. Joty, and Helen M. Meng. 2015. Fine-grained opinion mining with recurrent neural networks and word embeddings. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, pages 1433–1443. The Association for Computational Linguistics.

Mike Mintz, Steven Bills, Rion Snow, and Daniel Jurafsky. 2009. Distant supervision for relation extraction without labeled data. In *ACL 2009, Proceedings of the 47th Annual Meeting of the Association for Computational Linguistics and the 4th International Joint Conference on Natural Language Processing of the AFNLP, 2-7 August 2009, Singapore*, pages 1003–1011. The Association for Computer Linguistics.

Nir Ofek, Soujanya Poria, Lior Rokach, Erik Cambria, Amir Hussain, and Asaf Shabtai. 2016. Unsupervised commonsense knowledge enrichment for domain-specific sentiment analysis. *Cognitive Computation*, 8(3):467–477.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)*, pages 2227–2237. Association for Computational Linguistics.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick S. H. Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander H. Miller. 2019. Language models as knowledge bases? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 2463–2473. Association for Computational Linguistics.

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

Soujanya Poria, Erik Cambria, Lun-Wei Ku, Chen Gui, and Alexander F. Gelbukh. 2014. A rule-based approach to aspect extraction from product reviews. In *Proceedings of the Second Workshop on Natural Language Processing for Social Media, SocialNLP@COLING 2014, Dublin, Ireland, August 24, 2014*, pages 28–37. Association for Computational Linguistics and Dublin City University.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Marek Rei and Anders Søgaard. 2018. Zero-shot sequence labeling: Transferring knowledge from sentences to tokens. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)*, pages 293–302. Association for Computational Linguistics.

Changzhi Sun, Yuanbin Wu, Man Lan, Shiliang Sun, and Qi Zhang. 2017. Large-scale opinion relation extraction with distantly supervised neural network. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 1033–1043, Valencia, Spain. Association for Computational Linguistics.

Duy-Tin Vo and Yue Zhang. 2016. Don't count, predict! an automatic approach to learning sentiment lexicons for short text. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 2: Short Papers*, pages 219–224. The Association for Computer Linguistics.

Fangzhao Wu, Yongfeng Huang, and Jun Yan. 2017. Active sentiment domain adaptation. In *Proceedings of the 55th Annual Meeting of the Association*

for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers, pages 1701–1711. Association for Computational Linguistics.

Sixing Wu, Fangzhao Wu, Yue Chang, Chuhan Wu, and Yongfeng Huang. 2019. Automatic construction of target-specific sentiment lexicon. *Expert Systems with Applications*, 116:285–298.

Frank Z. Xing, Filippo Pallucchini, and Erik Cambria. 2019. Cognitive-inspired domain adaptation of sentiment lexicons. *Information Processing & Management*, 56(3):554–564.

Matthew D. Zeiler and Rob Fergus. 2014. Visualizing and understanding convolutional networks. In *Computer Vision - ECCV 2014 - 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part I*, volume 8689 of *Lecture Notes in Computer Science*, pages 818–833. Springer.

Yanyan Zhao, Bing Qin, and Ting Liu. 2012. Collocation polarity disambiguation using web-based pseudo contexts. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, EMNLP-CoNLL 2012, July 12-14, 2012, Jeju Island, Korea*, pages 160–170. ACL.

Jie Zhou, Qin Chen, Jimmy Xiangji Huang, Qinmin Vivian Hu, and Liang He. 2020a. Position-aware hierarchical transfer model for aspect-level sentiment classification. *Information Sciences*, 513:1–16.

Jie Zhou, Jimmy Xiangji Huang, Qinmin Vivian Hu, and Liang He. 2020b. SK-GCN: modeling syntax and knowledge via graph convolutional network for aspect-level sentiment classification. *Knowledge-Based Systems*, 205:106292.

Jie Zhou, Junfeng Tian, Rui Wang, Yuanbin Wu, Wenming Xiao, and Liang He. 2020c. Sentix: A sentiment-aware pre-trained model for cross-domain sentiment analysis. In *Proceedings of the 28th International Conference on Computational Linguistics, COLING 2020, Barcelona, Spain (Online), December 8-13, 2020*, pages 568–579. International Committee on Computational Linguistics.