CSenticNet: A Concept-Level Resource for Sentiment Analysis in Chinese Language

Haiyun Peng and Erik Cambria

School of Computer Science and Engineering
Nanyang Technological University
{PENG0065,cambria}@ntu.edu.sg

Abstract. In recent years, sentiment analysis has become a hot topic in natural language processing. Although sentiment analysis research in English is rather mature, Chinese sentiment analysis has just set sail, as the limited amount of sentiment resources in Chinese severely limits its development. In this paper, we present a method for the construction of a Chinese sentiment resource. We utilize both English sentiment resources and the Chinese knowledge base NTU Multi-lingual Corpus. In particular, we first propose a resource based on SentiWordNet and a second version based on SenticNet.

1 Introduction

The development of artificial intelligence (AI) has been rather rapid in recent years. As a major branch of AI, natural language processing (NLP) attracts much attention in both research and industrial fields [5]. One of the hottest topic in NLP is sentiment analysis, a 'suitcase' research problem that requires tackling many NLP sub-tasks, including aspect extraction [23], subjectivity detection [7], concept extraction [26], named entity recognition [17], and sarcasm detection [24], but also complementary tasks such as personality recognition [18] and user profiling [21]. However, research in the area of sentiment analysis can hardly progress much without a good pool of sentiment resources.

There are currently numerous English-language sentiment knowledge bases already in existence, such as SenticNet [4] and SentiWordNet [1]. When it comes to Chinese language, however, the numbers of similar resources are insufficient. Two major sentiment lexicons are currently available in Chinese: HowNet [9] and NTUSD [13]. However, both have their own drawbacks: HowNet only provides a positive or negative label for words. The labeling polarity does not give users information as to what extent a word expresses a sentiment. The entries in HowNet are basically simple words or idioms. As the fundamental elements (word level) in Chinese sentences and passages, their contribution to the overall sentiment is trivial compared with multi-word phrases. Furthermore, HowNet lacks semantic connections between its words. Their words are simply listed in pronunciation order, which makes it impossible to infer sentiment from semantics.

Although bigger than HowNet in size, NTUSD contains all the above drawbacks. To conclude, they are all word-level polarity lexicons. Because of these problems in the existing lexicons, this paper proposes a method to construct a concept-level sentiment resource in simplified Chinese to tackle the above issues, taking advantage of existing English sentiment resources and multi-lingual corpus.

The rest of the paper is organized as follows: Section 2 proposes the literature review of Chinese sentiment resources; Section 3 presents our framework for the construction of CSenticNet; Sections 4 and 5 explain in detail the first and second version of the Chinese-language sentiment resource, respectively; Section 6 presents evaluations of our methods; finally, Section 7 concludes the paper.

2 Literature Review

Two forms of sentiment resources are corpus and lexicon. A corpus is a collection of texts, especially if complete and self-contained: the corpus of Anglo-Saxon verse [19]. Due to the lack of large, expressively labeled Chinese language corpus, Chinese sentiment classification is very much hindered in its development. As such, some researchers decided to expand on or modify existing Chinese corpora. A relative fine-grained scheme was proposed by annotating emotion in text on three levels: document, paragraph and sentence [25]. Eight emotion classes (can be mapped to sentiment classes) were used to annotate the corpus and explore different emotion expressions in Chinese. Later, a Chinese Sentiment Treebank over social data was introduced [16]. 13550 sentences of movie reviews from social websites were crawled and manually labeled. Zhao et al. [30] created a fine-grained corpus with complex and manual annotation procedure.

Two issues exist in the above and current sentiment corpora. Firstly, they were manually built which is time and human-resource consuming. Secondly, they were annotated at sentence level. Sentiment corpus annotated at sentence-level is not enough. Because a corpus is usually utilized in a machine learning way. Words and phrases within a sentence play more important role in machine learning methodology compared with sentence itself. For instance in the negative sentiment sentence "I would prefer to read the novel after watching the movie" no negative words or phrases appeared. However, the words and phrases will wrongly be given a negative label due to sentence level annotation.

Another form of sentiment resource is sentiment lexicon. There are basically three types of sentiment lexicons in all [29]: 1) The ones only containing sentiment words, such as The Never-Ending Language Learner (NELL) [6]; 2) The ones containing both sentiment words and sentiment polarities (sentiment orientation), such as National Taiwan University Sentiment Dictionary (NTUSD) [13] and HowNet [9]; 3) The ones containing words and relevant sentiment polarity values (sentiment orientation and degree), such as SentiWordNet [1] and SenticNet [4]. In the first type, the lexicon only contains words for certain sentiments. It can help distinguish texts with sentiments from those that without. However, it is not able to tell whether the texts have positive or negative sentiments.

Furthermore, it is an English language corpus and not Chinese sentiment-related. The second, HowNet [9], is an on-line common-sense knowledge base which represents concepts in a connected graph. In terms of its sentiment resources, it has two lists which sentiment words are classified under: positive and negative. The problem this poses is a three-fold one. Firstly, it lacks semantic relationship among the words, as words are listed in alphabetical order. Secondly, it lacks multi-word phrases. Thirdly, it cannot distinguish the extent of the sentiment expressed by the words. For example, *uneasy* and *indignant* are both negative-connotation words but to different extents. HowNet classified these two words as equals in the 'negative' list with no discrepancy between them. NTUSD also has the above disadvantages.

With regards to the third type, both SentiWordNet and SenticNet provide polarity values for each entry in the lexicon. They are currently the most state-of-the-art sentiment resources available. However, their drawback is that they are only available in the English language, and hence do not support Chinese language sentiment analysis. Thus, some researchers seek to build sentiment resources via multi-lingual approach. Mihalcea et al. [20] tried projections between languages, but they have the problem of sense ambiguity during translation and time consuming annotation.

Hence, we propose a method that utilizes multi-lingual resources to construct a Chinese sentiment resource (third type above) which does not need manual labeling and solves the sense ambiguity issue. Its concepts are in connected graph and have both sentiment polarity and sentiment extent. Unlike existing cross-lingual approach [11, 8, 12, 14], there is no machine translation or mapping function learning step in the method. It discovers latent connection between two resources to map the English entity to Chinese in a dedicated way.

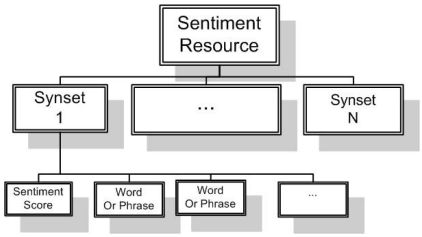
3 Framework

In this section, we introduce our proposal in general by listing the resources we are using and discussing the main steps we are taking. Our goal is to construct a Chinese sentiment resource, termed CSenticNet.

The CSenticNet should contain firstly sentiment words or phrases in simplified Chinese. The words and phrases should be organized in the form of synsets: a set of one or more synonyms. Under each synset node, we have words or phrases contributing to a similar meaning and a sentiment polarity value (between -1 and +1) they share. Figure 1a illustrates the data structure of CSenticNet.

3.1 Resources

By constructing the sentiment resource, we take advantage of existing resources available on the Internet within copyright/ethical guidelines. We present the different resources utilized in our resource below: *SenticNet* [4], *Princeton WordNet* [10], *NTU multi-lingual corpus* [27] and *SentiWordNet* [1].



(a) Data structure of CSenticNet

(影响,鼓动,感动,0.279) (touching, motivate)
(珍视,珍爱,珍重,0.859) (value, treasure)
(鄙弃,藐视,蔑视,-0.798) (contempt, disdain)
(神,偶像,神像,0.068) (idol, god)
(佩服,钦佩,0.781) (admire, esteem)
(镇压,禁止,抑制,-0.115) (suppress, prohibit)
(记住,学习,学会,0.777) (learnt, memorization)
(病症,疾病,病,症,-0.13) (disease, illness)

(b) First version results

Fig. 1: Data structure and examples of CSenticNet

SenticNet [4] is an English resource for concept-level sentiment analysis. It consists of 17k concept entries. Five affiliated semantic nodes are listed following each concept. These nodes are connected by semantic relations as illustrated in Fig. 2. There are also four sentics and a sentiment polarity value. The four sentics are a detailed emotional description of the concept they belong to (Fig. 4). The sentiment polarity value is an integrated evaluation of the concept sentiment based on the four parameters. Figure 3b gives an illustration of one such concept. *Princeton WordNet* [10] is a large lexical database of English. It contains four part-of-speech (POS) categories: Nouns, Verbs, Adjectives and Adverbs. Each category is a set of synsets. It totals 117k synsets, which are connected with each other by *conceptual relations*. It is the most popular English resource for its comprehensiveness and friendly access.

NTU MC (*NTU multi-lingual corpus*[1]) [27] translates Princeton *WordNet* into as many different languages as possible. NTU MC is a multilingual corpus that was built by Nanyang Technology University, and it contains 375,000 words (15,000 sentences) in 6 languages (English, Chinese, Japanese, Korean, Indonesian and Vietnamese) [27]. It has 42k Chinese concepts in the corpus and are linked by corresponding English translations in WordNet. Most importantly, concepts that are similar in English and Chinese were manually aligned, and such an approach makes NTU MC as the ideal referent for the mutual mapping of the concepts. Moreover, it is no longer merely a lexicon resource, because the translations comprise human semantic translation, like multi-word expressions and phrases. *SentiWordNet* is a lexical resource. It has one-to-one relations with *WordNet*, because it assigns each synset in *WordNet* with a positive score, a negative score and an objective score. The positive score represents the extent to which the word expresses a positive emotion, and vice versa for the negative score. With the above resources, we illustrate basic steps to show how to construct the Chinese sentiment resource.

[1] http://compling.hss.ntu.edu.sg/ntumc/

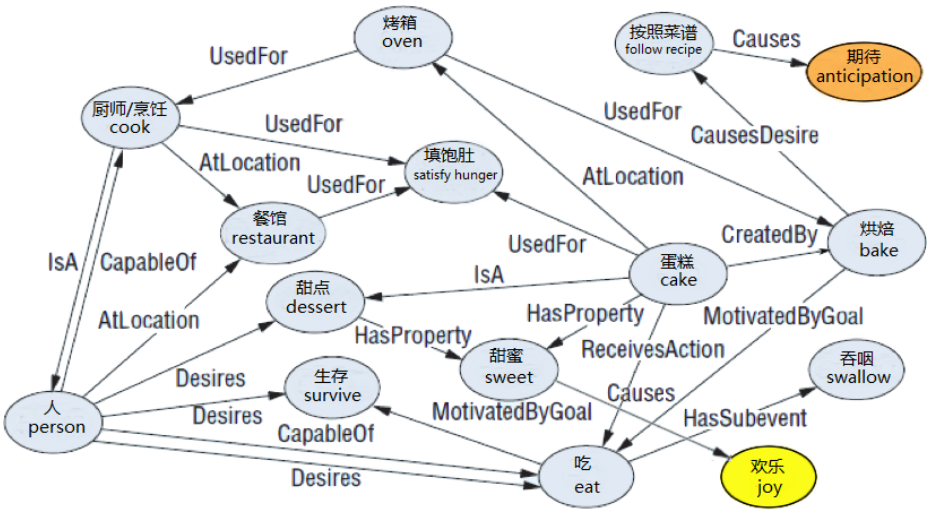


Fig. 2: CSenticNet graph

3.2 Two Versions

Among all the resources we are using, only NTU MC is in Chinese language. Therefore, it serves as the kernel of our resource. However, it does not have any information on sentiment, so our idea is to add affective information to this corpus to make it a sentiment resource.

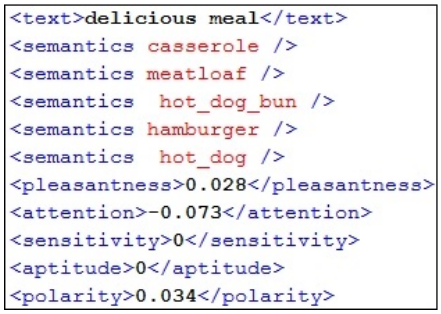
As for sentiment resources, we have SentiWordNet and SenticNet. Since these are independent of each other, we can use either of them to construct the sentiment resource. As such, we used SentiWordNet in the first version and then SenticNet in the second version.

In the first version, we map the sentiment information from SentiWordNet to NTU MC. Because SentiWordNet has corresponding sentiment polarity to each sense of WordNet, and NTU MC is manually translated from WordNet, we extract sentiment polarity from SentiWordNet and give to their Chinese translations in NTU MC via WordNet.

In the second version, we map the sentiment information from SenticNet to NTU MC. We first try to match all the single and multi-word concepts from SenticNet to WordNet. This is called direct mapping. We also proposed an enhanced version, which combines POS analysis and extended Lesk algorithm to deal with concepts and semantics that were not matched in the direct mapping. The increased number of matches is added to those derived through direct mapping. Finally, we find the overlap between the matched items and NTU MC.

In the following sections, we introduce these two versions in detail and present our evaluations.

(a) NTU MC data



(b) SenticNet data

Fig. 3: Example of used sentiment resources

4 First Version: SentiWordNet + NTU MC

As we explained previously, the role of the first version is to map the sentiment information from SentiWordNet to NTU MC. Because WordNet serves as the bridge that links SentiWordNet to NTU MC, we start by mapping both NTU MC and SentiWordNet to WordNet individually.

We begin by studying the structure of NTU MC. The knowledge base was organized in a lexical structure. The root hierarchy is 'LexicalResource'. Under the root node, there are two children branches: 'Lexicon' and 'SenseAxes'. 'Lexicon' is the mother of 61,536 'LexicalEntry(ies)'. Each 'LexicalEntry' has a Chinese word, its POS, its Sense ID and synset. Because some Chinese words can have different meanings in English, these 'LexicalEntries' sometimes have more than one pair of Sense ID and synset. Figure 3a below gives an example. The key clue that links NTU MC to WordNet is the synset ID. *synset=cmn-10-02208409-v* is a synset. The combination of *-02208409* and *-v* uniquely distinguish each synset(sense) in the NTU MC and in WordNet. Naturally, we re-organize the structure of this knowledge base by grouping all the words by synsets with unique synset ID. After processing, we have obtained 42,312 synsets and each synset has at least one Chinese word. The data was stored in a python dictionary.

Then we move to SentiWordNet. We firstly combine *POS* and *ID* of each synset and write them into the same format like NTU MC. Then we compute the sentiment polarity value of each synset. As each synset has a positive score and a negative score, we subtract the absolute value of negative score from positive score and treat the result as the sentiment polarity score. The range of final score is between -1 and +1, where polarity stands for sentiment orientation and absolute value means sentiment degree.

In some cases, the calculation results can be 0. This is due to either the synset having neither positive nor negative sentiment or the synset having equal positive and negative scores. We eliminate these synsets since they express no sentiment. Even though this reduces the size the resulting resource, the elimination of these synsets prevents introducing false information. However, the second reason may

be a future topic to study. The final version is in a text file format. Each line of the file has a synset (omitted in Figure) with its sentiment polarity score and the relevant Chinese words. Figure 1b shows some examples of the results.

5 Second Version: SenticNet + NTU MC

In the second version, we map the sentiment information from SenticNet to NTU MC. Because NTU MC is directly correlated with WordNet and WordNet is much bigger than SenticNet, it is better to map SenticNet to NTU MC rather than doing it the other way around. Thus, the complete mapping contains these three steps: map NTU MC to WordNet, map SenticNet to WordNet, then find and extract the overlap between SenticNet's and NTU MC's mappings in WordNet. As the first step of mapping NTU MC to WordNet was already finished in the first version, we directly inherit from there. The last step of finding and extracting the overlap is relatively straightforward and does not need much emphasis. Thus, in this second version, we mainly focus on the second step of how to map SenticNet to WordNet. Before that, we present an analysis of SenticNet below.

5.1 SenticNet and Preprocessing

As we can see from the Figure 3b, the sentiment value of the multi-word concept is *0.034*, which is a positive sentiment. The five semantics *casserole*, *meatloaf*, *hot_dog_bun*, *hamburger* and *hot_dog* all contribute to the concept of a delicious meal. We consider each of the semantics alone as sharing a similar sentiment value with the concept it describes, but we give each concept a higher priority than its semantics. From SenticNet, we have extracted about 17,000 concepts. Before mapping, we need to preprocess SenticNet. We extract every concept, its five semantics and its sentiment score and then put them in a python dictionary. The key of the dictionary is the concept, and the value is the corresponding semantics and sentiment score.

5.2 Mapping SenticNet to WordNet

After the preprocessing is done, we start step 2: mapping SenticNet to WordNet. Due to the diversity of SenticNet (single word, multi-word phrase, semantics), we have proposed two solutions to the problem: direct mapping and enhanced mapping. Direct mapping tries to map SenticNet to WordNet by word-to-word matching. Enhanced mapping integrate direct mapping with keyword extraction based on POS and extended Lesk algorithm.

Direct Mapping Since we have covered both SenticNet and WordNet in the python dictionary, we can conduct mapping directly. With WordNet, we have obtained a python dictionary which key is the word or phrase in WordNet and value is a list of synset ID.

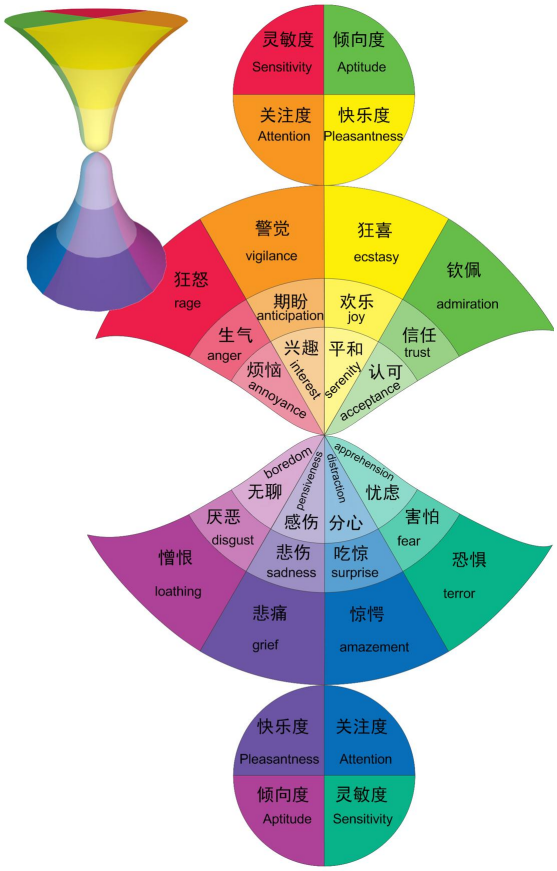


Fig. 4: Hourglass model for Chinese language

For WordNet, a key-value pair may look like this (concept followed by synset IDs): {abandoned : [cmn-10-01313004-a, cmn-10-01317231-a], ...}. For SenticNet, the key is concept and the value is its semantics, like {bank : [coffer, bank_vault, finance, government_agreement, money], ...}. We match each key in SenticNet dictionary to each key from the WordNet dictionary. If a key was matched, the hypernyms of each synset ID in the value from WordNet dictionary would be retrieved. Hypernyms are retrieved from WordNet itself. Synsets (hyponyms) are subordinates of their hypernyms. Then hypernyms of each synset ID will be matched with the words (both concept and semantics) in key-value pair from SenticNet.

If hypernyms from only one synset ID were matched, then this matched synset from WordNet shares the same meaning with the concept-semantics pair from SenticNet. Thus, the sentiment score of this concept from SenticNet will be given to this synset ID. If hypernyms from more than one synset ID were

matched, we compute how many words are matched with hypernyms for each synset ID and choose the synset that has most matched words as final matched synset, which will be given the sentiment score from SenticNet. The hypernym of synset ID is considered as layer 1. Hypernyms of the previous hypernyms are considered as layer 2 so on and so forth. If nothing was matched through the whole concept-semantics list in layer 1, we proceed to layer 2. If nothing was matched after layer 3, a concept is scraped. In the end, we accomplish mapping and obtain a dictionary whose key is the synset ID and value is the sentiment score.

The dictionary has 12,042 key-value pairs, which means we have mapped 12,042 synsets from SenticNet to WordNet, a size about one fourth of that of NTU MC. However, one issue that direct mapping failed to solve is the accuracy of matches. For example, referring to Figure 3b, we have a concept *delicious meal* and a sentiment score of 0.034. We can see that the sentiment score strongly represents the word *delicious* rather than *meal*. However, due to its non-exact match to WordNet, we lose the sentiment score of *delicious meal*, as well as the word *delicious*. In order to figure out the above-mentioned issue, we have developed an enhanced mapping method on top of direct mapping.

Enhanced Mapping with POS Analysis and Extended Lesk Algorithm As direct mapping has above problems, we develop POS analysis to tackle the exact match problem when concept was not matched, and combine extended Lesk algorithm to settle the problem of sense disambiguation when matching hypernyms failed. In this section, we first provide a review of the techniques we use and then introduce our methods. Before POS analysis, we tokenize the phrases first. This means breaking a string of short phrase into a string of tokens. Each token is a word from the phrase and this token can be read and analyzed by computer algorithms. Because we use python programming in our experiments, we apply the most popular third party tool *Natural Language Toolkit* to do the tokenization. After that is done, we annotate the tokens with POS tag. It helps to extract the key meaning in terms of sentiment and to distinguish the usage of a word in its different senses. We again take the example from Figure 3b. The concept *delicious meal* has a word *delicious* that is a POS adjective and a word *meal* which is a POS noun. The sentiment of this concept is expressed more by the adjective than the noun. By annotating the POS of each token, we have a better understanding of the sentiment of concept.

The Lesk algorithm is a word sense disambiguation algorithm developed by Michael Lesk in 1986 [15]. The algorithm is based on the idea that the sense of a word is in accordance with the common topic of its neighborhood. A practical example used in word sense disambiguation may look like this. Given an ambiguous word, each of its sense definition in the dictionary is fetched and compared with its neighborhood text. The number of common words that appear in both the sense definition and neighborhood text is recorded. At the end, the sense that has the biggest number of common words is the sense of this ambiguous word.

However, the ambiguous word may sometimes not have enough neighborhood text, so, people have developed ways to extend this algorithm. Timothy [2] explores different tokenization schemes and methods of definition extension. Inspired by their paper, we also developed a way of extension in our experiments. The extended algorithm can solve the ambiguous mapping problem in our direct mapping method.

In our experiments, all single words from SenticNet were easily matched to WordNet. The difficulty mainly falls in mapping multi-word phrases. We put a higher priority on the concepts in SenticNet and lower priority on its semantics. The reason is that sentiment scores in SenticNet are specifically computed for the concepts. Its semantics carry close-related meaning of the concept, so they share the same sentiment score. In a strict sense, this is not ideal.

Therefore, like direct mapping, we decide to match each concept in SenticNet to WordNet first. If it was not matched, we annotate the concept (if it is multi-word phrase) tokens with POS tags before sorting them by POS tag priority. The POS tag priority, from top to bottom, is: Verbs, Adjective, Adverb and Noun. This order of priority is based on the heuristics that top POS tags are more emotionally informative [22, 28]. The next step is to extend the contexts. We tokenize all five semantics of a concept and concatenate them with the concept token string to form a large token string. This string is considered as our extended context. At this point, we have prepared the necessary inputs for the Lesk algorithm.

The prioritized tokens with POS tags are considered as the ambiguous words while the large token string is the neighborhood text. We then treat the concept tokens one by one as ambiguous words, based on their POS priority, and apply these to the Lesk algorithm to compute the sense. Once the sense was matched to a sense in WordNet, the processing of this concept is finished and this sense and sentiment score is stored. If it was not matched after iterating through the concept tokens, then one of its semantics is POS tagged and the earlier listed procedures repeated. This process will not stop until a match is found in WordNet or all five semantics have been iterated. Figure 5 summarizes the framework of our two-version method.

In the end, we obtained a dictionary with 18,781 key-value pairs of synsets mapped from SenticNet to WordNet. This gave us 6,739 more pairs than the direct mapping method.

5.3 Find and Extract the Overlap

From the previous section, we obtained a python dictionary whose key-value pair is synset ID-sentiment score by mapping SenticNet to WordNet. In this section, we combine the dictionary with the NTU MC python dictionary we got in the first version and find their overlap. Altogether, 5,677 synsets were overlapping, which meant they had corresponding Chinese translations in NTU MC. Over 15,000 overlapped synsets with their sentiment score and Chinese translations were eventually written into a text file.

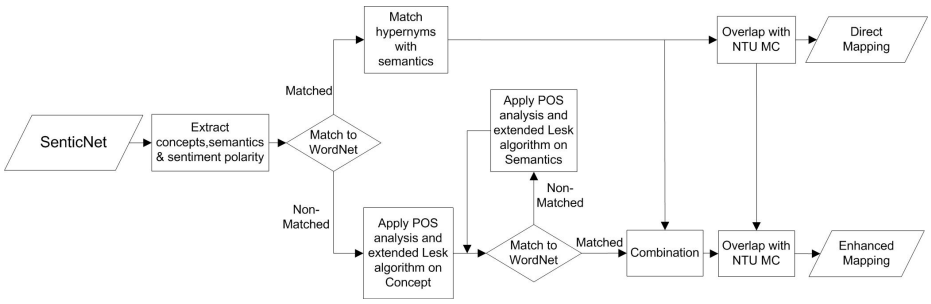


Fig. 5: Mapping Framework of SenticNet Version

Table 1: Accuracy of SentiWordNet and SenticNet version(column 2 to 7) and accuracy of small value sentimnet synsets(last 3 columns)

Annotator	**SentiWordNet version**			**SenticNet version**			**[-0.25, 0)**	**[0, 0.25]**	**Overall**
	Positive	**Negative**	**Overall**	**Positive**	**Negative**	**Overall**			
1	48%	64%	56%	82%	80%	**81%**	75%	81%	78%
2	50%	58%	54%	78%	76%	**77%**	75%	83%	79%
Kappa measure	0.96	0.79	-	0.88	0.88	-	0.73	0.70	-

6 Evaluation

In this section, we conduct three evaluations of our mapping. For manual validation, we asked two native Chinese speakers to each evaluate 200 entries in our final text files for the two versions of Chinese sentiment resource. Particularly for each of the two versions, 50 positive and 50 negative entries were randomly selected. Both experts were asked to label 200 entries from two versions as either positive or negative independently. We treat their manual labels as ground truth and compute the accuracies of our mapped sentiment resources. The results and inter-annotator agreement measures are in columns 2 to 7 of Table 1.

The results shown in the tables suggest that the SenticNet version outperforms the SentiWordNet version by almost 50 percent. This also validates our assumption that SenticNet is more reliable than SentiWordNet in terms of sentiment accuracy. As can be seen, the highest accuracy rate is over 80 percent. Moreover, there is still space to make improvements to this in the future.

In our mapping procedure, we assume synonyms and hypernyms share similar sentiment orientation with their root word. We believe this is true for the majority of words in the corpora. However, some words or expressions could have opposite sentiment orientation with their synonyms and hypernyms. As illustrated by the Hourglass model in [3], we know that words or expressions that have ambiguous sentiment orientation tend to have small absolute sentiment values. In order to validate our assumptions, we firstly inspect the sentiment value distribution of our SenticNet version sentiment resource and conduct manual validations.

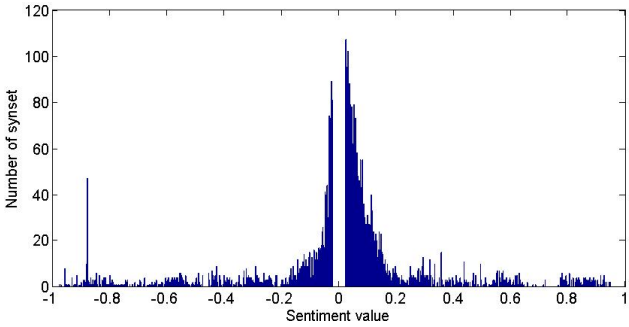


Fig. 6: Distribution of sentiment values

Table 2: Comparisons between CSenticNet and state-of-art sentiment lexicons

Sentiment resource	Chn2000			It168			Weibo		
	P	R	F1	P	R	F1	P	R	F1
NTUSD	50.08%	**99.18%**	66.55%	54.51%	**97.66%**	69.97%	51.17%	**99.39%**	67.56%
HowNet	53.29%	98.68%	69.21%	**61.07%**	96.79%	**74.89%**	50.76%	98.66%	67.03%
CSenticNet (SenticNet version)	**54.85%**	96.18%	**69.86%**	59.04%	94.19%	72.58%	**55.90%**	87.11%	**68.10%**

Figure 6 presents the distribution of all synsets based on their sentiment values. An empty interval exists in the sentiment axis around zero value. This suggests no synsets have very small absolute sentiment values. It partially proves our initial assumptions. However we also notice the high intensity of synsets with small values just beyond the empty interval. The sentiment of these synsets could be wrongly mapped due to our synonym and hypernym assumptions. Thus, we randomly picked up five subsets of synsets from sentiment value ranges (-0.25, 0] and (0, 0.25], respectively. Each subset contains 20 synsets. Then we asked the two native Chinese speakers to label sentiment orientation of the 200 chosen synsets and treat their labels as ground truth. Results are shown in last 3 columns of Table 1. Accuracies within the chosen intervals keep abreast with that of the whole axis. According to the second expert, the intervals even outperform the whole axis in sentiment orientation prediction. Furthermore, we also find that kappa measures of these intervals are less confident than that of the whole axis(columns 3 to 7 in Table 1). These results further support our initial assumptions and guaranteed the accuracy of our proposed sentiment resources.

Last but not least, shown in Table 2, we conduct sentiment analysis experiments to compare our CSenticNet (SenticNet version) with state-of-art baselines, HowNet and NTUSD. Three datasets we used are: Chn sentiment corpus 2000 (Chn2000[2]), It168[3] and Weibo dataset from NLP&CC[4]. The first dataset

[2] http://searchforum.org.cn/tansongbo/corpus/ChnSentiCorp_htl_ba_2000.rar

[3] http://product.it168.com

[4] NLP&CC is an annual conference of Chinese information technology professional committee organized by Chinese computer Federation (CCF). More details are available at http://tcci.ccf.org.cn/conference/2013/index.html

contains 1000 positive and 1000 negative reviews from hotel customers. We preprocess this dataset by manually selecting only one sentence which has clear sentiment orientation from each review. The second dataset contains 886 reviews of digital product downloaded and mannually labeled from a Chinese digital product website. The third dataset was micro-blogs originally used for opinion mining. We manually selected and labeled 1900 positive and negative sentences, respectively. We use a simple rule-based keyword matching classifier. Specifically for a test sentence, we match each of its words in sentiment lexicon and sum up the sentiment polarity of matched words in the sentence. For the baselines, positive words have +1 polarities and negative words have -1 polarities. If the final sum is above zero, then the sentence is positive and vice versa.

We see that CSenticNet outperforms the other two baselines in Chn2000 and Weibo datasets, at it has both higher precision and F1 score. However, it narrowly falls behind HowNet in the It168 dataset. We believe this is because of the highly domain biased dataset. It168 reviews are mostly in digital fields, but CSenticNet is not tuned for that domain. Thus, it was not supposed to defeat the other two baselines, but even thought it still performs better than NTUSD. We also find that the recall of CSenticNet is not high, and this gives us a chance to further enlarge the resource by using new versions of SenticNet in the future.

7 Conclusion

In this paper, we presented a method to construct the first concept-level Chinese sentiment resource. Instead of using machine translation, we mapped English sentiment resources to the Chinese corpus using a multilingual corpus. Special techniques were designed to solve issues such as ambiguity. We provide two versions of Chinese sentiment resource: one based on SentiWordNet, the other based on SenticNet. The SenticNet version outperforms state-of-the-art Chinese sentiment lexicons in our evaluations. Moreover, the proposed method can also be applied to other languages in NTU MC.

In the near future, we will focus on the unmatched cases and utilize other sources to enlarge the size of the proposed sentiment resource.

References

1. Baccianella, S., Esuli, A., Sebastiani, F.: Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In: LREC. vol. 10, pp. 2200–2204 (2010)
2. Baldwin, T., Kim, S., Bond, F., Fujita, S., Martinez, D., Tanaka, T.: A reexamination of mrd-based word sense disambiguation. ACM Transactions on Asian Language Information Processing (TALIP) 9(1), 4 (2010)
3. Cambria, E., Hussain, A.: Sentic Computing: A Common-Sense-Based Framework for Concept-Level Sentiment Analysis. Springer, Cham, Switzerland (2015)

4. Cambria, E., Poria, S., Bajpai, R., Schuller, B.: SenticNet 4: A semantic resource for sentiment analysis based on conceptual primitives. In: COLING. pp. 2666–2677 (2016)
5. Cambria, E., Wang, H., White, B.: Guest editorial: Big social data analysis. Knowledge-Based Systems 69, 1–2 (2014)
6. Carlson, A., Betteridge, J., Kisiel, B., Settles, B., Hruschka Jr, E.R., Mitchell, T.M.: Toward an architecture for never-ending language learning. In: AAAI. vol. 5, p. 3 (2010)
7. Chaturvedi, I., Cambria, E., Vilares, D.: Lyapunov filtering of objectivity for Spanish sentiment model. In: IJCNN. pp. 4474–4481. Vancouver (2016)
8. Chen, Q., Li, W., Lei, Y., Liu, X., He, Y.: Learning to adapt credible knowledge in cross-lingual sentiment analysis. In: ACL (2015)
9. Dong, Z., Dong, Q.: HowNet and the Computation of Meaning. World Scientific (2006)
10. Fellbaum, C.: WordNet: An Electronic Lexical Database. Bradford Books (1998)
11. Gui, L., Xu, R., Lu, Q., Xu, J., Xu, J., Liu, B., Wang, X.: Cross-lingual opinion analysis via negative transfer detection. In: ACL (2). pp. 860–865 (2014)
12. Jain, S., Batra, S.: Cross-lingual sentiment analysis using modified brae. In: EMNLP. pp. 159–168. Association for Computational Linguistics (2015)
13. Ku, L.W., Liang, Y.T., Chen, H.H.: Opinion extraction, summarization and tracking in news and blog corpora. In: AAAI spring symposium: Computational approaches to analyzing weblogs. vol. 100107 (2006)
14. Lambert, P.: Aspect-level cross-lingual sentiment classification with constrained smt. In: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Short Papers). pp. 781–787. Association for Computational Linguistics (2015)
15. Lesk, M.: Automatic sense disambiguation using machine readable dictionaries: how to tell a pine cone from an ice cream cone. In: Proceedings of the 5th annual international conference on Systems documentation. pp. 24–26. ACM (1986)
16. Li, C., Xu, B., Wu, G., He, S., Tian, G., Hao, H.: Recursive deep learning for sentiment analysis over social data. In: Proceedings of the 2014 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT)-Volume 02. pp. 180–185. IEEE Computer Society (2014)
17. Ma, Y., Cambria, E., Gao, S.: Label embedding for zero-shot fine-grained named entity typing. In: COLING. pp. 171–180. Osaka (2016)
18. Majumder, N., Poria, S., Gelbukh, A., Cambria, E.: Deep learning based document modeling for personality detection from text. IEEE Intelligent Systems 32(2), 74–79 (2017)
19. McArthur, T., McArthur, F.: The Oxford companion to the English language. Oxford Companions Series, Oxford University Press (1992)
20. Mihalcea, R., Banea, C., Wiebe, J.M.: Learning multilingual subjective language via cross-lingual projections (2007)
21. Mihalcea, R., Garimella, A.: What men say, what women hear: Finding gender-specific meaning shades. IEEE Intelligent Systems 31(4), 62–67 (2016)
22. Pavlenko, A.: Emotions and the body in russian and english. Pragmatics & Cognition 10(1), 207–241 (2002)
23. Poria, S., Cambria, E., Gelbukh, A.: Aspect extraction for opinion mining with a deep convolutional neural network. Knowledge-Based Systems 108, 42–49 (2016)
24. Poria, S., Cambria, E., Hazarika, D., Vij, P.: A deeper look into sarcastic tweets using deep convolutional neural networks. In: COLING. pp. 1601–1612 (2016)

25. Quan, C., Ren, F.: Construction of a blog emotion corpus for chinese emotional expression analysis. In: EMNLP. pp. 1446–1454. Association for Computational Linguistics (2009)
26. Rajagopal, D., Cambria, E., Olsher, D., Kwok, K.: A graph-based approach to commonsense concept extraction and semantic similarity detection. In: WWW. pp. 565–570. Rio De Janeiro (2013)
27. Tan, L., Bond, F.: Building and annotating the linguistically diverse ntu-mc (ntu-multilingual corpus). Int. J. of Asian Lang. Proc. 22(4), 161–174 (2012)
28. Wierzbicka, A.: Preface: Bilingual lives, bilingual experience. Journal of multilingual and multicultural development 25(2-3), 94–104 (2004)
29. Wu, H.H., Tsai, A.C.R., Tsai, R.T.H., Hsu, J.Y.j.: Building a graded chinese sentiment dictionary based on commonsense knowledge for sentiment analysis of song lyrics. J. Inf. Sci. Eng. 29(4), 647–662 (2013)
30. Zhao, Y., Qin, B., Liu, T.: Creating a fine-grained corpus for chinese sentiment analysis. Intelligent Systems, IEEE 30(1), 36–43 (2015)