Chapter 7
Sentiment Analysis

Sentiment analysis (SA) refers to the use of computational linguistics to identify and extract subjective information in source material, usually unstructured and heterogeneous text data [24]. This chapter summarizes the recent findings of the authors' research team on SA [7, 15, 21–24, 28]. It has two sections. Section 7.1 is word embedding with two Sect. 7.1.1 is about single sense model vs. multiple sense model while Sect. 7.1.2 is about intrinsic vs extrinsic evaluation. Section 7.2 outlines the SA applications.

SA can be extensively applied to a large number of application scenarios such as improving customer service and analyzing social media. There are three types of SA in terms of classification levels. i.e., aspect-level, sentence-level and document-level SA. Document-level SA means predicting the sentiment polarity of a document which is composed of several sentences. Sentence-level SA refers to detecting the sentiment of a single sentence. Furthermore, a document or sentence may describe some aspects of a product like travel product and we sometimes need to know the exact sentiment polarities of each aspect. This is regarded as aspect-level SA. Taking a tourism review as an example, "We went to The Forbidden City last week, and the tour guide is knowledgeable, but it was crowded in The Forbidden City. The worst thing was that it rained when we visited" discusses the cicerone, scenery spot and weather [24]. Here the tour guide is praised whereas the scenery spot and weather are criticized in this tourism review.

Text, audio, image and video comprise the basic information carriers in modern times. Users post plenty of microblogs and tweets on social media platform such as Microblog and Twitter. People can obtain valuable knowledge from different kinds of aforementioned medium. In fact, not only text modality but also audio video modalities convey sentiment information. However, how to effectively extract useful information from unstructured text data remains a challenge. To this end, researchers presented a large amount of sophisticated SA techniques to tackle this challenge. As shown in Fig. 7.1, existing SA techniques can be classified as three types, namely lexicon-based, traditional machine learning based and new approaches.

Y. Shi, *Advances in Big Data Analytics*,
https://doi.org/10.1007/978-981-16-3607-3_7

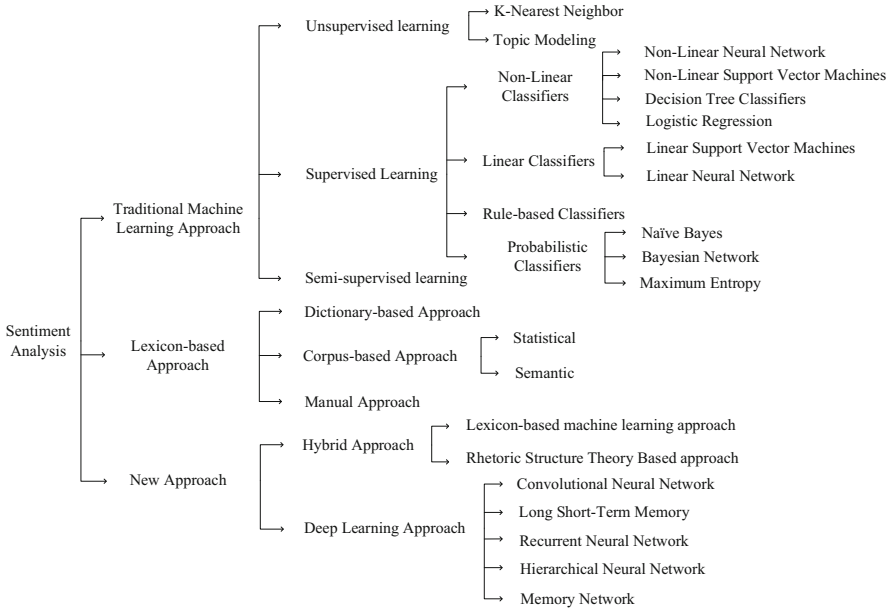

Fig. 7.1 Concept map of SA techniques [24]

Sentiment lexicon, which contains numerous sentiment words with different sentiment polarities or intensities, is one of the most popular lexicon-based approaches. sentiment lexicon can be built by expert knowledge, which requires huge human work and resources. Therefore, some data-driven methods [15, 27] are created to automatically construct the sentiment lexicon and cover as many sentiment words as possible. General steps of sentiment lexicon based SA are as follows:

- extract sentiment words from a given sentence or document.
- sum the sentiment scores of each sentiment words in the sentence or document.
- classify the sentence or document according to the total sentiment score.

Traditional machine learning based approaches, such as naive bayes and rule based approach get good performances on a number of SA tasks including sentence-level and document-level. Taking rule-based approaches as an example, it consists of, as the name partially suggests, a set of rules that classify data in data space. For instance, VADER [11], a simple rule-based modal, contains a gold-standard list of lexical features. extensive experiments on four distinct domain datasets demonstrated that VADER outperformed some benchmarks in the social media domain.

Recently, a lot of new approaches are proposed among which deep learning based and hybrid approaches attract more attention. Deep learning based approaches employ deep learning techniques such as recurrent neural network [25], long short-term memory model [9], convolutional neural network [13], memory networks

[26] and transformer [29] to classify sentences or documents. Hybrid approaches combine the advantages of different methods and get the state-of-the-art on many benchmark SA datasets. For example, cambria et al. [5] proposed an ensemble of symbolic and sub-symbolic AI techniques to perform sentiment analysis. More concretely, they built a new three-level knowledge representation SenticNet 5 for SA, where long short-term memory model (LSTM) was used to discover verb-noun primitives. The generation process largely extend the coverage of basic SenticNet4 [4].

In general, raw data contains numerous useless information. Therefore, it is necessary to pre-process raw data for machine learning and deep learning based SA. For example, HTML contains a lot of HTML tags and non-alphabetic signs which should be removed to improve the data quality. In this paper [7], the authors proposed a data pre-processing framework which is proved to be effective in SA task. The first step is data transformation, where all the HTML tags, non-alphabetic signs, stop words and non-informative words like file, movie, actor, actress, scene are removed from the raw data. After that, stemming is performed on the documents to reduce redundancy. Then, three feature matrices are constructed for each of the datasets based on three different types of feature weighting : TF-IDF, FF and FP. Furthermore, the second step is filtering step, where univariate method chi-squared is utilized to conduct filtering procedure and it measures the dependency between the word and the category of the document it is mentioned in. If the word is frequent in many categories, chi-squared value is low and vice versa. Extensive experiments indicate that appropriate text pre-processing methods including aforementioned data transformation and filtering can significantly improve the classifier's performance. Nevertheless, machine learning and deep learning models cannot directly process words or sentences. In this case, we need a more generalized method to convert the unstructured text data into vector space where words, sentences and even documents can be represented as vectors. This process is named as word embedding or word representation which will be introduced in next section.

7.1 Word Embedding

7.1.1 Methods: Single Sense Model vs Multiple Sense Model

Distributional hypothesis proposed by Harris [8] is the footstone of NLP and word embedding. The word embedding methods consists of three types [23]: matrix-based methods such as TF-IDF matrix, Latent Semantic Analysis (LSA) [14], and GloVe [20]; cluster-based methods, such as Brown [3]; and neural network based methods, such as Neral Network Language Model (NNLM) [1], Log-Bilinear Language model (LBL) [18], C&W [6], skip-gram [17], continuous bag-of-words model (CBOW) [17], and FastText [12], etc.

The aforementioned methods deal with the single sense word embedding, which fails to represent words with multiple meanings. For example, word "apple" is a kind of fruit when it occurs in an article about food or botany; while "apple" refers to a technology company when it comes with MacBook, iphone, etc. In single sense model, only one vector is generated for word "apple" by averaging the results for all meanings, and it cannot comply with all language rules. This calls for a multiple sense word embedding, where each sense of the word corresponds to an independent word vector which serves as an auxiliary vector in the mean while.

How to determine a word's sense according to the current context is one of the greatest challenges for multiple sense word embedding. An intuitive method is to represent the corresponding sense vector via clustering all contexts of the target word by maximizing the probability $P(S)$ in formula (7.1).

$$P(S) = \prod_{i=1}^{n} \prod_{j=1}^{m_i} p(w_i^j | C(w_i^j)) \approx \prod_{i=1}^{n} \prod_{j=1}^{m_i} p(w_i^j | C(w_i)) \tag{7.1}$$

where $S = w_1, w_2, \ldots, w_n$ is the word sequence of a sentence, w_i is the ith word, $C(w_i)$ is the context of word w_i, m_i is the number of total senses of word w_i, w_i^j is the embedding for the jth sense of the ith word. Here, replacing $C(w_i^j)$ with $C(w_i)$ is an effective simplification method.

Based on this intuitive idea, tow-stage hard clustering method [10] was proposed to learn the multiple prototype embedding by means of spherical K-means algorithm and relabeling contexts with corresponding cluster centroids before training. However, the spherical k-means cluster is time-consuming and relabeled context may lose certain detailed information. Zheng Y. et al. [28] presented multi-prototype continuous bag-of-words model (MCBOW) based on a common word vector cell and CBOW, which is illustrated in Fig. 7.2. The objective of this model is to predict multiple target word embedding by exploiting different context information. For each prediction, every context word containing several different sense embeddings (represented by light green modes), the red arrows show how each sense embedding is chosen to form a temp context vector (represented by pink nodes) in Fig. 7.2. Here, u_i^j is the weight (coefficient) for the ith word sense w_i^j. The selection procedure is based on the similarity between the target word's context and the temp context vector. Then the embedding of target word is determined and the vectors is updated through stochastic gradient descent [2] based back-propagation.

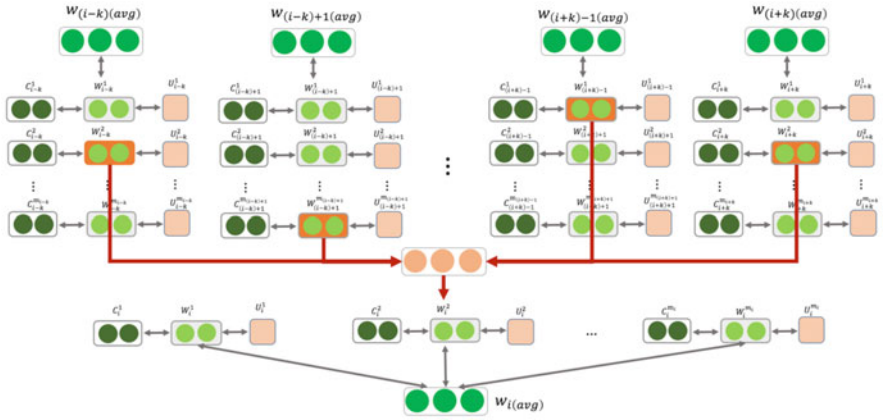

Fig. 7.2 Multi-prototype continuous bag-of-words model [28]

7.1.2 Evaluation: Intrinsic vs Extrinsic

Intrinsic evaluation and extrinsic evaluation are two perspectives to measure the quality of learned word embeddings. Intrinsic evaluation mainly focus on evaluating the word properties, such as the semantic information, syntactic information, and morphology information, etc. While, extrinsic concentrate on specific natural language processing (NLP) tasks in practice.

Some methods employed as intrinsic evaluation for single sense word embedding are shown as follows. Word similarity (WS) measures the similarity between two words by calculating the cosine values of their word vectors; Word analogy (WA) explores the analogical relationship between the word pair and searches for the other matching word pairs satisfying the same relationship; Word synonym detection (WSD) is to find the synonyms given the current token; Selectional preferences (SP) utilizes the verb-noun pairs to construct a verb-noun phrase and noun-verb phrase.

For multiple sense embedding models, each word contains various sense vectors and the aforementioned basic evaluation methods may fail. Thus, four cosine value based distances were proposed by [10], i.e. $AvgSimC$, $GlobalSim$, $AvgSim$, and $LocalSim$.

For extrinsic evaluation, the learned word embeddings are fed to a specific model as features or for the initialization of neural networks. The tasks includes named entity recognition (NER), part-of-speech tagging (POST), text classification (TC), and sentiment analysis (SA), etc. Different from the small intrinsic evaluation datasets, extrinsic evaluation datasets usually contains sufficient training and test samples. However, the input word embedding and the experiment settings may both influence the final results. Thus, it is important to control the experiment settings while evaluating the learned word embeddings using extrinsic evaluation methods [23].

7.2 Sentiment Analysis Applications

As mentioned in Sect. 7.1, sentiment analysis can be applied to a number of application scenarios such as product review analysis and investor sentiment analysis.

The rapid development of online travel websites like TripAdvisor[1] lead to a significant increase in user-generated content (UGC) [16]. Here UGC refers to reviews and interactions among users on travel websites. These user generated reviews will be displayed under a certain travel product and other users who browse the web page of this travel product will see related reviews. Therefore, it is necessary to design an algorithm to automatically analyze such reviews and travel websites can improve their service accordingly. In 2018, [15] proposed a framework named DWWP for tourism review SA, where a domain-specific new words detection method (DW) and word propagation (WP) are presented. Tourism reviews have a number of user-invented domain-specific words such as ” (a large costume show), proper nouns, converted words and multiword expressions (MWEs) which are not included in the existing sentiment lexicons. Manual detection of such words is time-consuming and costly. Therefore, automatic and effective construction of a high-quality tourism-specific sentiment lexicon is of great value. What's more, Chinese SA was even harder due to the lack of segmentation symbols like blank space in English. In this case, the aforementioned four types of words cannot be easily detected by Chinese word segmentation tools. Besides, one limitation of existing data driven sentiment lexicon construction methods is the lack of robustness. To this end, DWWP framework is presented to solve the above issues and build a high-quality tourism-specific sentiment lexicon.

Figure 7.3 is the concept map of the proposed DWWP framework. As shown in the figure, raw data is first collected, pre-processed and then fed into domain new words detection (DW) block. First, Chinese word segmentation tool is utilized to segment the raw text data into a series of single morphemes. Then the authors proposed a statistical indicator named Assembled Mutual Information (AMI) to determine if a candidate word is a valid word or not. The formula of AMI is as follows:

$$AMI(w) = \sum_{j}(log\frac{n_w/N}{\sqrt[T]{\prod_{i=1}^{T}[(n_{w_i}^{j} - n_w + sf)/N]}}) \tag{7.2}$$

where w means the candidate new word which is composed of T single morphemes. n_w is the occurrence number of w, whereas n_{m_i} is the occurrence number of m_i. N stands for the total number of documents. If the AMI score of a candidate new word is high, it is more probable to be a valid new word and vice versa. With the help of this domain new words detection algorithm, plenty of domain new

[1] https://www.tripadvisor.com.sg/.

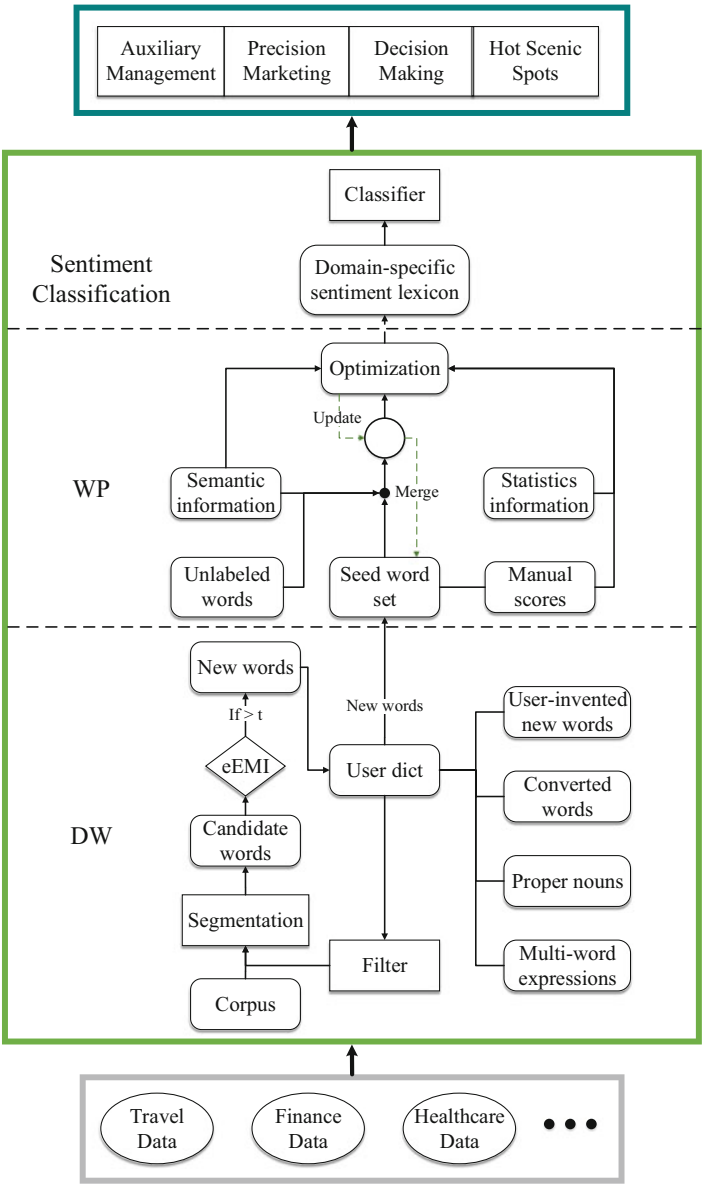

Fig. 7.3 Domain words detection and word propagation system [15]

words, converted words, proper nouns and multiword expressions can be obtained. Next, a word propagation algorithm is employed to build a high-quality tourism-specific sentiment lexicon. Here the algorithm starts from a small set of seed words (detected new words are partly included in the seed words). Then both the semantic similarity (measured by cosine value between two word vectors) and statistical similarity (measured by pointwise mutual information) are used to measure the real world similarity between two words. Additionally, an optimization function which considers seed word, semantic similarity, statistical similarity is designed to tune the sentiment scores of sentiment words. Extensive experiments on Chinese tourism review dataset demonstrate the superiority of the proposed DWWP SA framework.

Besides the product review analysis, SA can also be applied to investor sentiment analysis. Shi et al. [21] proposed a text mining system using data cleaning, text representation, feature extraction and a two-step sentiment analysis techniques to identify individual investor sentiment and comply an index. Then the investor sentiment index is applied to Chinese stock market to study its relationship with CSI 300 stock index returns. Investor sentiment measure process is as follows:

- Extract individual investor posts from large-scale online stocks forum posts on East money stock forum.
- Employ linguistic module to process posts data, where text representation, feature extraction and noise classifier are utilized.
- The processed text data is fed into a sentiment identification block (support vector machine classifier in this section). Here bullish-bearish classifier is used to identify investor sentiment as either bullish or bearish.
- Build investor sentiment index and the index formula is:

$$M_t = ln\left[\frac{1 + M_t^{BUY}}{1 + M_t^{SELL}}\right] \tag{7.3}$$

  where M_t^{BUY} is the total bullish posts in time interval t, and M_t^{SELL} is the total bearish posts in time interval t.

Based on 5,163,210 online posts on East money stock forum, investor sentiment index is built by means of the aforementioned investor sentiment index construction method. The result shows that on average, investor sentiment is towards bullish, which verifies the viewpoint of investors' irrational biases in behavioral finance [19]. Moreover, this chapter studied the relationship between CSI 300 index and investor sentiment index. The similarity rate of investor sentiment is 60.76% and is much higher than that of institutional view, which suggests that investor sentiment from online stock forum can predict stock returns, especially short term. To this end, this chapter established a 3 order VAR model which indicates the asymmetric effects of investor sentiment on stock market.

Sentiment analysis, a promising research field in natural language processing (NLP), attracts more attention in recent years. With the development of deep learning based NLP techniques, the performance of SA increases fast. Basic SA techniques such as word embedding as well as the application scenarios will be the future research focus.

References

1. Bengio, Y., Ducharme, R., Vincent, P., Jauvin, C.: A neural probabilistic language model. J. Mach. Learn. Res. **3**(Feb), 1137–1155 (2003)
2. Bottou, L.: Large-scale machine learning with stochastic gradient descent. In: Proceedings of COMPSTAT'2010, pp. 177–186. Springer (2010)
3. Brown, P.F., Desouza, P.V., Mercer, R.L., Pietra, V.J.D., Lai, J.C.: Class-based n-gram models of natural language. Computational Linguistics **18**(4), 467–479 (1992)
4. Cambria, E., Poria, S., Bajpai, R., Schuller, B.: Senticnet 4: A semantic resource for sentiment analysis based on conceptual primitives. In: Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers, pp. 2666–2677 (2016)
5. Cambria, E., Poria, S., Hazarika, D., Kwok, K.: Senticnet 5: Discovering conceptual primitives for sentiment analysis by means of context embeddings. In: Thirty-Second AAAI Conference on Artificial Intelligence (2018)
6. Collobert, R., Weston, J.: A unified architecture for natural language processing: Deep neural networks with multitask learning. In: Proceedings of the 25th International Conference on Machine Learning, pp. 160–167 (2008)
7. Haddi, E., Liu, X., Shi, Y.: The role of text pre-processing in sentiment analysis. Procedia Comput. Sci. **17**, 26–32 (2013)
8. Harris, Z.S.: Distributional structure. Word **10**(2-3), 146–162 (1954)
9. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Computation **9**(8), 1735–1780 (1997)
10. Huang, E.H., Socher, R., Manning, C.D., Ng, A.Y.: Improving word representations via global context and multiple word prototypes. In: Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1, pp. 873–882. Association for Computational Linguistics (2012)
11. Hutto, C.J., Gilbert, E.: Vader: A parsimonious rule-based model for sentiment analysis of social media text. In: Eighth International AAAI Conference on Weblogs and Social Media (2014)
12. Joulin, A., Grave, E., Bojanowski, P., Mikolov, T.: Bag of tricks for efficient text classification. Preprint (2016). arXiv:1607.01759
13. Kim, Y.: Convolutional neural networks for sentence classification. Preprint (2014). arXiv:1408.5882
14. Landauer, T.K., Foltz, P.W., Laham, D.: An introduction to latent semantic analysis. Discourse Processes **25**(2-3), 259–284 (1998)
15. Li, W., Guo, K., Shi, Y., Zhu, L., Zheng, Y.: Dwwp: Domain-specific new words detection and word propagation system for sentiment analysis in the tourism domain. Knowl. Based Syst. **146**, 203–214 (2018)
16. Marine-Roig, E.: Online travel reviews: A massive paratextual analysis. In: Analytics in Smart Tourism Design, pp. 179–202. Springer (2017)
17. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. Preprint (2013). arXiv:1301.3781

18. Mnih, A., Hinton, G.: Three new graphical models for statistical language modelling. In: Proceedings of the 24th International Conference on Machine Learning, pp. 641–648 (2007)
19. Odean, T.: Are investors reluctant to realize their losses? J. Finance **53**(5), 1775–1798 (1998)
20. Pennington, J., Socher, R., Manning, C.D.: Glove: Global vectors for word representation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1532–1543 (2014)
21. Shi, Y., Tang, Y.R., Cui, L.X., Long, W.: A text mining based study of investor sentiment and its influence on stock returns. Econom. Comput. Econom. Cybernet. Stud. Res. **52**(1), 183–199 (2018)
22. Shi, Y., Zheng, Y., Guo, K., Li, W., Zhu, L.: Word similarity fails in multiple sense word embedding. In: International Conference on Computational Science, pp. 489–498. Springer (2018)
23. Shi, Y., Zheng, Y., Guo, K., Zhu, L., Qu, Y.: Intrinsic or extrinsic evaluation: An overview of word embedding evaluation. In: 2018 IEEE International Conference on Data Mining Workshops (ICDMW), pp. 1255–1262. IEEE (2018)
24. Shi, Y., Zhu, L., Li, W., Guo, K., Zheng, Y.: Survey on classic and latest textual sentiment analysis articles and techniques. Int. J. Inf. Tech. Dec. Making **18**(04), 1243–1287 (2019)
25. Tang, D., Qin, B., Liu, T.: Document modeling with gated recurrent neural network for sentiment classification. In: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pp. 1422–1432 (2015)
26. Weston, J., Chopra, S., Bordes, A.: Memory networks. Preprint (2014). arXiv:1410.3916
27. Wu, F., Huang, Y., Song, Y., Liu, S.: Towards building a high-quality microblog-specific chinese sentiment lexicon. Decis. Support Syst. **87**, 39–49 (2016)
28. Zheng, Y., Shi, Y., Guo, K., Li, W., Zhu, L.: Enhanced word embedding with multiple prototypes. In: 2017 4th International Conference on Industrial Economics System and Industrial Security Engineering (IEIS), pp. 1–5. IEEE (2017)
29. Zhu, Z., Zhou, Y., Xu, S.: Transformer based chinese sentiment classification. In: Proceedings of the 2019 2nd International Conference on Computational Intelligence and Intelligent Systems, pp. 51–56 (2019)