2022 International Conference on Industrial Automation, Robotics and Control Engineering (IARCE) | 978-1-6654-5440-7/22/$31.00 ©2022 IEEE | DOI: 10.1109/IARCE57187.2022.00025

Interpretable sentiment analysis based on sentiment words' syntax information

Qingqing Zhao*
School of computer science, Beijing Institute of Technology, China.
Haidian Qu, Beijing, China.
zqing102@163.com

Huaping Zhang
School of computer science, Beijing Institute of Technology, China.
Haidian Qu, Beijing, China.
Kevinzhang@bit.edu.cn

Jianyun Shang
School of computer science, Beijing Institute of Technology, China.
Haidian Qu, Beijing, China.
shangjia@bit.edu.cn

***Abstract* — In recent years, with the vigorous development of deep learning, the pre-trained models such as Bert and GPT have been brilliant, and the sentiment analysis task has made increasingly outstanding achievements. The sentimental accuracy of model recognition is getting higher and higher, and the related application fields are also getting wider and wider. However, because deep learning is a black box model, its internal decision-making mechanism is not transparent to users, and it can't reasonably explain the output of the model, which brings great limitations to the application of sentiment analysis. In this paper, we integrate the syntax tree based on sentiment words into the embedding module and the attention module of the interpretable sentiment model, and filter the evidence tokens output by the model to achieve the interpretability of sentiment analysis. The model is validated on the DuTrust dataset, and the experiment proves the validity of sentiment words' syntax in interpretable sentiment analysis.**

Keywords* — *Interpretability, Sentiment Analysis, Deep Learning, Syntax Tree, Sentiment Words

I. INTRODUCTION

The deep neural networks [1][2] can map word vectors to higher-level representations and encode rich language and semantic information in the potential vector space. With the continuous development of neural networks, models based on deep learning have achieved excellent results in various natural language processing tasks, and the use scenarios of deep learning models are more and more extensive.

However, deep learning has long been criticized for its lack of interpretability. Model interpretability refers to the understanding of the internal mechanism and the results of the model. The higher the interpretability of the model, the easier it is for people to understand why they make certain decisions or predictions. Especially when we need a machine learning model to deal with unknown situations and emergencies, it is very important for the model to provide reasonable explanations. Such as in the medical system and the financial system.

Sentence level sentiment analysis is a basic classification task in natural language processing. Its purpose is to classify a given text into specific sentiment polarity (positive, negative, neuter or other). sentiment analysis plays an important role in public opinion analysis or commodity recommendation.

With the gradual maturity of traditional sentiment classification tasks, simple sentiment polarity prediction has achieved good results, and interpretable sentiment analysis has become the next topic of interest. Interpretable sentiment analysis needs to explain the reasons for the prediction to the user, for example, "*The price is expensive and the service is poor.* ", and the prediction evidence "*price expensive, service poor* ", needs to be extracted as the reason for the prediction of sentimental polarity. As shown in Figure 1:

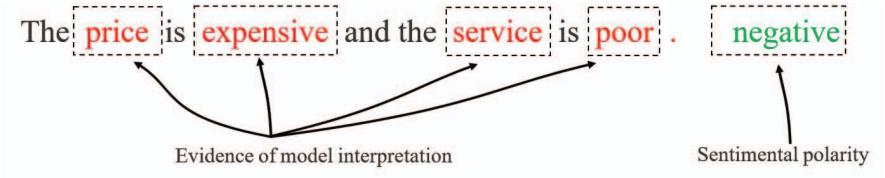

Figure 1. The explanation of the sentiment analysis model shows that the red words in the left side are the sentimental tokens given by the model, and the right side is the sentimental polarity of the output.

At present, some studies have focused on interpretable sentiment. Yadav et al. [3] realized the interpretability of aspect level sentiment analysis by converting complex position related text semantics into binary form and mapping all features to bag of words. Perikos et al. [4] used Hidden Markov Model (HMM) to indicate the sentimental part in a sentence. These methods have achieved certain results, but they ignore the importance of semantic information for interpretability.

The syntax tree contains rich semantic information, which has achieved good results in the sentimental analysis as a sentence feature [5][6]. However, inputting the syntax tree of the whole sentence into the model will contain too much interference information and cannot be well concentrated in the sentimental information.

Therefore, in our work, we take the existing sentimental knowledge as the center, annotate the words related to the sentimental words, including the positions of the sentimental words themselves, and integrate the syntactic information into the self-attention layer through the syntactic analysis tools. In the process of deep learning, we get not only the semantic information of words, but also the syntactic information related to sentimental words. Our model is shown in Figure 2.

The major contributions of this paper are summarized as follows:

978-1-6654-5440-7/22/$31.00 ©2022 IEEE
DOI 10.1109/IARCE57187.2022.00025

- The location information of sentiment words and their related words are integrated into the embedding module of the model, so that the model can fully learn the location information based on sentiment words.
- The syntactic information based on sentiment words is integrated into the self-attention module, so that the model can make full use of syntactic information in the process of self- attention learning.
- Experiments show that the performance of interpretable sentiment analysis can be improved by integrating the syntactic information based on sentiment words into the model.

II. Related Work

In this section, we briefly review the following related work: 1) interpretable sentiment analysis methods, 2) interpretable model evaluation methods, and 3) interpretable model evaluation indicators.

A. Interpretable Sentiment Analysis Methods

Sun Xiaowan [7] used self-attention, multi-head attention and other mechanisms to obtain the global structure information of the text and some information related to specific aspects at the same time, so as to obtain the importance of each word to the classifier decision and explain the classification results of the model. For the document level sentiment analysis task, it obtains the sentiment tendency score of fine-grained interpretation by fusing the external knowledge of sentiment, and then uses the score to guide the decision-making process of the model classifier. Bacco L et al. [8] connected the two transformer models in series, and used the attention weight of the second transformer to construct a summary as an explanation of the output. For text level emotional analysis, the author applied a single transformer to classify individual sentences in the document, and combined the summary scores of the sentences to construct a summary as an explanatory output. Meng y et al. [9] explained the prediction

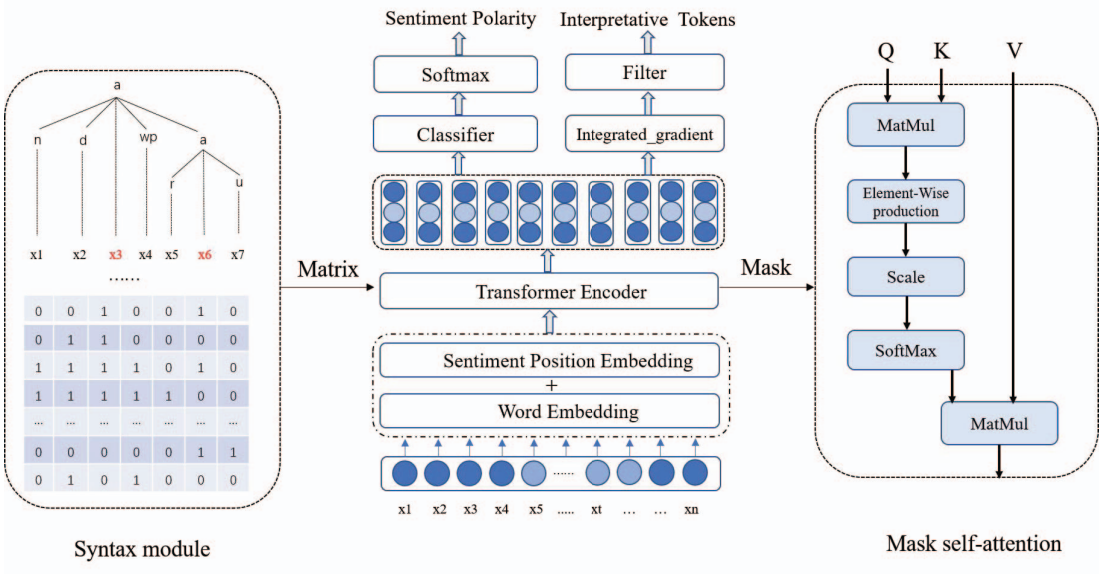

Figure 2. As shown in the figure, the overall structure of our model is shown. On the left side is the syntax analysis module. The obtained syntactic information matrix M is transferred to the transformer layer of the model. After the processed embedding is transferred to the transformer, the model masks the attention according to the matrix M, and finally the output values are transferred to different processors to obtain the sentimental polarity and interpretable tokens of the sentence.

of the model by jointly examining the training history samples and the test stimuli. The test stimuli were first identified by the gradient-based method, and then the gradient based significance scores were propagated to the training examples by using the influence function to determine which training example parts made the model focus on the test stimuli. In order to get rid of the limitations of prior information and manual annotation, Chen h et al. [10] proposed a variational word mask (VMASK) method to automatically learn important words of specific tasks and reduce irrelevant information for classification, thus ultimately improving the interpretability of model prediction.

B. Interpretable Model Evaluation Methods

Bastings J et al. [11] analyzed the rationality of the interpretability of attention and saliency methods, and pointed out that in the interpretable field, saliency methods are more faithful and reliable than attention mechanisms. The following is a list of common sentimental interpretation methods:

- **Attention-based methods** [12]: Attention-based methods use attention weights as importance scores, and the acquisition of attention weights depends on the specific model architecture.
- **Gradient-based method** [13]: Gradient-based saliency methods compute the importance of a specific input feature (e.g., vector dimension, word or span) based on the first-order derivative with respect to that feature.
- **Erasure-based method** [14]: Erasure-based methods compute the saliency score of an input feature by erasing the feature, passing the altered input again into the model and measuring the output change.
- **LRP-based method**: Layerwise Relevance Propagation [15], or LRP for short, provides full,

layer-by-layer decomposition for the attribute values from the model prediction backward to the input features in a recursive fashion.

- **Linear-based method(LIME)**: LIME [16] uses the token weights learned by the linear model as importance scores.

C. Interpretable Model Evaluation Indicators

Plausibility and Faithfulness [17] are frequently used in the evaluation of interpretable models. The plausibility refers to the degree of fitting between the evidence provided by the model and the manually labeled evidence. The faithfulness is to verify the consistency of the evidence under data disturbance.

Plausibility：we take Macro-F1 as the evaluation metric. Assuming that S_i^p is the *i*-th prediction evidence of input and S_i^g is the *i*-th standard evidence of input, the formula for calculating F1 value is as follows [18]:

$$\mathrm{F1} = \frac{1}{N}\sum_{i=1}^{N}\left(2 * \frac{P_i * R_i}{P_i + R_i}\right) \#(1)$$

Where

$$P_i = \frac{|S_i^p \cap S_i^g|}{|S_i^p|} \#(2)$$

And

$$R_i = \frac{|S_i^p \cap S_i^g|}{|S_i^g|} \#(3)$$

Faithfulness： we evaluate faithfulness based on the consistency of explanations under perturbations. we use Mean Average Precision (MAP) to evaluate the consistency of their token importance lists, as shown below.

$$MAP = \frac{\sum_{i=1}^{|X^a|}\left(\sum_{j=1}^{i} F(X_j^a, X_{1:i}^o)\right)/i}{|X^a|} \#(4)$$

Where X^o and X^a represent the sorted token importance list of the original instance and the adversarial instance respectively. $|X^a|$ represents the number of tokens in the list X^a. $X_{1:i}^o$ represents a portion of the list X^o , which consists of top-*i* tokens. The function F(*x* , Y) is used to determine whether the token *x* belongs to the list Y. If *x* is in the list Y, F(*x*, Y) returns 1. The high MAP indicates the high consistency [18].

III. MODEL

In this section, we present the details of sentiment words' syntax information and neural networks framework for sentiment interpretation. In order to integrate the syntactic information centered on sentiment words into the model, we first analyze the sentences by using syntactic analysis tools to extract the sentiment words and their related words contained in the sentences. Then the position information of sentiment words and their related words is integrated into the embedding module, and the syntactic information based on sentiment words is integrated into the self-attention module. In the following sections, we will introduce the specific details of the model implementation.

A. Syntactic Information based on sentiment words

Assuming that the input sentence is *x* and *x* contains *n* tokens, we define *x* as $x = < x_1, x_2 ... x_n >$. The sentence *x* can generate a corresponding syntax tree **T** through the syntax analysis tool, and the sentimental position mark of each token can be represented by *T* (x). Find the sentiment word **S** contained in the grammar tree **T** and the word **R** associated with **S**, mark its sentiment position **P** as **1**, and mark the rest as **0**. The formula is as follows:

$$P = \begin{cases} 1, & if(x = S \ or \ x = R) \\ 0, & other \ . \end{cases} \#(5)$$

After obtaining the sentimental position information, we can input it into the sentiment position embedding layer to obtain the vector representation $< w_1, w_2 ... w_t ... w_n >$, where w_t represents the sentimental position information embedding of the *t*-th mark in the sentence. The whole process is shown in the Figure 3:

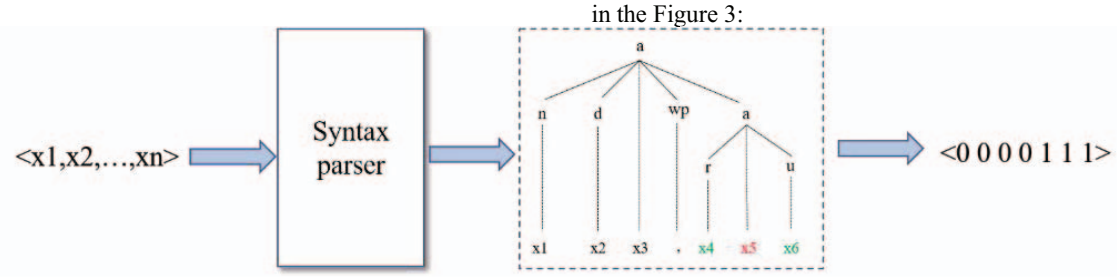

Figure 3. Syntactic location information annotation process based on sentiment words, where x5 is sentiment word, x4 and x6 are words associated with it. Sentimental words and their surrounding words are marked as 1, and other words are marked as 0.

B. Deep Learning for Sentiment interpretation

Embedding. we obtain corresponding vector representation from a lookup table as $< e_1, e_2, \ldots, e_t, \ldots, e_n >$, where e_t stands for the word embedding of the *t*-th token in the sentence. Then we transfer the marked sentimental location information to sentiment_pos_embedding layer and get position_embedding. So

$$Embedding = word\_embedding + position\_embedding \#($$

SKEP pre-training model. SKEP [19] is a pre-training model for sentiment analysis, which can learn a unified sentiment representation for multiple sentiment analysis tasks. In the process of pre-training, it adds the prior knowledge of sentiment and focuses on the text features of some sentiment words. In our experiment, SKEP is used to learn the general sentiment language, which provides better initializ-ation for the model.

Transformer. Transformer [20] is a deep learning model using self-attention mechanism, which can assign different weights according to the importance of each part of input data. The most important part of transformer is the self-attention mechanism. The core formula of self-attention is:

$$Attention(Q,K,V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \#(7)$$

In this formula, Q, K and V represent query, key and value respectively. In order to fully apply the self-attention layer to the syntactic information based on sentiment words, our model adopts the idea proposed in paper to improve the self-attention layer.

Masked self-attention. Inspired by this paper [21], we first encode the syntactic information of sentimental words into a matrix M, and then transfer the matrix M to the self-attention layer for operation. The formula is as follows:

$$Mask\_Attention(Q,K,V,M) = softmax\left(\frac{QK^T \odot M}{\sqrt{d_k}}\right)V \#(8)$$

Where M represents for the matrix of syntax mask and $\odot$ denotes an operator for element-wise production. The whole process is shown in Figure 4.

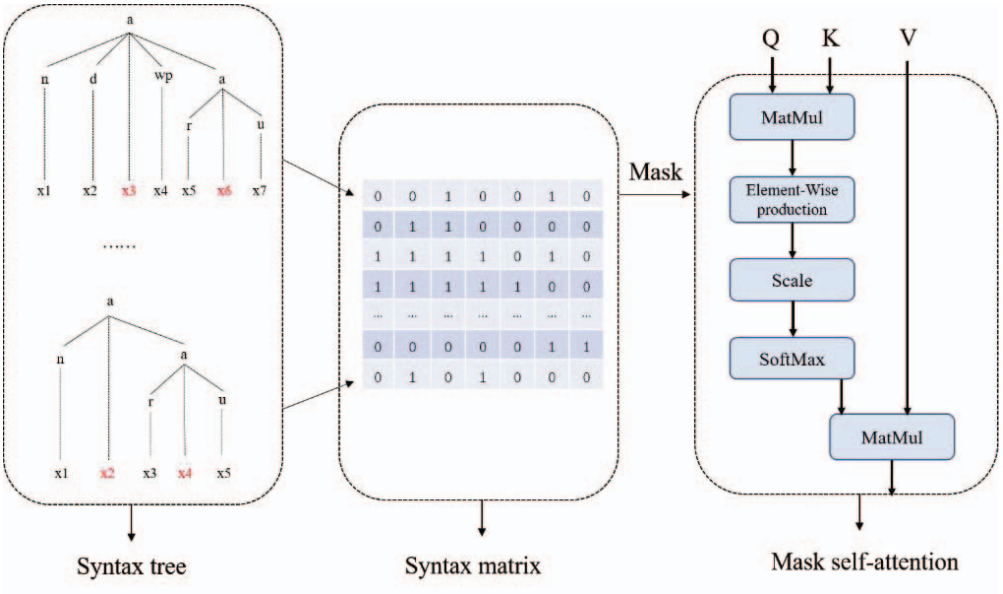

Figure 4: Mask Self-attention, the matrix M generated by the syntax tree will be calculated in the self-attention layer.

IV. EXPERIMENT

A. Dataset

The sentiment vocabulary contains positive / negative words and their corresponding polarity labels (for example, happy → positive, sad → negative). Since most of the words are constructed manually or by machine learning algorithms, there is a congenital defect of incomplete vocabulary. In this paper, the sentiment dictionary we use is from [1].

[1] https://github.com/Syd-Q/Text-Mining-Sentiment-Analysis/tree/master/emotion_dict

The training set adopts the Chnsenticorp[2] dataset, which is a commonly used dataset in Chinese sentiment analysis, including online shopping reviews of hotels, notebooks and books. The test data adopts the sentimental interpretability assessment task in Dutrust dataset [18], which is the trusted assessment data set published by Baidu.

B. Compared methods

To comprehensively evaluate the performance of our model, we compared it with the following methods:

- **LSTM：** Long short-term memory network (LSTM) [22] is a kind of time cycle neural network, which is specially designed to solve the long-term dependence problem of general RNN.
- **Roberta：** Roberta model [23] mainly modifies the Bert model from the two aspects of model structure and data. Currently, for the Chinese pre-training language model, Roberta may be the best choice for most of the time and most of the downstream tasks. In the experiment, we compared the two versions Roberta-base and Roberta-large at the same time.
- **Ernie：** Ernie [24] was proposed by Baidu to solve the problem of insufficient pre-training of Bert in Chinese corpus. Through this model, many Chinese tasks can be solved by Ernie model. In the experiment, we compared the two versions Ernie-base and Ernie-large at the same time.

[2] https://aistudio.baidu.com/aistudio/datasetdetail/10320

C. Implementation detail

AdamW [25] is used as our optimizer with learning rate of 3e-6, dropout rate of 0.1 and a mini-batch of 16 samples. We implement all models in the Pytorch environment using the same input data, embedding size, dropout rate, optimizer, etc.

D. Experimental results

In order to keep consistent with the recent baseline methods, we used the Accuracy, F1 and MAP as evaluation indicators and compared the three

interpretable methods of IG(Integrated_gradient), ATT(attention) and LIME. The experimental results are summarized in the form of percentages in Table 1, and the best performance is shown in bold.

At the same time, we also recorded the accuracy of different model structures on the Chnsenticorp testset, and the results are shown in Table 2:

TABLE 1. INTERPRETABILITY EVALUATION RESULTS ON DUTRUST DATASET OF THREE INDICATORS. THE RESULTS OF LSTM AND ERNIE IN THE FIGURE ARE FROM [18], AND THE RESULTS OF ROBERTA ARE FROM [26].

Models+Methods	Acc	F1	MAP
LSTM + IG	56.8	38.8	59.6
RoBERTa-base + IG	62.4	37.4	64.1
RoBERTa-large + IG	65.3	35.0	40.6
Ernie-base + IG	65.2	35.1	36.4
Ernie-large + IG	68.2	37.8	34.1
Skep + IG	66.5	38.6	30.5
Skep + PE + IG	66.7	37.8	35.3
Skep + PE + MSA +IG	69.4	42.5	79.6
LSTM + ATT	56.8	33.3	69.8
RoBERTa-base + ATT	62.4	33.2	69.2
RoBERTa-large + ATT	65.3	23.3	75.9
Ernie-base + ATT	65.2	24.6	65.1
Ernie-large + ATT	68.2	27.9	64.6
Skep + ATT	66.5	29.6	60.7
Skep + PE + ATT	66.7	29.8	68.3
Skep + PE + MSA +ATT	69.4	32.4	**81.4**
LSTM + LIME	56.8	37.2	59.4
RoBERTa-base + LIME	62.4	41.5	61.0
RoBERTa-large + LIME	65.3	41.4	62.9
Ernie-base + LIME	65.2	37.8	46.7
Ernie-large + LIME	68.2	39.8	42.3
Skep + LIME	66.5	39.3	43.8
Skep + PE + LIME	66.7	39.2	44.2
Skep + PE + MSA + LIME	**69.4**	**44.8**	80.2

TABLE 2. PERFORMANCES OF SIX MODELS ON CHNSENTICORP TEST SET.

Models	Acc_{chn}
LSTM	86.8
RoBERTa-base	94.3
RoBERTa-large	94.7
ERNIE-base	95.4
ERNIE-large	95.8
SKEP	**96.3**

E. Experimental analysis

From the experimental results in Table 1, it can be seen that IG performs better on F1 and ATT performs better on MAP. LIME combines the performance of both. LSTM is the simplest model architecture among them, but its experimental results are very good, which to a certain extent shows that the simpler the model structure, the more explanatory it is. Roberta's accuracy is not as high as Ernie's, but the MAP and F1 are higher than Ernie's.

The effect of using the SKEP model alone is not ideal. However, in order to take advantage of the sentimental knowledge pretrained in the skep model, we still adopted SKEP as our pre-training model. After adding PE (position embedding) to the SKEP model, the model has been improved to a certain extent, especially the MAP under the ATT method. After adding MSA (masked self-attention), the accuracy of the model increased from 66.7% to 69.4%, the F1 increased from 39.2% to 44.8%, and the MAP increased from 68.3% to 81.4%.

As shown in Table 2, the accuracy of the model on the Chnsenticorp test set has reached 96.3%, but the accuracy on the evaluation set has dropped significantly. The main reason is that the evaluation set is not from the Chnsenticorp dataset, which leads to the uneven distribution of the training set and the evaluation set. At the same time, because our experiment is based on the sentimental interpretability of sentimental common sense, it is more effective for evaluating explicit emotions. However, the evaluation set covers three kinds of sentimental descriptions: single sentiment, multi sentiment and inexplicit sentiment. It can't judge the sentimental polarity of multi sentiment and inexplicit sentiment well, which also leads to the decrease of accuracy.

Finally, in order to better understand the extracted experimental results, we give several examples for your reference, as shown in Table 3.

TABLE 3. EXAMPLES OF EXPERIMENTAL RESULTS

Sentence	Polarity	Standard evidences	Prediction evidences
Although it bangs a very cliched drum at times, this crowd-pleaser's fresh dialogue, energetic music, and good-natured spunk are often infectious.	positive	["fresh ", " dialogue "],[" energetic ", " music "], ["good ", " natured ", " spunk "]	fresh, good, natured
A particularly	negative	["rubbish", "movie",	rubbish, poor

rubbish movie shop, poor service attitude.	"shop"],	["poor", "service", "attitude"]

V. Conclusion and Future Work

In this paper, the position information and syntax information of sentiment words are integrated into the neural network model for sentence level interpretable sentiment analysis. We realize the location information embedding based on sentiment words, and the focus is to make full use of the semantic information around the sentiment words. Since the self-attention layer adds the semantic information knowledge, the self-attention mechanism of the model is more accurate. At the same time, due to the use of sentimental prior knowledge, the sentimental common sense of the model is further enhanced. The experimental results demonstrate the effectiveness of our proposed model.

The future work should pay more attention to the interpretability of implicit sentiment, and how to extract and interpret tokens from such sentiment instances. At the same time, more and more effective interpretable sentiment datasets are urgently needed to be constructed. The sentiment interpretation methods can also be discussed from many aspects, so as to find a more suitable interpretation method for the model.

VI. Acknowledgments

We thank the anonymous reviewers for their constructive feedback and great help, and all those who put forward valuable opinions in the process of writing the paper.

References

[1] Chen Y. Convolutional neural network for sentence classification[D]. University of Waterloo, 2015.

[2] Zaremba W, Sutskever I, Vinyals O. Recurrent neural network regularization[J]. arXiv preprint arXiv:1409.2329, 2014.

[3] Yadav, Rohan K., et al. "Human-level interpretable learning for aspect-based sentiment analysis." Proceedings of the AAAI Conference on Artificial Intelligence. Vol. 35. No. 16. 2021.

[4] Perikos, Isidoros, Spyridon Kardakis, and Ioannis Hatzilygeroudis. "Sentiment analysis using novel and interpretable architectures of Hidden Markov Models." Knowledge-Based Systems 229 (2021): 107332.

[5] Dai J, Yan H, Sun T, et al. Does syntax matter? a strong baseline for aspect-based sentiment analysis with roberta[J]. arXiv preprint arXiv:2104.04986, 2021.

[6] Tian Y, Chen G, Song Y. Enhancing aspect-level sentiment analysis with word dependencies[C]//Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume. 2021: 3726-3739.

[7] Sun Xiaowan. Research on interpretable emotion analysis model based on deep learning [D]. Jilin: Jilin University,2020.

[8] Bacco L, Cimino A, Dell'Orletta F, et al. Explainable sentiment analysis: a hierarchical transformer-based extractive summarization approach[J]. Electronics, 2021, 10(18): 2195.

[9] Meng Y, Fan C, Sun Z, et al. Pair the dots: Jointly examining training history and test stimuli for model interpretability[J]. arXiv preprint arXiv:2010.06943, 2020.

[10] Chen H, Ji Y. Learning variational word masks to improve the interpretability of neural text classifiers[J]. arXiv preprint arXiv:2010.00667, 2020.

[11] Bastings J, Filippova K. The elephant in the interpretability room: Why use attention as explanation when we have saliency methods?[J]. arXiv preprint arXiv:2010.05607, 2020.

[12] Wiegreffe S, Pinter Y. Attention is not not explanation[J]. arXiv preprint arXiv:1908.04626, 2019.

[13] Sundararajan M, Taly A, Yan Q. Axiomatic attribution for deep networks[C]//International conference on machine learning. PMLR, 2017: 3319-3328.

[14] Li J, Monroe W, Jurafsky D. Understanding neural networks through representation erasure[J]. arXiv preprint arXiv:1612.08220, 2016.

[15] Bach S, Binder A, Montavon G, et al. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation[J]. PloS one, 2015, 10(7): e0130140.

[16] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. " why should i trust you?" explaining the predictions of any classifier. In Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining, pages 1135–1144, 2016.

[17] Jacovi A, Goldberg Y. Towards faithfully interpretable NLP systems: How should we define and evaluate faithfulness?[J]. arXiv preprint arXiv:2004.03685, 2020.

[18] Wang L, Liu H, Peng S, et al. DuTrust: A Sentiment Analysis Dataset for Trustworthiness Evaluation[J]. arXiv preprint arXiv:2108.13140, 2021.

[19] Tian H, Gao C, Xiao X, et al. SKEP: Sentiment knowledge enhanced pre-training for sentiment analysis[J]. arXiv preprint arXiv:2005.05635, 2020.

[20] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need[J]. Advances in neural information processing systems, 2017, 30.

[21] Bai J, Wang Y, Chen Y, et al. Syntax-BERT: Improving pre-trained transformers with syntax trees[J]. arXiv preprint arXiv:2103.04350, 2021.

[22] Hochreiter S, Schmidhuber J. Long short-term memory[J]. Neural computation, 1997, 9(8): 1735-1780.

[23] Liu Y, Ott M, Goyal N, et al. Roberta: A robustly optimized bert pretraining approach[J]. arXiv preprint arXiv:1907.11692, 2019.

[24] Sun Y, Wang S, Li Y, et al. Ernie: Enhanced representation through knowledge integration[J]. arXiv preprint arXiv:1904.09223, 2019.

[25] Loshchilov I, Hutter F. Fixing weight decay regularization in adam[J]. 2018.

[26] Wang L, Shen Y, Peng S, et al. A Fine-grained Interpretability Evaluation Benchmark for Neural NLP[J]. arXiv preprint arXiv:2205.11097, 2022.