Aspect Based Sentiment Analysis Model Based on Multi-Channel Feature Fusion

Hao Wang and Zhonglin Zhang

April 18, 2024

Aspect based sentiment analysis model based on multi-channel feature fusion

Hao Wang and Zhonglin Zhang
(School of Electronic and Information Engineering, Lanzhou Jiaotong University, Lanzhou,730070)

Abstract

Aspect Based Sentiment Analysis (ABSA) is a natural language processing (NLP) task used for recognizing sentiment polarity about given aspects in the sentence. The current methods mostly combine the semantics and syntax of text. Which disregards the assistance of integrating external knowledge to enhance feature representation. In this context, the paper introduces a Multi-Channel Feature Fusion (MCFF) model, which captures sentiment feature from three diverse perspectives: context-based, syntax-based and aspect-term-concept-enhanced. Firstly, the model proposed in this paper learns context and syntax representations in parallel to adequately extract semantic features. Secondly, after enhancing aspect terms with external knowledge, the model merges them with specific aspect features to obtain knowledge-enhanced aspect representations. We extract sentiment features from multiple perspectives and fuse these feature representations to acquire the final text representation. The model in this paper has been verified to have excellent performance on standard datasets.

Keywords: Aspect Based Sentiment Analysis; Graph Convolutional Network; Concept Enhancement; Feature Fusion

1 Introduction

Given the swift evolution of the Internet, users are posting a large number of opinions and views on social platforms every day, which includes their emotions and attitudes towards certain events. Sentiment analysis[1] refers to analyze the expressions of a large amount of text to determine whether they convey positive, negative, or neutral sentiments. It is a subtask of NLP[2]. Through the classification of text granularity, researchers divide text sentiment analysis into three types, which we call document level, sentence level and aspect level sentiment analysis. They represent the sentiment polarity[3] expressed in the entire text, individual sentences and specific aspects within sentences respectively. In most cases, coarse-grained sentiment analysis and judge the sentiment polarity of just a text or a single sentence, it is very unreasonable. In daily life, our comments may contain different views and opinions. Taking the sentence "The $landscape$ is $great$ but the $weather$ is a little $bad.$" as an example, in this sentence, the sentiment polarities expressed by the opinion words corresponding to these two aspects are positive and negative, respectively. Therefore, in such cases, we should conduct more fine-grained[4] methods such as aspect-level sentiment analysis.

We are living in the era of big data explosion, users are posting an array of opinions and views on social platforms every day, which includes their emotions and attitudes towards certain events. Therefore, understanding and analyzing users' emotional tendencies is extremely important. Aspect based sentiment analysis differs from conventional sentiment analysis tasks in that it focuses on identifying categories of sentiment in specific aspects of a sentence. This fine-grained task enables us to analysis user's emotional polarity towards specific aspects of the text precisely. This task is being applied to various realistic domains in daily life, such as product design, brand management and public opinion monitoring. Through aspect-based sentiment analysis, understanding users' perceptions of products and services can be achieved, thus we can improve product design and marketing strategies. In early sentiment analysis tasks, it mainly relied on manual feature engineering, such as designing sentiment lexicons, dependency information, etc., and then employing conventional classifiers such as maximum entropy and SVM for sentiment classification.

2 Relative work

2.1 Aspect Based Sentiment Analysis

The rapid development of various self-media platforms has made it increasingly difficult for us to acquire helpful information from various sources. At the same time, internet commentaries have become an important source for people seeking decision-making reference information. Therefore, we must devote more effort to researching fine-grained sentiment analysis methods. Traditional machine learning methods rely on manually designed features, requiring a considerable amount of manual effort for feature extraction and selection. What’s more, they struggle to grasp the complex semantic relationships between words and are ill-equipped to handle lengthy texts and deep semantic information. However, traditional machine learning methods still have certain use cases, especially in scenarios with small datasets or high interpretability requirements.

In contrast, with the rise of neural networks, especially the successful application of attention and memory mechanisms in various natural language processing tasks, deep learning[5] has achieved significant improvements in performance. The biggest advantage of deep learning is that it does not rely on manually defined features but instead can autonomously study the sentiment information embedded in the text. This alleviates the burden of feature engineering. Furthermore, methods based on deep learning can learn complex representations and semantic information from the data. It has the capability to grasp the relationships between words and contextual semantic information effortlessly, which can promote the performance in sentiment analysis tasks.

Long Short-Term Memory[6] (LSTM) was put forward by Hochreiter and Schmidhuber in 1997. Due to its design characteristics, LSTM is very suitable for problems involving sequential data. It is equipped with long-term and short-term memory units, enabling it to effectively handle long-term dependencies. In 2016, Tang et al.[7] proposed Target-Dependent LSTM (TD-LSTM). It utilizes two LSTMs to respectively model the information on the left and right sides of the target word which fully considering its contextual information. Wang et al.[8] was the first to incorporate attention mechanism into sentiment analysis tasks and propose ATAE-LSTM in 2016. It can capture essential segments of text relevant to attribute words, thus improving the accuracy of attribute word extraction. Which lifts the effectiveness of models to some extent.

Although ATAE-LSTM introduces attention mechanism to dynamically identify and weight aspect terms, its attention weights are calculated through simple weighting without considering global contextual information. This may lead to suboptimal performance of the model when dealing with complex sentence structures. Ma et al.[9]proposed the Interactive Attention Network (IAN) to effectively model the relationship between the target word composed of multiple words and the context, focusing on the part with higher correlation with the target word. However, these methods only focus on the contextual sequence features of the sentence, resulting in the syntactic dependence between the contextual words becoming a neglected link.

Of course, the semantic information contained in the text is crucial for sentiment discrimination. Meanwhile, it is significant to give greater consideration to the syntactic dependencies among words in sentence. Zhang et al.[10] utilize syntactic dependency trees to capture node information between words in sentences, constructing an adjacency matrix based on this. They then employ graph convolutional networks to model the relationship between syntactic information and contextual semantics.

Although these models have achieved relatively good performance, not considering the incorporation of external knowledge to enrich semantic features and enhance text representation may impact the performance of ABSA tasks.

2.2 Application of external common sense in ABSA

External knowledge plays a crucial role in ABSA tasks. This external common sense can include domain knowledge, sentiment lexicons, factual knowledge and so on. We rely on our previously acquired knowledge when we read a sentence. For instance, when we hear “summer” and we might associate it with concepts like “hot weather”, “happy vacation” and so forth. We leverage this external knowledge to enhance our outstanding. Without accessing such external knowledge, it becomes challenging to grasp the semantics when we encounter an unfamiliar domain.

SenticNet serves as a repository of 100,000 natural language concepts linked to semantics, emotions, and polarity. SenticNet is capable of recognizing sentiment polarity, intensity, and relevance within text. Therefore, SenticNet is widely applied to help us understand text content. By incorporating the sentiment information from SenticNet, researchers can enhance the performance of sentiment analysis tasks and gain deeper insights into the sentiment content of the text. Ma et al.[11] were the first to integrate external sentiment knowledge into LSTM networks for discerning the sentiment polarity of specific aspects. Liang et al.[12] constructed a Graph Convolutional Network (GCN) by integrating effective knowledge from SenticNet in 2022, embedding sentiment knowledge into the dependency graph of sentences to explore the sentiment features between aspect words and contextual words.

Inspired by the aforementioned work, this paper proposes the MCFF model, which utilizes three different perspectives: context-based, syntax-based and aspect-based sentiment feature representations. These perspectives are then fused together to estimate the sentiment polarity.

3 Architecture of MCFF

The overall structure of our proposed model is shown in Figure 1. MCFF extracts feature representations from three different perspectives. First, SenticNet is used to enhance the prior conceptual knowledge of aspect words to obtain enhanced feature representations of aspect words. Then, context information is encoded through a bidirectional long short-term memory network to obtain context-based and syntax-based feature representations. Finally, we fuse the three feature representations to get the final text representation for sentiment classification.

3.1 Aspect term concept enhancement

This paper employs SenticNet for aspect term conceptualization, yielding a concept set $C = \{c_1, c_2, \cdots, c_n\}$, comprising n concept data points, where c_i represents the i-th concept in the set.

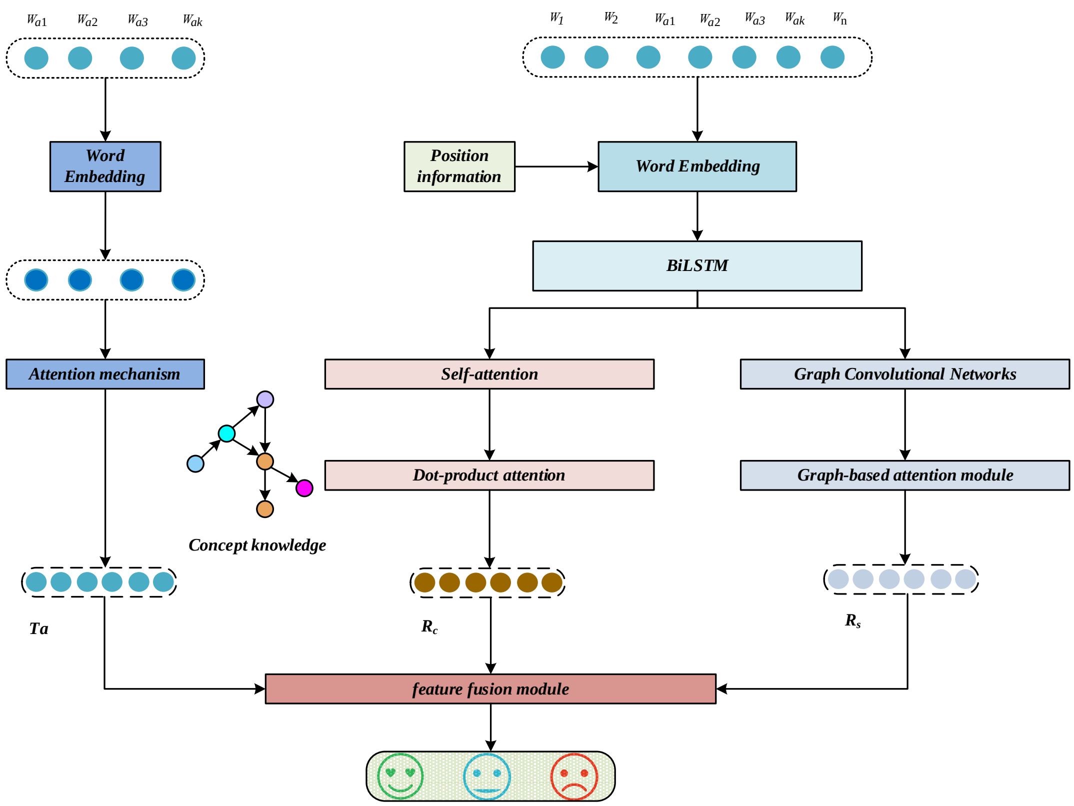

Figure 1: Architecture of the MCFF model

Through self-attention mechanism calculation, the concept vector for each aspect term is obtained. The calculation process is as follows:

$$\alpha_i = \text{softmax}\left(V \cdot \tanh\left(W_{c_i}\right) + b\right) \tag{1}$$

In the equation, W and b represents the weight matrix and bias term, V is the weight vector, and α_i represents the attention score for the i-th concept in the concept set. The final concept vector for the aspect term is obtained through weighted calculation.

$$T_i = \sum_{i=1}^{v} \alpha_i c_i \tag{2}$$

The concept vector T and the aspect vector α are concatenated to obtain the concept-enhanced aspect features $Ta = \{Ta_1, Ta_2, \cdots, Ta_k\}$.

3.2 Contextual feature representations

Context-aware representations have been successful in improving language understanding capabilities, thereby achieving better performance. In this paper, the word embedding model maps the input to the corresponding vector space through the embedding matrix $E \in \mathbb{R}^{|V| \times d_w}$, we set $|V|$ as the size of the vocabulary and d_w as the dimension of word embeddings. Additionally, we encode relative positional features into the embeddings of sentence.

Then, we utilize BiLSTM to learn contextual information of the sentence. As a member of the recurrent neural network family, BiLSTM consists of a forward LSTM and a backward LSTM.

Compared to classical RNNs, this network effectively alleviates the gradient explosion or vanishing problem through gate mechanisms. The combination of forward and backward mechanisms better captures bidirectional semantic dependencies, making it particularly suitable for modeling textual context information. The computational rules are described as follows:

$$h_i^s = \left[\overrightarrow{LSTM}(x_i^s), \overleftarrow{LSTM}(x_i^s)\right], \qquad i \in \{1, m\} \tag{3}$$

$$h_j^t = \left[\overrightarrow{LSTM}(x_j^t), \overleftarrow{LSTM}(x_j^t)\right], \qquad j \in \{1, n\} \tag{4}$$

Therefore, we gain the hidden outputs of BiLSTM for the sentence as $H_c^s = \{h_1^s, h_2^s, \dots, h_i^s, \dots, h_m^s\}$, which preserves contextual information, and the target representation as $H_c^t = \{h_1^t, h_2^t, \dots, h_n^t\}$. In this paper, the weights were distributed to every word through self-attention mechanism based on its relevance and importance relative to other words in the sentence. This enables the model to focus on significant word interactions and capture potential semantic relationships between words. Subsequently, we employ another dot-product attention mechanism to assign weights from each word in S to T, obtaining the final context representation.

3.3 Syntax-Based Representations

In the absence of distinguishing between different dependency types, a single dependency relationship may lead to information loss. Considering enhancing the modeling of dependency relationship types, we employ interactive learning to deeply represent each type of dependency. To enable the model to more fully learn syntactic relationships within sentences, we design a dependency relationship embedding module in the graph convolutional network. This module obtains a sentence representation enhanced with dependency type embeddings. It shares the same hidden vector representation encoded by Bi-LSTM as the previous section, thereby reducing the overall model parameter count, i.e., $H_s = H_c^s$.

First, we establish the syntactic dependency tree and acquire the adjacency matrix A. In actual operation, we consider each word adjacent to its child nodes and itself, setting the adjacency node value to 1. The update of the hidden state vector through GCN is as follows:

$$H_s^{l+1} = ReLU\left(\frac{AH_s^{(l)}W^l}{(D+1)} + B^l\right), \qquad l \in \{0,1\} \tag{5}$$

where D is the degree matrix of A $(i.e, D_{ii} = \sum_j A_{ij}$), and W^l is the weight matrices for the $(l+1)$-th GCN layer. B^l is the bias. The original hidden state vector of H_s is H_s^0, and the final output of the GCN is $H_s^{(2)}$.

Finally, we fuse the feature representations obtained from the three branches, concatenate them and employ the cross-entropy loss function to guide optimization and training.

$$\mathcal{L} = -\sum_i \sum_i y_i^j \log\left(p_i^j\right) \tag{6}$$

Here, i represents an instance in the ABSA dataset, and j represents the sentiment polarity.

4 Experiments

4.1 Datasets and Settings

The experiments were conducted on three publicly available datasets: Laptop14, Restaurant14 and the Twitter dataset. Further details can be found in Table 1. The evaluation metrics encompass accuracy and macro F1 score.

Datasets	Division	#Positive	#Negative	#Neutral
Laptop14	Train	980	858	454
	Test	340	128	171
Restaurant14	Train	2159	800	632
	Test	730	195	196
Twitter	Train	1567	1563	3127
	Test	174	174	346

Table 1: Datasets

In the experiments, we utilized GloVe embeddings with a dimensionality of 300. We set the learning rate to $1e-3$. We applied dropout to the word embeddings with a dropout rate of 0.5 to alleviate overfitting. Adam optimizer was employed for optimization and training.

4.2 Experiments analysis

The paper selected several baseline models for comparison, as shown in Table 2 of the experimental results.

Method	Laptop14		Restaurant14		Twitter	
	Acc.(%)	F1.(%)	Acc.(%)	F1.(%)	Acc.(%)	F1.(%)
ATAE-LSTM	68.88	63.93	78.60	67.02	68.64	66.60
RAM	74.49	71.35	80.23	70.80	69.36	67.30
TNet-AS	76.54	71.75	80.69	71.27	74.97	73.60
IAN	72.05	67.38	79.26	70.09	72.50	70.81
ASGCN	75.55	71.05	80.86	72.19	72.15	70.40
OURS	76.91	75.21	82.16	73.27	75.65	73.76

Table 2: Experiment results

The experimental results from Table 2 indicate that the proposed model in this paper outperforms the others, validating the effectiveness and superiority of integrating context-based, syntax-based and aspect concept enhancement-based. Compared to single-feature extraction models, the model proposed in this paper demonstrates significant improvements in both accuracy and macro F1 score. The experimental results of this study indicate that extracting sentiment features from multiple perspectives greatly enhances the complementarity between context, syntax and external knowledge, thereby enhancing the performance of sentiment analysis models.

5 Summary and Outlook

We capture sentiment features from three diverse perspectives: context-based, syntax-based and aspect words concept-enhanced. Experiments on three standard datasets have explained the effectiveness of our model. Future work will focus on investigating the noise and information redundancy issues introduced by the incorporation of external knowledge and exploring more efficient feature fusion mechanisms.

* Project supported by the National Natural Science Foundation of China (Grant Nos.61662043) and Phased Research Results of Gansu Philosophy and Social Sciences Planning Project (20YB056)

References

[1] Wankhade M, Rao A C S, Kulkarni C. A survey on sentiment analysis methods, applications, and challenges[J]. Artificial Intelligence Review, 2022, 55(7): 5731-5780.

[2] Zhou M, Duan N, Liu S, et al. Progress in neural NLP: modeling, learning, and reasoning[J]. Engineering, 2020, 6(3): 275-290.

[3] Bordoloi M, Biswas S K. Sentiment analysis: A survey on design framework, applications and future scopes[J]. Artificial Intelligence Review, 2023, 56(11): 12505-12560.

[4] Xuxu W, Peng C, Huan J. Micro-Blog Fine-Grained Sentiment Analysis Based on Multi-Feature Fusion[J]. Data Analysis and Knowledge Discovery, 2023, 7(12): 102-113.

[5] Chauhan G S, Nahta R, Meena Y K, et al. Aspect based sentiment analysis using deep learning approaches: A survey[J]. Computer Science Review, 2023, 49: 100576.

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

[7] Tang D, Qin B, Feng X, et al. Effective LSTMs for target-dependent sentiment classification[J]. arXiv preprint arXiv:1512.01100, 2015.

[8] Wang Y, Huang M, Zhu X, et al. Attention-based LSTM for aspect-level sentiment classification[C]//Proceedings of the 2016 conference on empirical methods in natural language processing. 2016: 606-615.

[9] Ma D, Li S, Zhang X, et al. Interactive attention networks for aspect-level sentiment classification[J]. arXiv preprint arXiv:1709.00893, 2017.

[10] Zhang C, Li Q, Song D. Aspect-based sentiment classification with aspect-specific graph convolutional networks[J]. arXiv preprint arXiv:1909.03477, 2019.

[11] Ma Y, Peng H, Khan T, et al. Sentic LSTM: a hybrid network for targeted aspect-based sentiment analysis[J]. Cognitive Computation, 2018, 10: 639-650.

[12] B. Liang, H. Su, L. Gui, E. Cambria, and R. Xu, “Aspect-based sentiment analysis via affective knowledge enhanced graph convolutional networks,” Knowl.-Based Syst., vol. 235, 2022, Art. no. 107643.