ORIGINAL ARTICLE

DCASAM: advancing aspect-based sentiment analysis through a deep context-aware sentiment analysis model

Xiangkui Jiang[1] · Binglong Ren[1] · Qing Wu[1] · Wuwei Wang[1] · Hong Li[1]

Abstract

Aspect-level sentiment analysis plays a pivotal role in fine-grained sentiment categorization, especially given the rapid expansion of online information. Traditional methods often struggle with accurately determining sentiment polarity when faced with implicit or ambiguous data, leading to limited accuracy and context-awareness. To address these challenges, we propose the Deep Context-Aware Sentiment Analysis Model (DCASAM). This model integrates the capabilities of Deep Bidirectional Long Short-Term Memory Network (DBiLSTM) and Densely Connected Graph Convolutional Network (DGCN), enhancing the ability to capture long-distance dependencies and subtle contextual variations.The DBiLSTM component effectively captures sequential dependencies, while the DGCN component leverages densely connected structures to model intricate relationships within the data. This combination allows DCASAM to maintain a high level of contextual understanding and sentiment detection accuracy.Experimental evaluations on well-known public datasets, including Restaurant14, Laptop14, and Twitter, demonstrate the superior performance of DCASAM over existing models. Our model achieves an average improvement in accuracy by 1.07% and F1 score by 1.68%, showcasing its robustness and efficacy in handling complex sentiment analysis tasks.These results highlight the potential of DCASAM for real-world applications, offering a solid foundation for future research in aspect-level sentiment analysis. By providing a more nuanced understanding of sentiment, our model contributes significantly to the advancement of fine-grained sentiment analysis techniques.

Keywords Aspect-based sentiment analysis · Context awareness · Sentiment polarity · Latent information

Introduction

Since 2004, there has been an exponential growth in the number of users registered on social networks. Owing to its pervasive popularity, this growth has shown no signs of abating significantly year after year [1]. 2023 social networking statistics show that as of July 2023, 4.88 billion users worldwide use social media, with a penetration rate of 70% of Internet users, a figure that will continue to grow in the years to come. The perspectives voiced by individuals on social media serve as the primary wellspring of critical information for trading officers, governments, and researchers alike. Through the extraction, assessment, and analysis of the sentiment latent within the opinions sourced from these data repositories, traders, governments, and researchers can garner valuable insights into trade, policy formulation, and advisory, enabling them to expedite decision-making processes and enhance decision quality.

Aspect-Based Sentiment Analysis (ABSA) aims to identify the sentiment polarity of each specific aspect term within a sentence, induced by a manually labeled predictive model. This task is considered an intriguing trend in the field of social media artificial intelligence [2]. With the rapid growth of the digital economy today, aspect-level sentiment analysis has become a crucial component in various AI systems, such as stance detection, recommendation systems, decision-

✉ Xiangkui Jiang
jiangxiangkui@xupt.edu.cn

Binglong Ren
binglongren@stu.xupt.edu.cn

Qing Wu
wuqing@xupt.edu.cn

Wuwei Wang
wangwuwei@xupt.edu.cn

Hong Li
lihong@xupt.edu.cn

[1] School of Automation, Xi'an University of Posts and Telecommunications, No. 618 West Chang'an Street, Xi'an 710121, China

making, and opinion monitoring [3]. Traditional sentiment analysis methods primarily focus on inferring the overall sentiment orientation of a specific text or document, usually categorized into three different classes: positive, negative, or neutral [4]. However, ABSA aims to predict the specific sentiment polarity related to the focal aspect within a given sentence or document. For example, in the sentence "This car has excellent performance but is too expensive," ABSA would identify the sentiment polarity for both "performance" and "price" separately. In this instance, the review highlights two aspects of the car: "performance" and "price," with "performance" receiving positive sentiment and "price" eliciting negative sentiment. Traditional text-level or document-level sentiment polarity mining methods often struggle to accurately predict the polarity of specific aspects, as they overlook the nuanced polarity variations between different aspects. ABSA is increasingly being used in the construction and development of many practical applications [5]. For example, manufacturers can identify which aspects or components of their products are positively received by consumers and which are considered lacking, allowing them to improve the product by retaining favorable features and addressing areas needing improvement. Relevant government public opinion monitoring agencies can also use sentiment analysis based on vast amounts of online commentary data to assist in formulating policies that align with public sentiment.

To address the tasks of Aspect-Based Sentiment Analysis, numerous studies have been conducted over the past decade. Broadly speaking, recent work has focused on improving the discriminative accuracy of aspect term representation to enhance sentiment polarity recognition performance. In the field of ABSA, researchers have proposed various methods, such as knowledge-based, machine learning-based, or a combination of both. Additionally, to further link aspect terms and opinion words, some studies have constructed dependency trees of sentences and used Graph Convolutional Networks (GCNs) to generate representations of aspect terms [6]. Subsequent comparative studies have shown that GCN-based models perform well in sentiment analysis tasks, particularly in ABSA, as evidenced by models such as AHGCNWIN, Hier-GCN, and GCNSA. It can be inferred that the use of GCNs can improve the performance metrics of previous techniques in ABSA tasks. However, GCN-based sentiment analysis models typically have the following limitations:

- Two-layer GCNs perform best in capturing neighborhood information, and continuing to add layers fails to capture more structural information.
- It is not possible to consider useful inter-contextual information about what is hidden between words.

The presence of implicit emotions is widespread in ABSA datasets, presenting a significant challenge in handling senti-ment analysis tasks. This challenge prompted us to design a novel ABSA method that combines a contextual focus mechanism, DBiLSTM, and DGCN, aiming to enhance the efficiency of capturing the subtle nuances of context related to aspects and their associated sentiments. The construction of the model is driven by several factors. Firstly, the emergence of pre-trained models like BERT has garnered significant interest in the academic community. Many researchers have utilized BERT and similar pre-trained models for aspect-level sentiment analysis, demonstrating the feasibility of integrating such models into sentiment analysis [7]. BERT, or Bidirectional Encoder Representations from Transformers, is a model that leverages attention mechanisms to analyze input data sequences. It employs an encoder-decoder architecture to establish relationships between words. This architecture allows BERT to capture bidirectional contextual information from the input sequence, thereby generating high-quality contextualized word representations. Unlike sequential word embedding models, BERT processes entire sentences or extended paragraphs at once using the Transformers architecture. This capability enables BERT to capture rich contextual information and dependencies among words in the input text, producing more robust context-aware textual representations [8]. However, many previous aspect-level sentiment analysis studies primarily focused on predicting the polarity of different aspects within a sentence, often overlooking the nuanced interactions between sentiment polarity and local context. Research indicates that semantic information is indispensable in linking words with sentences, relying on contextual cues to extract implicit aspects embedded within sentences or documents [9]. Therefore, by leveraging the subtleties of contextual data, it is possible to map connections between different data entities to improve the accuracy of sentiment classification. For instance, in the sentence, "I went to the travel agency to inquire about travel plans, and they offered me four travel options," the phrase "four travel options" should convey positive sentiment but is often considered neutral. This highlights the importance of context surrounding specific aspects in enhancing the accuracy of ABSA.

Additionally, these studies typically employ single-head or multi-head attention mechanisms, where each head operates independently. To enhance the attention effect, we chose to adopt the Talking-Heads Attention (THA) mechanism. Unlike traditional methods, THA combines the independently operating heads to produce an enhanced attention effect, aiming to improve results [10]. Finally, unlike basic syntactic models that primarily rely on statistical features, DBiLSTM has the ability to capture contextual information by encoding sentences bidirectionally. This enables DBiL-STM to capture the true meaning of words within their context. GCNs have demonstrated outstanding performance in effectively learning graph representations, and they excel in a range of applications and tasks, including classification

tasks. Therefore, combining these models can significantly enhance the efficiency of aspect-level sentiment analysis. Our proposed method comprises the following stages: first, BERT is used to convert words in a sentence into vector representations. During this process, data is segmented into global and local aspects based on Semantic Relative Distance (SRD) [11]. Subsequently, text feature extraction is performed by integrating the contextual focus mechanism and the talking-head attention mechanism. Next, DBiLSTM is used to generate contextual word representations based on the word vectors. Finally, aspect-level sentiment classification is performed through a tightly integrated GCN layer. Experiments conducted on three established datasets demonstrate that our proposed model outperforms previous context-based GCN methods. The contributions of our proposed method can be summarized as follows:

1. Utilizing the Contextual Dynamic Mask (CDM) and Talking-Heads Attention (THA) mechanism to extract global and local contextual features. Subsequently, combining local and global features enhances the model's target efficacy;
2. Developing a densely connected GCN network to enhance the ability to capture intricate structural information;
3. Proposing the integration of BERT, DBiLSTM, and DGCN models to explicitly uncover hidden contextual information between words, thereby obtaining more valuable insights.

The subsequent sections of this paper are organized as follows: Section "Related work" delivers a detailed survey of advanced methodologies and mechanisms that form the groundwork for our proposed model. In section "DCASAM model", the research difficulties pertaining to our method are specified, shedding light on the challenges' definitions and the relevant research inquiries. In addition, a mathematical model intended to surmount these challenges is introduced. Section "Model training and output" shows the complete training process of the model and its output for different contexts. Section "Experiments" concisely details the dataset and the comparative results of our proposed model against renowned existing methods. Section "Concluding remarks" concludes with the findings of our research and charts possible trajectories for future exploration in this area.

Related work

Sentiment analysis, often referred to as opinion mining, is defined as the process of extracting opinion tuples, represented as $< OpinionHolder, TargetEntity, TargetAspect, Sentiment, Timestamp >$, abbreviated as $< H, E, A, S, T >$ [12]. Aspect-based sentiment analysis involves extracting "aspect-sentiment" pairs from textual content, with each pair representing a focal aspect and its corresponding sentiment. This technique is widely used to review product or service feedback, aiding in assessing consumer sentiment towards different aspects of a product or service.

In the early stages of aspect-based sentiment analysis, the primary method was to construct sentiment lexicons and assign sentiment values to the sentiment words in documents [13]. Although constructing sentiment lexicons is relatively straightforward, the effectiveness largely depends on the quality and comprehensiveness of the lexicon. This dependency often results in limited diversity of sentiment words. Moreover, sentiment lexicons are mostly created manually, which involves a substantial amount of work.

Subsequently, in the research of sentiment analysis tasks, scholars have tended to adopt machine learning methods. For instance, Brinati et al. [14] applied Extreme Random Trees, Logistic Regression, and Naive Bayes algorithms to predict COVID-19 patients using white blood cells and platelets, demonstrating that machine learning can serve as an alternative to rRT-PCR testing tools. Additionally, Tutsoy and Koç [15] proposed a deep self-supervised machine learning algorithm that combines multidimensional adaptive feature elimination, self-feature weighting, and a novel feature selection method for multidimensional health risk classification based on blood test data, significantly enhancing the model's performance. Similarly, A self-supervised learning algorithm was introduced in the field of image super-resolution, which estimates a specific downscaling kernel through generative and discriminative networks [16]. These methodological innovations, such as self-supervised learning and specific kernel estimation, also provide important insights for sentiment analysis tasks. They indicate that self-supervised learning methods can be used to automatically extract and optimize features, reducing dependence on large-scale labeled data, thereby improving the robustness and accuracy of models.

Following the progress in aspect-based sentiment analysis research, scholars have increasingly tended to adopt deep learning methods. Wang et al. [17] incorporated Long Short-Term Memory (LSTM) models into tasks of aspect-based sentiment analysis, unveiling the Target-Dependent Long Short-Term Memory (TD-LSTM) and Target-Connection Long Short-Term Memory (TC-LSTM) models, both predicated on LSTM architecture. Although these models demonstrated improved performance relative to the foundational LSTM model, the sentiment attributes extracted included emotions linked to non-aspect terms, thus affecting the precision to some extent. To more thoroughly incorporate aspect information, Wang et al. [17] proposed an enhancement of the LSTM model with the integration of attention mechanisms. This model emphasizes key sentence fragments via attention mechanisms, and empirical findings further highlight

the importance of aspect terms in distinguishing sentiment. Additionally, Woźniak et al. [18] proposed a data balancing strategy that combines Bidirectional Long Short-Term Memory (BiLSTM) and decision tree models to address imbalanced medical data in IoT systems. This study demonstrated high efficiency in health status prediction, particularly in automated diagnostic support, achieving accuracy and recall rates exceeding 96%. This approach, which integrates deep learning with traditional machine learning methods, further enhances the effectiveness of aspect-based sentiment analysis, indicating the significant potential of deep learning models in processing complex contextual information. Zhang et al. [19] presented the Interactive Attention Network model, which strengthens the interplay between aspect terms and contextual data through interactive attention mechanisms. Initially, aspect terms and context were modeled, with the attention for aspect information being extracted via a pooling layer. Following this, attention mechanisms were utilized to enable interactive processes between the two, ultimately leading to the extraction of the definitive sentiment attributes. Jing et al. [20] unveiled the Multi-Granularity Attention Network (MGAN), tackling the issue of information loss that plagues earlier coarse-grained attention approaches. MGAN seizes the interactive data between aspect terms and context using a meticulous attention mechanism. This data is then amalgamated with coarse-grained attention, constructing a framework for a multi-granularity attention network that effectively addresses the problem of information loss.

In later studies, graph neural networks have seen extensive use in aspect-based sentiment analysis (ABSA) endeavors, especially through the deployment of graph convolutional networks (GCN), owing to their superior representational prowess. The GCN represents a specialized variant of the graph neural network model, specifically designed for graph-structured data. Given a graph G = V, E, A, with V and E symbolizing the node and edge sets respectively, and A signifying the adjacency matrix, GCN focuses on mastering node representations through convolutional layers, adeptly assimilating information from adjacent nodes, as described by Chen et al. [21]. A single-layer convolution in GCN can only utilize information from neighboring nodes to represent a node, while multi-layer convolutions in GCN can capture information from a broader neighborhood.

To deepen the GCN layers, the convolutional layers of the CNN model proposed in study [22] can be adopted. However, directly using CNN on GCN in this manner does not account for context-aware word representations and has not been applied to ABSA. Various GCN-based ABSA methods also emphasize the consideration of contextual information. The AHGCN-WIN model proposed in study [23] is an explicit aspect model based on graph node context representations. This model initially uses a Bidirectional Long Short-Term Memory (BiLSTM) network to grasp the context informa-

tion of neighboring words. Subsequently, a multi-layer Graph Convolutional Network (GCN) model is employed to capture the sentiment features of aspect terms and other words in the opinion. Finally, a masking layer is cleverly used to identify segments related to specific aspects. The AHGCN-WIN model performs excellently, achieving an accuracy of 82.02% and an F1 score of 73% on the restaurant and laptop datasets, respectively. However, this approach has two limitations outlined in the first part. In study [24], the HierGCN framework was proposed, which integrates dual graph convolutional layers while encoding both the intrinsic connections between aspects and the extrinsic connections between sentiment and aspects. The Hier-GCN model achieved the highest F1 score of 74.55% on the Restaurant-16 dataset. Although it partially addresses the first limitation mentioned earlier, it does not fully resolve the second limitation. The Adaptive Probabilistic Graph Structure Embedding Variational Autoencoder (APGVAE) proposed by Ke et al. [25], through embedding graph structure information, achieves decoupled representation learning of higher-order features, providing important insights for sentiment analysis tasks. In study [26], the GCNSA framework was proposed, which combines the functionalities of GCN and LSTM architectures. It processes text graphs through convolutional GCN layers to obtain hidden representations of the entire text while integrating attention mechanisms in LSTM to simultaneously capture specific regional information. GCNSA achieved the highest F1 score of 78.12% on the restaurant dataset. However, GCN-based aspect-level sentiment analysis methods have two limitations mentioned in the first part.

As research has progressed, addressing uncertainties has become a major research focus. Tutsoy et al. [27] proposed a completely model-free adaptive control method capable of operating under parametric uncertainties, non-parametric uncertainties, and random control signal delays. Their study demonstrated that stable control systems could be achieved without using any model by learning Q-functions and control strategies. The key to this method lies in using radial basis functions to approximate control signals and continuously optimizing control strategies through a combination of exploration and exploitation. In contrast, our approach to handling similar types of uncertainties employs a combination of contextual focus mechanisms, DBiLSTM, and DGCN to enhance the efficiency of capturing subtle contextual variations related to aspects and their associated sentiments. Additionally, we have incorporated the work of Zhang and Lu [28] by introducing multi-head attention mechanisms to further enhance model performance in complex scenarios. This approach not only allows us to address uncertainties effectively but also significantly improves the accuracy and efficiency of sentiment classification tasks.

Additionally, with the growing prevalence of pre-trained models, numerous researchers have begun exploring their

applicability in aspect-based sentiment analysis. By fine-tuning these models, researchers have made significant advancements in sentiment classification tasks. Among these pre-trained models, BERT has become a popular choice due to its exceptional language understanding and representation capabilities. BERT achieves deeper semantic understanding through bidirectional encoder representations, which enables it to perform exceptionally well in various natural language processing tasks. For instance, study [29] employed a method involving the creation of auxiliary sentences, pairing them with the original sentences. By carefully adjusting existing BERT models, researchers redefined the aspect-oriented sentiment evaluation task as a classification of sentence pairs, resulting in significant improvements in analysis. Similarly, study [30] investigated the information encoding in the intermediate layers of the BERT model, effectively addressing previous deficiencies in semantic understanding. They integrated an attention mechanism in the intermediate layers of the model to more effectively extract text weights, significantly enhancing BERT's performance in ABSA tasks. Study [31] proposed a BERT-based sentiment analysis framework, demonstrating excellent performance in tweet sentiment classification tasks by combining models such as CNN, RNN, and BiLSTM. This study conducted experiments on six tweet datasets collected from Kaggle, showing that the combination of BERT and these deep learning models excelled in accuracy, precision, recall, and F1 score. Compared to other pre-trained models, BERT's bidirectional attention mechanism and deep semantic encoding provide it with unparalleled advantages in sentiment analysis tasks. Overall, BERT holds great promise for applications in sentiment analysis, and its unique advantages make it a powerful tool in this field.

Based on the above reasons, we have the foundation and motivation to design an ABSA model that combines the contextual focus mechanism, DBiLSTM, and DGCN to improve the efficiency of capturing the subtle nuances of aspects and their related sentiments. Our proposed Deep Context-Aware Sentiment Analysis Model (DCASAM) integrates the features of DBiLSTM and DGCN, effectively capturing long-distance dependencies and deep latent information in complex text sequences. This integration enhances the model's ability to handle subtle contextual variations, improving the accuracy and efficiency of sentiment classification tasks. Comprehensive experimental evaluations on multiple public datasets show that DCASAM outperforms other models in terms of accuracy and F1 score. These improvements not only demonstrate DCASAM's stability and effectiveness in handling and analyzing implicit and ambiguous sentiment data but also lay a solid foundation for further advancements in ABSA. In summary, the superior performance of DCASAM in ABSA tasks validates its

design concept and application value, and future research can further explore and optimize based on this foundation.

DCASAM model

In this segment, our aim is to explicate the principle and procedural framework of the DCASAM(Deep Context-Aware Sentiment Analysis Model) we suggest. Four main elements constitute the model: an initial BERT pre-training segment, a layer for extracting features, another for learning these features, and finally, the output segment, all illustrated in Fig. 1.

BERT pre-trained model layer

The purpose of crafting the BERT pre-training model lies in augmenting the efficacy of tasks related to natural language processing. This document employs a dual BERT pre-training model approach to extract both local and global semantic information from text, aiming to bolster the text model's performance for subsequent tasks. The input text's words are modeled by these frameworks. As an embedding layer, the BERT pre-training tier operates as a pre-trained model for sequence-to-sequence tasks in language understanding [32]. The pre-training regime of BERT incorporates a pair of distinct tokens, identified as [CLS] and [SEP]. The [CLS] token functions to represent the whole sentence semantically. In this investigation, samples are pre-processed into the structures "x=[CLS]+context+[SEP]" and "x=[CLS]+context+[SEP]+aspect word+[SEP]". Two differentiated BERT pre-training models, represented by X^l and X^g, are engaged in the task of modeling the words present in the input sentences. The outcome of this process is the initial production of features from local and global contexts.

$$
\begin{aligned}
O^l_{BERT} &= BERT^l(X^l) \\
O^g_{BERT} &= BERT^g(X^g)
\end{aligned}
\tag{1}
$$

In this instance, O^l_{BERT} and O^g_{BERT} symbolize the output representations for the processors of local context and global context, in that order. Furthermore, $BERT^l$ and $BERT^g$ signify the respective BERT pre-training models' roles in modeling the local and global contexts.

Feature extraction layer

Within the layer designated for feature extraction, a blend of the local context focus mechanism along with the conversational attention mechanism is chiefly applied to derive features of the local context. Conversely, the conversational attention mechanism alone is harnessed to extract features pertaining to the global context.
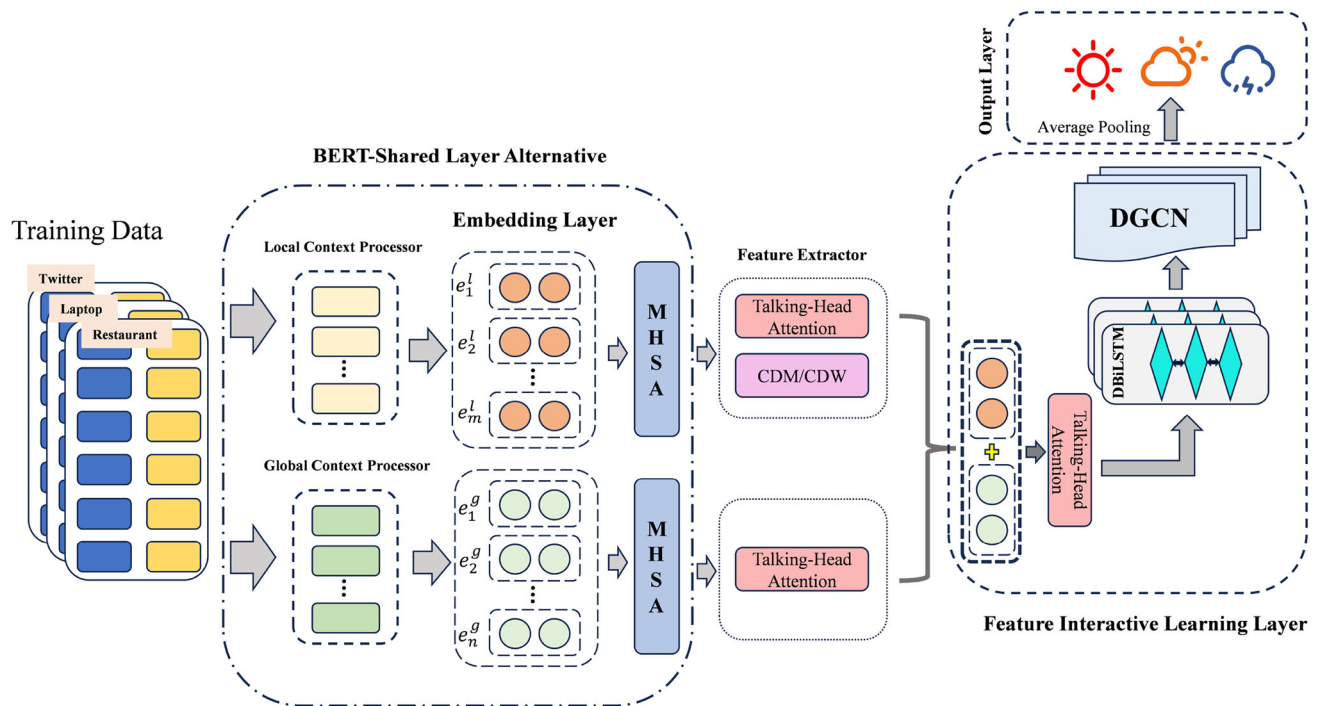
Fig. 1 Overall architecture of DCASAM design. BERT-shared Layer Alternative:Replaces the traditional BERT layer, handling both local and global context to enhance sentiment detection accuracy; Feature Extractor: Further extracts features using Talking-Head Attention and CDM/CDW modules; Feature Interactive Learning Layer: Integrates outputs from the feature extraction modules; Output Layer: Uses average pooling to aggregate features and produces the final sentiment classification results

Local context focus mechanism

Numerous preceding research efforts have segmented input sequences into sequences of aspects and contexts, with the objective of mapping the interplay between them. However, we ignore that there are often many local contexts of target aspects that contain more important information. Hence, ascertaining if a word from the context is relevant to the local context of a given aspect is of significant importance. Thus, this paper adopts Semantic Relative Distance (SRD) to support the model in seizing critical contextual information. To delve deeper into the information, this paper incorporates the conversational attention mechanism following the contextual feature mask layer.

(1) SRD (Semantic Relative Distance)

Semantic relative distance, as proposed by Kenett et al. [33], focuses on Token-Aspect pairs, indicating the locations of words within the sentence relative to the aspect word. It characterizes the distance between a Token and an Aspect, quantified by the count of intervening words. The formula is as follows:

$$D_i = |i - F_a| - \left\lceil \frac{n}{2} \right\rceil \tag{2}$$

In the given formula, i signifies the location of a particular word, and F_a corresponds to the position of the aspect word

within the sentence. n denotes the number of characters that make up the aspect word. Thus, D_i signifies the distance between the position of the ith word and the aspect in focus.

(2) Context-sensitive dynamic mask

Beyond the local context features, the context dynamic feature masking layer is designed to obscure the non-local context features that have been discerned by the $BERT^l$ layer. This selective masking serves to concentrate the model's attention on the local context by effectively nullifying the influence of non-local context features. The process of dynamic masking of context features entails converting the features at all positions identified as non-local contexts into zero vectors. If we take O^l_{BERT} to be the initial output feature from the $BERT^l$ layer, then we can derive the resulting local context feature, denoted as O^l_{CDM}, through the application of this masking process.

$$V_i = \begin{cases} E, & D_i \le a \\ O, & D_i > a \end{cases}$$

$$M = [V_1, V_2, \ldots, V_n] \tag{3}$$

$$O^l_{CDM} = O^l_{BERT} \bullet M$$

In this context, M represents the masking matrix that is applied to filter out non-local context features. The matrix M operates by selectively enabling or disabling certain fea-

tures based on their relevance to the aspect in question. The vector V_n is the masking vector that corresponds to each context word within the input sequence, effectively determining which features should be masked based on their semantic distance from the aspect. The variable a is the threshold for the Semantic Relative Distance (SRD), which is used to distinguish between local and non-local context. Finally, n indicates the total number of words in the input sequence, including the aspect word itself. When the Semantic Relative Distance (SRD) related to the target aspect is smaller than the threshold value a, this condition is indicative of a local context. In this formulation, E represents a vector of ones, and O symbolizes a zero vector, both of which have the dimensionality of n, the length of the input sequence including the aspect.

Local–global feature fusion

The integration of local and global features plays a crucial role in enhancing the model's performance in sentiment analysis tasks. This section details the collection and fusion of local and global features, as illustrated in Fig. 2.

Collection of Local Features: Each word embedding (such as Wi) interacts with the embeddings of its neighboring words (such $W2$, $W3$, and others) within a defined context window (solid lines) to capture fine-grained contextual information. These interactions are facilitated by the MHSA mechanism, enabling the model to capture detailed relationships between words.

Collection of Global Features: Positional encodings (such as $POS1$, $POS2$, $POSi$, and others) introduce positional information of each word in the sequence, aiding the model in capturing broader context. Through masking or weight ignoring (dashed lines), the model can adjust and filter positional encodings that contribute significantly to global features, thus effectively integrating global information. The collection of global features is also achieved via the MHSA mechanism, ensuring the capture of global context across the entire sequence.

During model training, local features (word embeddings) and global features (positional encodings) are combined through weighted integration using the defined context window (solid lines) and masking/weight ignoring (dashed lines). This weighted integration mechanism allows the model to dynamically adjust the weights of local and global information, thereby optimizing sentiment classification performance. The combination of local and global features provides the model with the following advantages in handling sentiment analysis tasks:

(1) Enhanced Fine-Grained Context Understanding: Local features offer an understanding of fine-grained context,

capturing local sentiment variations within sentences, and refining contextual comprehension.

(2) Broader Semantic Understanding: Global features provide an understanding of overall semantics, capturing document-level sentiment trends.

(3) Dynamic Weight Adjustment: The dynamic weighting mechanism allows the model to adjust the weights of local and global features in different contexts, thereby improving classification accuracy and robustness.

Talking-head attention

In the pioneering work by Vaswani et al. [34] $< Attention\ is\ All\ You\ Need >$, a modeling framework was proposed, wherein the traditional recurrent neural network was replaced by an attention mechanism. The results are computed by scaling the dot product multiple times and concatenating the results together. The outputs of the multiple attention heads can be obtained directly through a single linear transformation, as follows:

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$
$$Q^{(1)} = QW_Q^{(1)}, K^{(1)} = KW_K^{(1)}, V^{(1)} = VW_V^{(1)},$$
$$O^{(1)} = Attention(Q^{(1)}, K^{(1)}, V^{(1)})$$
$$Q^{(2)} = QW_Q^{(2)}, K^{(2)} = KW_K^{(2)}, V^{(2)} = VW_V^{(2)},$$
$$O^{(2)} = Attention(Q^{(2)}, K^{(2)}, V^{(2)})$$
$$\vdots$$
$$Q^{(h)} = QW_Q^{(h)}, K^{(h)} = KW_K^{(h)}, V^{(h)} = VW_V^{(h)},$$
$$O^{(h)} = Attention(Q^{(h)}, K^{(h)}, V^{(h)})$$
$$O_{MHA} = concat[\{O^{(1)}, O^{(2)}, \ldots, O^{(h)}\} \bullet W^{WH}]$$

$$(4)$$

Here, $W_Q^{(h)}$, $W_K^{(h)}$, and $W_V^{(h)}$ represent parameter matrices, where $W_Q \in \mathbb{R}^{d_1 \times d_Q}$, $W_K \in \mathbb{R}^{d_1 \times d_K}$, and $W_V \in \mathbb{R}^{d_1 \times d_V}$. $\sqrt{d_k}$ serves as a scaling factor, h denotes the number of attention heads, and W^{WH} is a weight matrix.

Researchers have observed that the model's performance plateaus when the number of attention heads for multi-headed attention reaches 12 or higher during the training process using Transformer. However, the performance of conversational attention continues to improve with an increase in the number of attention heads. This discrepancy arises due to the presence of a Low-Rank bottleneck in multi-headed attention [35]. A linear transformation is applied to the dimensions of attention heads within the conversational attention framework, both pre- and post-softmax, to bolster the interchange of information across various attention processes, thereby augmenting the model's efficacy. During this procedure, a matrix of parameters is utilized to amalgamate several atten-
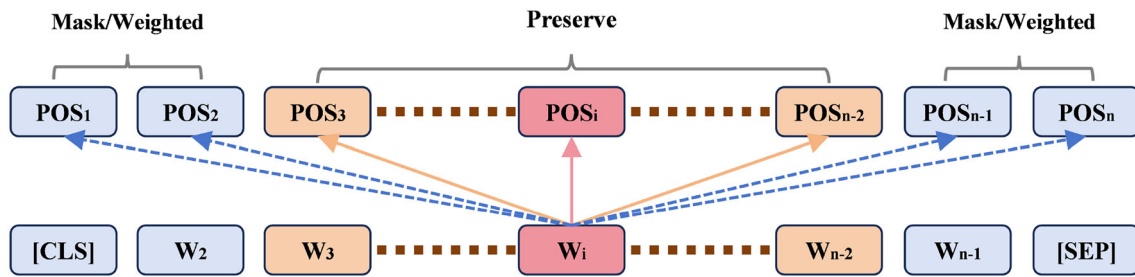
Fig. 2 Illustration of local and global feature fusion. [CLS] represents the classification token for the overall sequence representation. W_i is the current word vector, with W_2, W_3, W_{n-2}, and W_{n-1} as neighboring word vectors. Positional encodings POS_1, POS_2, POS_i, POS_{n-2}, POS_{n-1}, and POS_n indicate word positions. Mask/Weighted refers to the process of ignoring or assigning weights to certain positional encodings. Preserve denotes the context range influencing W_i. [SEP] marks the end of the sequence or separates different parts of it

tion heads, forming a collection of composite attentions. Each composite attention thus generated integrates attentions from the initial singular heads in the manner described below:

$$\hat{j}^{(1)} = \frac{Q^{(1)}K^{(1)T}}{\sqrt{d_k}}, \hat{j}^{(2)} = \frac{Q^{(2)}K^{(2)T}}{\sqrt{d_k}}, \ldots, \hat{j}^{(h)}$$
$$= \frac{Q^{(h)}K^{(h)T}}{\sqrt{d_k}}$$

$$\begin{bmatrix} J^{(1)} \\ J^{(1)} \\ \vdots \\ J^{(1)} \end{bmatrix} = \begin{bmatrix} \lambda_{11} & \lambda_{12} & \ldots & \lambda_{1h} \\ \lambda_{11} & \lambda_{12} & \ldots & \lambda_{1h} \\ \vdots & \vdots & \ddots & \vdots \\ \lambda_{h1} & \lambda_{h2} & \ldots & \lambda_{hh} \end{bmatrix} \begin{bmatrix} \hat{j}^{(1)} \\ \hat{j}^{(2)} \\ \vdots \hat{j}^{(h)} \end{bmatrix} \quad (5)$$

$$P^{(1)} = softmax(J^{(1)}),$$
$$P^{(2)} = softmax(J^{(2)}), \ldots, P^{(h)} = softmax(J^{(h)})$$
$$O^{(1)} = P^{(1)}V^{(1)}, O^{(2)} = P^{(2)}V^{(2)}, \ldots, O^{(h)} = P^{(h)}V^{(h)}$$
$$O_{THA} = concat[\{O^{(1)}, O^{(2)}, \ldots, O^{(h)}\} \bullet W^{WH}]$$

Here, $J^{(h)}$ signifies the linear transformation applied to individual attention heads prior to the execution of the softmax function. $O^{(h)}$ is indicative of the outcome produced by the attention mechanism, specifically $Attention(Q^{(h)}, K^{(h)}, V^{(h)})$. The term $\sqrt{d_k}$ functions as a normalizing constant, h corresponds to the count of attention heads, and W^{WH} represents a matrix of weights.

The THA encoder is utilized to acquire and equalize the obscured local context features, thereby addressing the disparity in feature distribution that results from the CDM process. Local context features, denoted as O^l, are derived by channeling the local features obtained from the CDM layer through the THA layer, which operates in tandem with the context focus mechanism.

$$O^l = THA(O^l_{CDM}) \quad (6)$$

For the global context feature O^g, the feature output is obtained directly from the THA encoder:

$$O^g = THA(O^g_{BERT}) \quad (7)$$

Feature learning layer

To improve the model's performance in complex natural language processing tasks, especially when handling text data with intricate dependencies and hierarchical structures, we introduce Deep Bidirectional Long Short-Term Memory Networks (DBiLSTM) and Densely Connected Graph Convolutional Networks (DGCN). This is aimed at improving the model's depth of understanding and its ability to handle context in textual data.

Deep bidirectional long and short-term memory networks

To better align with people's reading habits, a unidirectional LSTM considers only the preceding words, but in reality, a word is influenced not only by preceding words but also by subsequent ones. BiLSTM captures contextual information from both directions, effectively addressing the model's ability to process context. To further strengthen the model's capability to extract deep semantic information between aspect words and their context, we employ DBiLSTM, its structure is shown in Fig. 3.

The LSTM computation formula is represented as Eq. (8):

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$$
$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$$
$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C)$$
$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t \quad (8)$$
$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)$$
$$h_t = o_t * \tanh(C_t)$$

Here, i_t, f_t, and o_t symbolize the input gate, forget gate, and output gate, correspondingly. σ denotes the sigmoid activation function, and W and b are the weight and bias parameters. $\tilde{C}_t$ and C_t represent the temporary and final values of the memory cell at time t. h_t and h_{t-1} correspond to the outputs of the LSTM at time t and $t-1$, respectively.

BiLSTM builds upon the capabilities of LSTM by incorporating the ability to propagate information in both forward and backward directions. It comprises two LSTM layers, one handling forward information and the other handling backward information. The computation formula for BiLSTM [36] is represented as Eq. (9):

$$\begin{aligned}
\overrightarrow{h_t} &= LSTM(\overrightarrow{h_{t-1}}, x_t) \\
\overleftarrow{h_t} &= LSTM(\overleftarrow{h_{t-1}}, x_t) \\
h_t &= [\overrightarrow{h_t}, \overleftarrow{h_t}]
\end{aligned} \tag{9}$$

DBiLSTM consists of multilayer BiLSTM, which is calculated as shown in Eq. (10):

$$\begin{aligned}
\overrightarrow{h_t^n} &= LSTM(\overrightarrow{h_{t-1}^n}, \overrightarrow{h_t^{n-1}}) \\
\overleftarrow{h_t^n} &= LSTM(\overleftarrow{h_{t-1}^n}, \overleftarrow{h_t^{n-1}}) \\
h_t^n &= [\overrightarrow{h_t^n}, \overleftarrow{h_t^n}]
\end{aligned} \tag{10}$$

Here, $\overrightarrow{h_t^{n-1}}$ and $\overleftarrow{h_t^{n-1}}$ represent the outputs of the $\overrightarrow{LSTM}$ and $\overleftarrow{LSTM}$ at layer $n-1$ at time t, respectively. $\overrightarrow{h_{t-1}^n}$ and $\overleftarrow{h_{t-1}^n}$ represent the outputs of the $\overrightarrow{LSTM}$ and $\overleftarrow{LSTM}$ at layer l at time $t-1$, respectively. h_t^n is the output of the BiLSTM at layer n at time t.

Densely connected graph convolution networks

GCN is a neural network capable of directly operating on graph structures, and by adjusting the number of GCN layers, it can extract structural information from different neighborhoods. Assuming an L-layer GCN operates on a dependency graph $g = (v, \epsilon)$, where v and ϵ represent the node and edge sets, respectively, the computation for the output representation $h_i^{(k)}$ of node i at layer k is given by Eq. (11):

$$h_i^{(k)} = \rho \left(\sum_{j=1}^{q} A_{ij} W^{(k)} h_j^{(k-1)} + b^{(k)} \right) \tag{11}$$

$h_j^{(k-1)}$ represents the output representation of node j at the $(k-1)$-th layer of GCN, $W^{(k)}$ denotes the weight matrix, $b^{(k)}$ represents the bias vector, ρ is the non-linear activation function ReLU, and A_{ij} denotes the connection weight between node i and node j.

The Densely Connected Graph Convolutional Network (DGCN) enhances traditional GCN by incorporating dense connections. This enables each layer not only to receive outputs solely from the preceding layer but also to receive outputs from all layers positioned between the initial layer and the current layer. This structural design facilitates smoother information flow within the network, enabling each layer to fully utilize information from preceding layers, thereby enhancing feature extraction capability and efficiency. The determination of each dense connection is influenced by both the layer count L and the input feature dimension d, denoted as $d_{hidden} = d/L$. In the end, the output from each layer is concatenated to form a new representation with the same dimension as the input feature dimension d. The computation is represented by Eq. (12):

$$g_u^{(l)} = [X_u; h_u^{(1)}; \ldots; h_u^{(l-1)}] \tag{12}$$

In this context, X_u denotes the initial node representation, while $g_u^{(l)}$ represents the representation of node u at the l-th layer computed via dense connections.

With the introduction of dense connections, the connections between GCN layers become more intimate. This not only allows capturing rich local neighborhood information between nodes but also captures deep-level structural information between layers. As illustrated in the diagram, Fig. 4 is a 3-layer Densely Connected Graph Convolutional Network (DGCN). The solid black lines between layers represent the original connections in the graph convolutional network, while the dashed lines indicate the newly added dense connections. In the absence of dense connections, node representations are solely influenced by the network depth and the output of the preceding layer. The incorporation of dense connections strengthens the relationships between layers, integrating deeper structural information into node representations.

To prevent the neglect of node self-information, an identity matrix I is added to the original adjacency matrix A. Therefore, the original GCN computation formula needs to be modified, and the modified formula is represented as Eq. (13):

$$h_i^{(k)} = \rho \left(\sum_{j=1}^{q} c_i \hat{A}_{ij} W^{(k)} g_j^{(k)} + b^{(k)} \right) \tag{13}$$

Here, $W^{(k)} \in \mathbb{R}^{d_{\text{hidden}} \times d^{(k)}}$; $d^{(k)} = d + d_{\text{hidden}} \times (k-1)$, and $\hat{A}$ is the adjacency matrix considering the node self-loop, calculated as $\hat{A} = A + I$. To prevent a change in the original feature distribution when multiplying the unnormalized adjacency matrix $\hat{A}$ by the feature matrix $g^{(k)}$, a normalization constant c^i is introduced. Its value is $1/d$, where d_i represents the degree of node i, given by $\sum_{j=1}^{q} \hat{A}_{ij}$. Additionally, $g^{(0)}$

Fig. 3 DBiLSTM structural schematic. Red LSTM: Units representing the forward pass; Orange LSTM: Units representing the backward pass; Purple boxes: Denote the input features; Green ellipses: Represent the output results; Blue boxes: Indicate the hidden states; Dashed arrows: Signify the recurrent connections across time steps; Solid arrows: Indicate the data flow
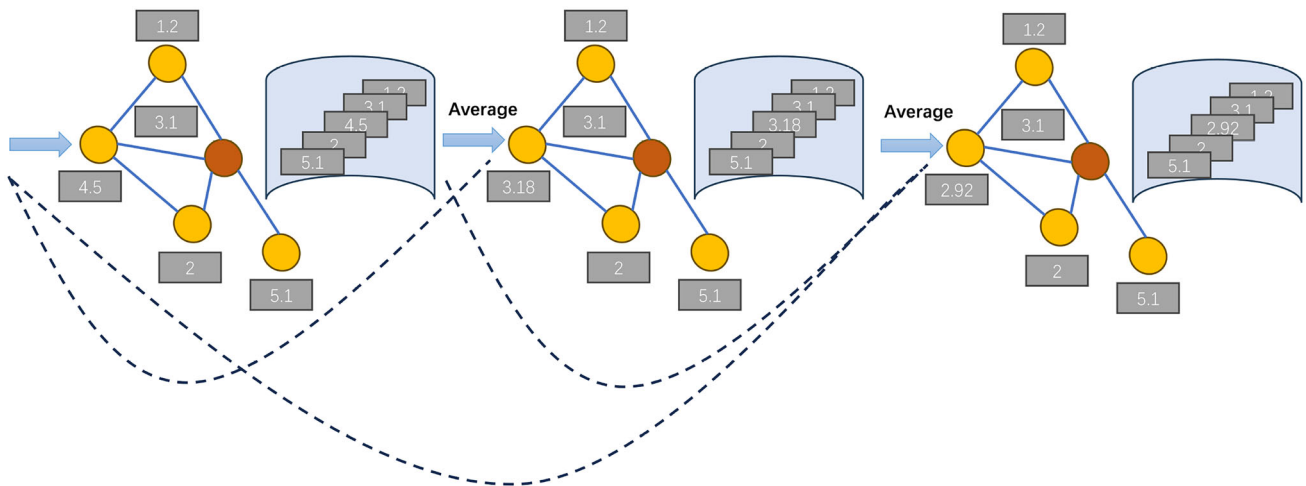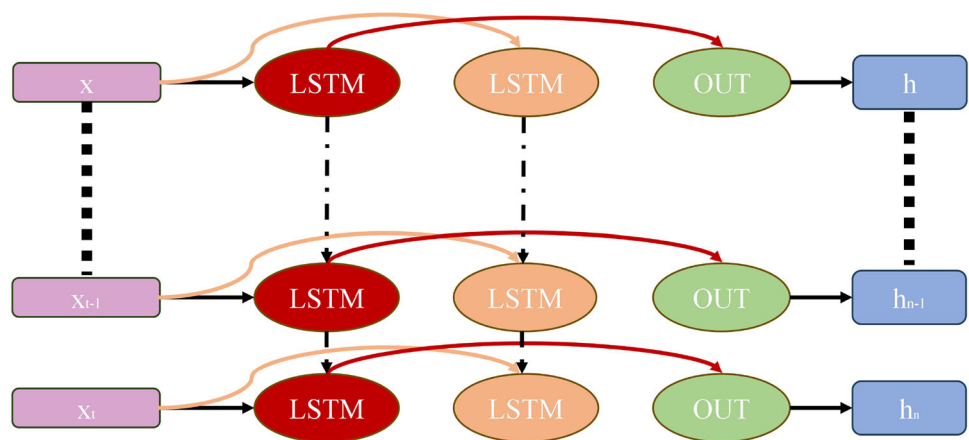
Fig. 4 DGCN structural schematic. Yellow nodes: Represent the initial nodes, where each node's feature vector is extracted from the original text using NLP embedding methods; Orange nodes: Indicate the target nodes being processed; Gray boxes: Denote the node feature values; Blue connecting lines: Represent the edges between nodes, indicating citation relationships; Feature aggregation process: Uses an averaging function, averaging the features of the nodes and their neighboring nodes to obtain new feature vectors; Black curved dashed lines: Represent densely connected, meaning direct connections from any layer to all subsequent layers, ensuring efficient information transfer between layers

takes the last layer output representation h_t^l of the DBiLSTM as the initial value.

Model training and output

In this paper, we employ the DCASAM model for sentiment analysis. The model training process includes the following steps Fig. 5.

First, we preprocess the data, including tokenization and generating input tensors. Next, we initialize the DCASAM model based on the pre-trained BERT model and its corresponding tokenizer. After initializing the model, we perform multiple iterations of training. In each iteration, we switch the model to training mode, batch by batch, to obtain training data, compute model outputs and loss, and update model

parameters through backpropagation. To prevent the model from overfitting, the loss function combines cross-entropy loss and regularization, which is formulated as follows:

$$L(\theta) = -\sum_{i=1}^{C} y_i \log \hat{y}_i + \lambda \sum_{\theta \in \Theta} \theta^2 \qquad (14)$$

where $L(\theta)$ denotes the loss function with parameters θ, $-\sum_{i=1}^{C}$ represents the sum over all categories C, y_i is the true sentiment value, $\log \hat{y}_i$ is the logarithm of the predicted probability, λ is the coefficient for the regularization term, $\sum_{\theta \in \Theta}$ represents the sum over the set of all parameters θ in Θ, and θ^2 denotes the square of the parameter, used for L2 regularization.

During training, we evaluate model performance on the validation set, record training loss and accuracy, and save

the model with the best performance. The goal of model training is to optimize model parameters by minimizing the loss function, thereby improving the model's accuracy and F1 score on the test set.

After training, we evaluate the final performance of the saved best model on the test set. Through this training process, we aim for the model to achieve high classification accuracy and robustness in practical applications. The figure below illustrates the entire model training process:

This training process ensures not only the performance of the model but also its generalization ability across different datasets.

To derive the final sentiment polarity of the aspect term, we apply the softmax function, formulated as follows:

$$\hat{y} = \text{softmax}(W_o O + b_o) \tag{15}$$

In this equation, W_o represents the weight matrix, b_o signifies the bias vector, and $\hat{y}$ denotes the predicted sentiment value.

Error analysis and contextual dependency in DCASAM

DCASAM (Deep Context-Aware Sentiment Analysis Model) is a sophisticated sentiment analysis model designed to address the complexities of sentiment detection. Its purpose is to accurately identify explicit and implicit sentiment expressions across various contexts, thus providing high-precision sentiment analysis. Figure 6 illustrates the detailed processing flow of DCASAM, demonstrating its capability to accurately determine sentiment polarity for both explicit text, such as "It's a beautiful day," and implicit text, like "This restaurant has an elegant ambience and a wide range of food."

First, DCASAM addresses the challenge of incomplete data inputs through data preprocessing and contextual inference. During the data preprocessing stage, rigorous data cleaning and filling in missing data ensure that the model utilizes as much useful information as possible. Contextual inference leverages pre-trained language models (such as BERT) to obtain context-related embeddings, enabling the model to understand and process text content even when some information is missing. This mechanism allows DCASAM to handle the inevitable issue of incomplete data inputs in real-world applications.

Second, DCASAM employs the Contextual Dependency Mechanism (CDM) to address the issue of category overlap. In sentiment expression, category overlap is common, where the same text may convey multiple sentiments. By considering the context, the CDM can adjust sentiment classification, reducing misclassification caused by category overlap, and improving classification accuracy. This strategy ensures that

the model can accurately identify primary and secondary sentiments when dealing with complex sentiment expressions.

Despite its advanced design, DCASAM occasionally encounters challenges when dealing with ambiguous sentiments. For instance, the phrase "That's funny" may be misclassified without contextual information, whereas the sentiment polarity can be accurately determined when context is provided. This indicates that DCASAM relies significantly on contextual cues to correctly interpret ambiguous emotions.

Overall, DCASAM excels in accurately processing a wide range of texts, especially when sufficient context is provided. Its high accuracy in sentiment analysis highlights its effectiveness in understanding and interpreting complex sentiments, making it a robust tool for sentiment detection across various contexts.

Experiments

Dataset description and preprocessing

In this study, we utilized the restaurant14, laptop14, and Twitter datasets from SemEval-2014 Task 4 [37] for aspect-level sentiment analysis across different domains. Within the annotations of these three publicly available datasets, each aspect term in every sentence corresponds to three distinct sentiment polarities, as outlined in Table 1. These datasets are derived from real user reviews and meticulously annotated to ensure accurate classification of each aspect and its corresponding sentiment polarity. The restaurant dataset comprises 3608 training reviews and 1120 testing reviews, primarily sourced from social media and restaurant review websites, covering customer evaluations of restaurant services, food, and environment. The laptop14 dataset includes 2328 training reviews and 638 testing reviews, sourced from e-commerce sites and tech forums, reflecting users' experiences and opinions on various aspects of laptops. The Twitter dataset [38] was collected using keywords (such as "bill gates," "taylor swift," "xbox," "windows 7," "google") to query the Twitter API, manually annotated with sentiment labels, and balanced through random sampling, containing 6248 training tweets and 692 testing tweets.

These datasets exhibit significant diversity: domain diversity requires the model to adapt to different vocabularies and expressions, and sentiment polarity diversity increases the complexity of classification, especially with the presence of conflicting sentiments where reviewers hold both positive and negative views on a given aspect. Additionally, each review may contain multiple aspects, with aspects and their corresponding sentiment polarity annotated at the sentence level, requiring the model to perform fine-grained sentiment analysis.
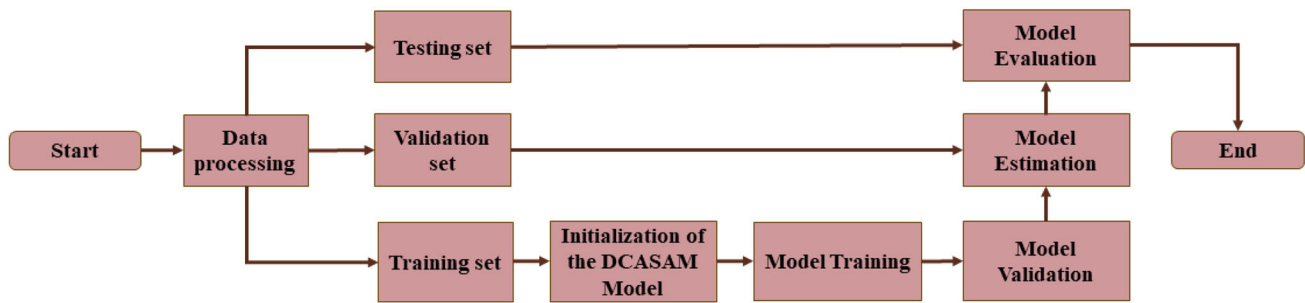
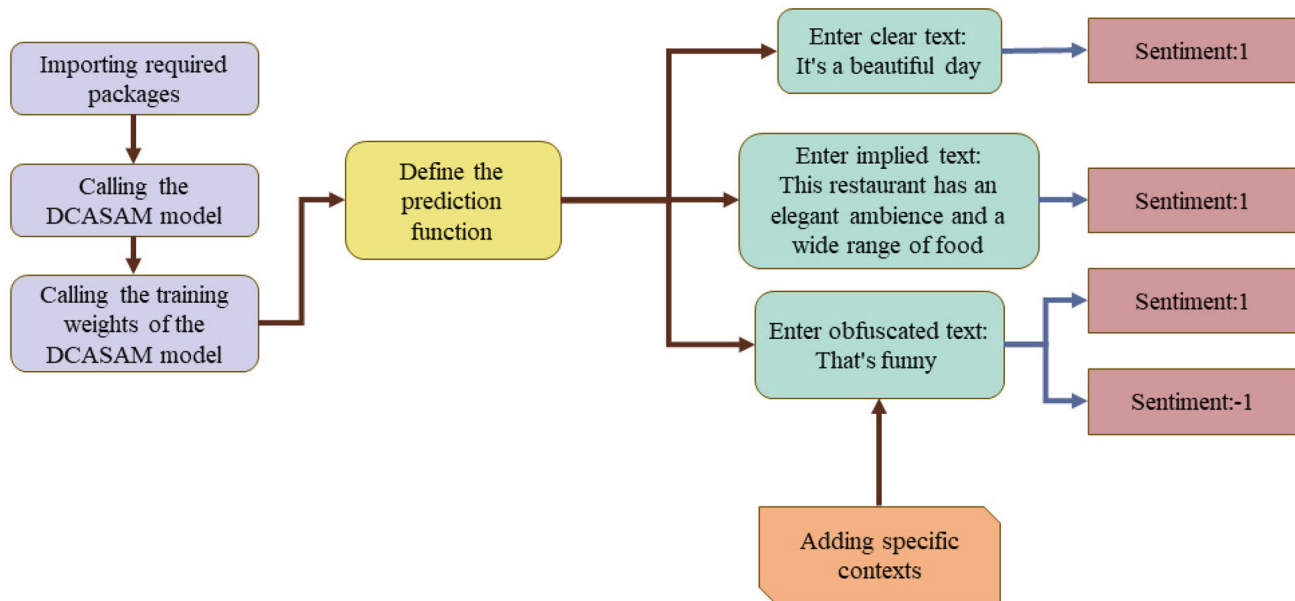**Fig. 5** DCASAM Model training flowchart

Fig. 6 DCASAM sentiment classification workflow

The input data for our DCASAM model consists of text data with associated sentiment labels. Each input sample includes a text segment, such as a sentence or a document, and a corresponding sentiment label indicating the sentiment expressed in the text (e.g., positive, negative, neutral). The output data from the model comprises predicted sentiment scores, which reflect the likelihood of each sentiment category for the given text input.

The data preprocessing steps involved several stages: first, loading the training and testing sets. For BERT models, specific tokenization tools were used; for other models, tokenizers and embedding matrices were generated. Next, the training set was split into training and validation sets based on a specified ratio; if the validation ratio was zero, the testing set was used as the validation set. The input text data was then standardized by removing noise, tokenizing, and stemming to reduce the vocabulary size and enhance the model's generalization capability. Subsequently, data loaders for the training, validation, and testing sets were created, with batch sizes and data shuffling parameters set accordingly. For non-

BERT models, embedding matrices were constructed and the model's embedding layer was initialized, while pre-trained model weights were loaded for BERT models. Finally, to ensure reproducibility of the experimental results, random seeds were set for Python, NumPy, and PyTorch, and deterministic algorithms were used in CuDNN.

In summary, the restaurant, laptop14, and Twitter datasets, with their diversity and biases in domains, sentiment polarity, and review structures, provide comprehensive evaluation and training data for sentiment analysis models, showcasing the models' adaptability and robustness in real-world application scenarios.

Baseline methods

To demonstrate the superior performance of our model compared to others, we applied seven different methods, comprising our proposed approach and six baseline methods, across three datasets:

Table 1 Databases used in experiments

Data	Restaurant14		Laptop14		Twitter	
	Train	Test	Train	Test	Train	Test
Positive	2164	728	994	341	1561	173
Neutral	637	196	464	169	3127	346
Negative	807	196	870	128	1560	173
Total	3608	1120	2328	638	6248	692

Table 2 Experimental setup

Environment	Configuration
Operating system	Windows 10 Professional 64-bit
CPU	ES-2680 V4
Memory	30GB RAM
GPU	NVIDIA RTX A4000
Framework	Pytorch 1.7.0

- TD-LSTM: The model was introduced by Wang et al. [17] in 2016, utilizing bidirectional context to represent specific aspects while fully incorporating the contextual information of a target word. Two bidirectional Long Short-Term Memory (LSTM) networks are employed to capture the information relevant to the target word. A non-linear layer receives the concatenated hidden states derived from the pair of LSTMs, serving as input for the analysis of sentiment.
- RAM: The model, introduced by Chen et al. [39] in 2017, tackles the challenge of distant dependencies in sentiment features by utilizing multiple attention mechanisms to capture them. It predicts sentiment polarity and demonstrates enhanced robustness to irrelevant information.
- MGAN: In 2018, the study by Fan et al. [40] presented a model that utilizes a detailed attention mechanism for grasping the interactive data shared between aspect terms and their context. The detailed attention is then amalgamated with a broader attention approach, constructing a multi-level attention network structure that concludes the sentiment polarity.
- AEN-BERT: In 2019, Song et al. [41] put forth a model that features an encoder based on attention to delineate aspect words and context, addressing the challenge of non-parallelizability inherent in Recurrent Neural Networks (RNNs). Additionally, label regularization is introduced in the loss function. Finally, classification is performed through a non-linear layer.
- SPC-BERT: In 2019, Song et al. [41] introduced a model that utilizes BERT pre-trained models with sentence pairs as input, followed by a non-linear layer for sentiment classification. This approach has demonstrated promising results.
- LCF-BERT: In 2019, Zeng et al. [42] proposed a model based on the fundamental idea of emphasizing the importance of words near aspect terms. This is achieved by incorporating a self-attention mechanism, and the output is then fed into a non-linear layer for sentiment classification tasks.
- DREGCN-BERT: In 2020, Liang et al. [43] introduced a model employing an end-to-end Aspect-Based Sentiment Analysis (ABSA) multitask learning with enhanced

Table 3 DCASAM parameter configuration

Parameters	Value
Optimizer	Adam
Learning rate	0.00002
Training iterations	100
Batch size	64
Maximum sentence length	85
Attention heads	12
DBiLSTM layers	2
L2 regularization weight	0.01
BERT model	Bert-base-uncased

dependency syntactic knowledge in an interactive structure. This architecture harnesses graph convolutional networks to fully leverage syntactic knowledge.
- DDGCN: In 2022, Sun et al. [44] proposed a model that utilizes two Graph Convolutional Networks to capture two types of structural information at different time stages. Such an approach permits the nuanced acquisition of dynamic event representations and their progressive consolidation, which captures the sequential impact, consequently refining the efficacy of language detection.
- FaiMA: In 2024, Yang et al. [45] introduces a ground-breaking framework named Feature-aware In-context Learning for Multi-domain Aspect-Based Sentiment Analysis (FaiMA). This innovative approach leverages In-context Learning (ICL) as a mechanism sensitive to specific features, aiming to enhance the adaptability of learning processes across various domains within ABSA tasks.

Experimental setup and parameter configuration

The experimental setup was configured within the PyTorch deep learning framework, and the specific configurations are detailed in Table 2.

The summarized configurations of selected hyperparameters for the DCASAM model are outlined in Table 3.

Experimental evaluation indicators

For the assessment of the model's performance in this experiment, two prevalent and broadly used evaluation metrics for aspect-level sentiment analysis tasks are employed. The metric of accuracy, which gauges the ratio of accurately predicted samples against the total number of samples, taking into account predictions of both positive and negative nature, is computed in the following manner:

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} \qquad (16)$$

Here, TP represents the count of positive samples correctly predicted, TN represents the count of negative samples correctly predicted, FP represents the count of positive samples incorrectly predicted, and FN represents the count of negative samples incorrectly predicted.

The second evaluation metric, MF1, offers a representation of the average performance of the model across various emotion categories.

$$\begin{cases} P = \frac{TP}{TP+FP} \\ R = \frac{TP}{TP+FN} \\ F_1 = \frac{1}{|\mathcal{C}|} \sum_i^{|\mathcal{C}|} \frac{2 \cdot P \cdot R}{P+R} \end{cases} \qquad (17)$$

Here, P is indicative of the precision rate corresponding to the sentiment category, R is indicative of the recall rate for the sentiment category, and C totals the number of distinct sentiment categories.

Experimental results and analysis

Experimental comparisons were carried out across three publicly available datasets: Restaurant14, Laptop14, and Twitter3, utilizing a total of 8 baseline models. The results are presented in Table 4.

Analyzing the experimental outcomes presented in Table 4, it's clear that across the three public datasets Restaurant14, Laptop14, and Twitter, the proposed DCASAM model demonstrates superior performance compared to baseline models in both accuracy and F1 score. The accuracy improvement is 1.34%, 0.63%, and 1.25%, and the F1 score improvement is 2.83%, 0.63%, and 1.59%, respectively. The ANE-BERT, BERT-SPC, and LCF-BERT models were replicated based on the methods outlined in their respective original papers. Performance across all models on the Restaurant14 and Laptop14 datasets exceeds that on the Twitter dataset. The variance in performance is attributed to the nature of the datasets. Reviewers in the Restaurant14 and Laptop14 datasets tend to use simpler grammar in their reviews. However, the Twitter dataset contains entities cor-

responding to aspect terms that vary significantly, along with the prevalence of internet slang and sarcastic statements, leading to lower accuracy in classification. Moreover, in the first two public datasets, there is a higher proportion of positive sentiment polarity, leading to favorable sentiment classification results in all models.

Among the eight baseline models mentioned above, the TD-LSTM model demonstrates relatively inferior experimental performance. This can be attributed to the TD-LSTM model's failure to consider the semantic connections between aspect terms and context words. Moreover, the model lacks an attention mechanism, leading to its incapacity to ascertain the significance of words in a sentence in relation to the aspect. The overall performance of both the RAM and MGAN models is also less than satisfactory. This limitation stems from the GloVe word embeddings' approach of depicting a word with a solitary vector, which does not account for the polysemy of words across diverse contexts, consequently affecting the models' ultimate categorization outcomes. Relative to other aspect-level sentiment classification models that utilize BERT pre-trained architectures, the introduced DCASAM framework demonstrates enhanced performance. The BERT-SPC model, which employs a sentence pair structure, leverages BERT's pre-training capabilities to execute sentiment analysis on contextual words and aspect terms within a sentence. The ANE-BERT model aids in modeling aspect terms and context effectively. Nevertheless, these models do not account for the interplay between sentiment polarity and the immediate textual environment. In contrast, our model embeds a mechanism that zeroes in on the local context, deeply appreciating the relationship between sentiment polarity and its adjacent context. LCF-BERT primarily focuses on the importance of words near the aspect terms and then employs a non-linear layer for the task of sentiment classification. However, the multi-head attention mechanism implemented in the model presents a bottleneck problem. To address this, our model introduces a conversational attention mechanism. Specifically, linking independent attention heads produces a more robust attention design, thereby enhancing model performance. DREGCN-BERT utilizes a graph neural network on a syntactic relationship graph and enhances classification performance using a BERT pre-trained model. However, the results obtained are still to be improved compared to our model. The empirical results support the notion that our proposed framework for aspect-level sentiment analysis, based on the local context focus mechanism and dialogic attention, surpasses others in performance for tasks of aspect-level sentiment analysis.

Figures 7, 8 and 9 illustrate the training results of our proposed DCASAM model. The blue lines represent the training set results, while the orange lines represent the test set results. As observed, our model achieved significant performance after only 10 iterations, and it maintained high accuracy

Table 4 Databases used in experiments

Model	Restaurant14		Laptop14		Twitter	
	Accuracy	F1	Accuracy	F1	Accuracy	F1
TD-LSTM	77.80	64.84	72.12	68.43	70.98	68.64
RAM	80.32	70.81	74.48	71.35	69.37	67.32
MGAN	81.25	71.94	75.40	72.50	72.55	70.83
AEN-BERT	82.05	73.76	79.93	76.31	73.71	72.13
BERT-SPC	83.85	72.34	78.99	75.03	73.55	72.16
LCF-BERT	85.36	78.36	79.47	75.83	74.42	73.19
DREGCN-BERT	83.61	75.79	79.37	76.37	74.30	73.10
DDGCN	83.11	76.26	77.78	73.76	75.63	73.66
FaiMA	85.40	81.39	78.12	70.58	74.41	68.45
DCASAM	86.70	81.19	80.56	77.00	76.88	75.25
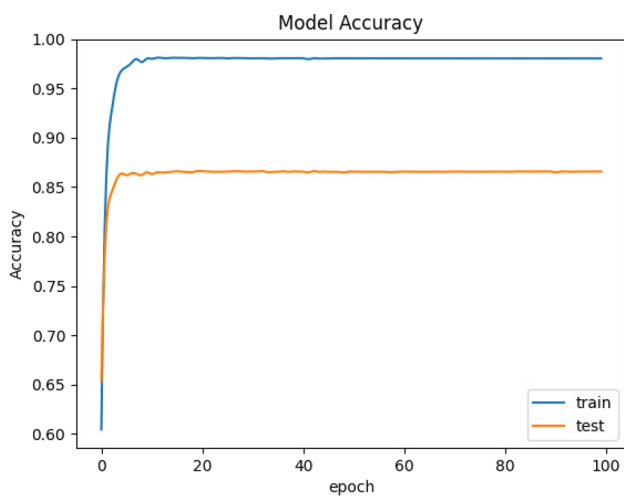
Fig. 7 Model accuracy comparison
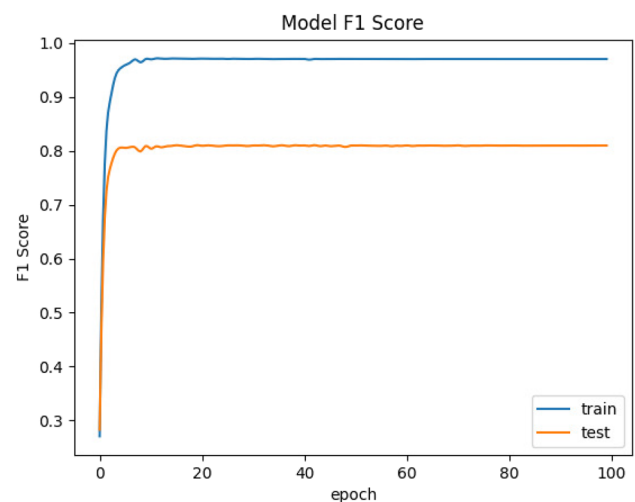
Fig. 8 Model F1 score comparison

and F1 scores throughout further training. Additionally, the model's loss rate was exceptionally low, remaining around 1%, demonstrating the excellent learning capability and stability of the DCASAM model.

Parameter selection experiment

Parameter experiment for SRD

Within the DCASAM framework, the semantic proximal distance plays a pivotal role in the extraction of local contextual features. Consequently, this study carries out an experimental analysis to investigate the influence of semantic proximal distance on the precision rate within three distinct datasets. As depicted in Figs. 10 and 11, A semantic proximal distance threshold set to 4 results in the best accuracy and F1 scores for the Restaurant14 dataset, with the highest accuracy recorded at 86.7% and the F1 score at 81.19%. Regarding the Laptop14 dataset, a threshold of 5 for semantic proxi-
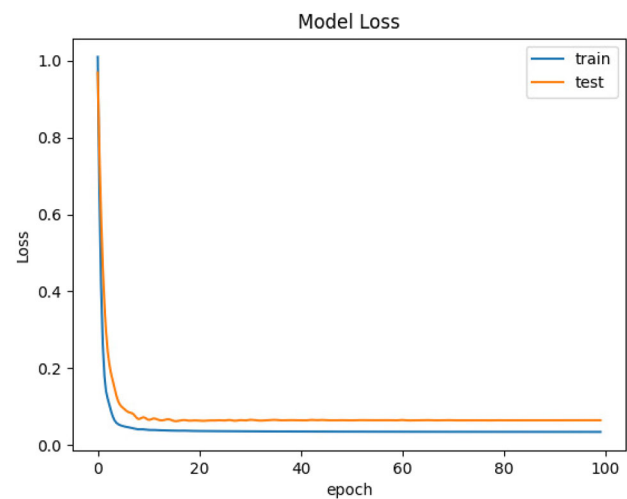
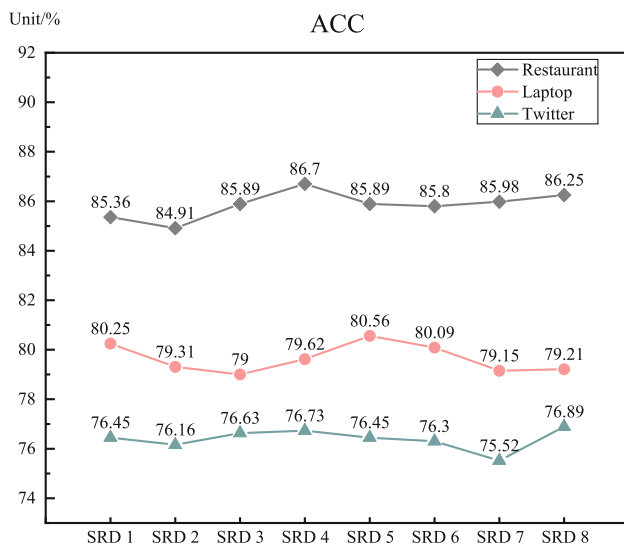**Fig. 9** Model loss comparison

ACC

Fig. 10 Accuracy of the model on the Restaurant14, Laptop14, Twitter dataset under different SRD thresholds
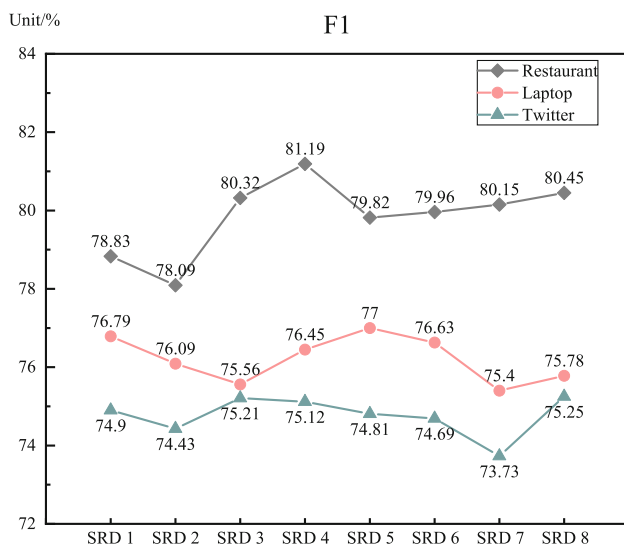
F1

Fig. 11 F1 of the model on the Restaurant14, Laptop14, Twitter dataset under different SRD thresholds

mal distance results in the most favorable accuracy and F1 scores, achieving 80.56% and 77% respectively. For the Twitter dataset, setting the semantic proximal distance threshold to 8 enhances both accuracy and F1 values, attaining 76.88% in accuracy and 75.25% in F1 score.

Determination of L2 regularization parameter (lambda)

To determine the optimal L2 regularization parameter (lambda), we conducted experiments on three different datasets: restaurant14, laptop14, and Twitter. The purpose of these experiments was to analyze how different lambda values affect the model's performance in terms of accuracy (ACC) and F1 scores.

Table 5 Ablation experiment results

Data	Restaurant14 Accuracy/F1	Laptop14 Accuracy/F1	Twitter Accuracy/F1
w/o DB	85.36/78.36	79.47/75.83	74.42/73.19
w/o DC	85.00/79.13	81.35/77.68	74.71/72.74
Ours	86.70/81.19	80.56/77.00	76.88/75.25

We systematically adjusted the lambda values and trained and evaluated the model for each value. The range of lambda values tested included [0.0001, 0.001, 0.01, 0.1], ensuring a comprehensive understanding of their impact.

Figures 12, 13 and 14 illustrate the impact of different lambda values on the model's performance metrics (accuracy and F1 scores). The x-axis represents different models, the y-axis represents performance metrics (accuracy and F1 scores), and the different colors of the bar charts represent different lambda values.

The experimental results on the three datasets (restaurant14, laptop14, and Twitter) show that when the L2 regularization parameter (lambda) is close to 0.01, the model achieves the highest accuracy (ACC) and F1 scores. Although the optimal lambda value may vary slightly across different datasets, the overall trend is consistent, with the best performance observed at a lambda value of 0.01. Therefore, we chose 0.01 as the optimal L2 regularization parameter to achieve the best model performance.

Ablation experiment

This study carries out ablation testing on three widely accessible datasets to ascertain the significance of each component within the DCASAM framework. In this context, "w/o" stands for "without." Specifically, "w/o DB" indicates that the DBiLSTM component is omitted from the model while the remaining parts are intact, and "w/o DC" signifies that the DGCN component is excluded with the rest of the model remaining unaltered. The outcomes of these configurations are presented in Table 5.

Table 5 shows that the combination of the DBiLSTM Network and the DGCN Network is better than the single module, which indicates that DBiLSTM is helpful for mining deep semantic information in text and the introduction of tight connection makes GCN able to mine deep structural information, which makes up for the lack of the original only extracting the neighborhood structural information. In summary, the use of DBiLSTM and tight connections is helpful for mining deep information and improving the classification effect.

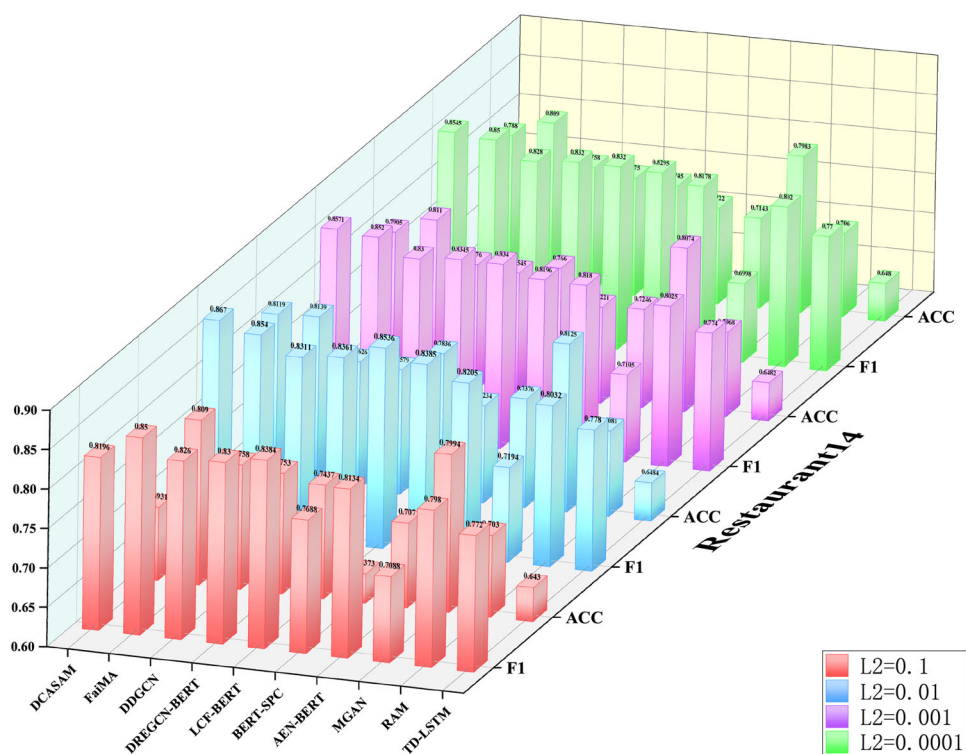**Fig. 12** Performance of different L2 parameters on the Restaurant14 dataset

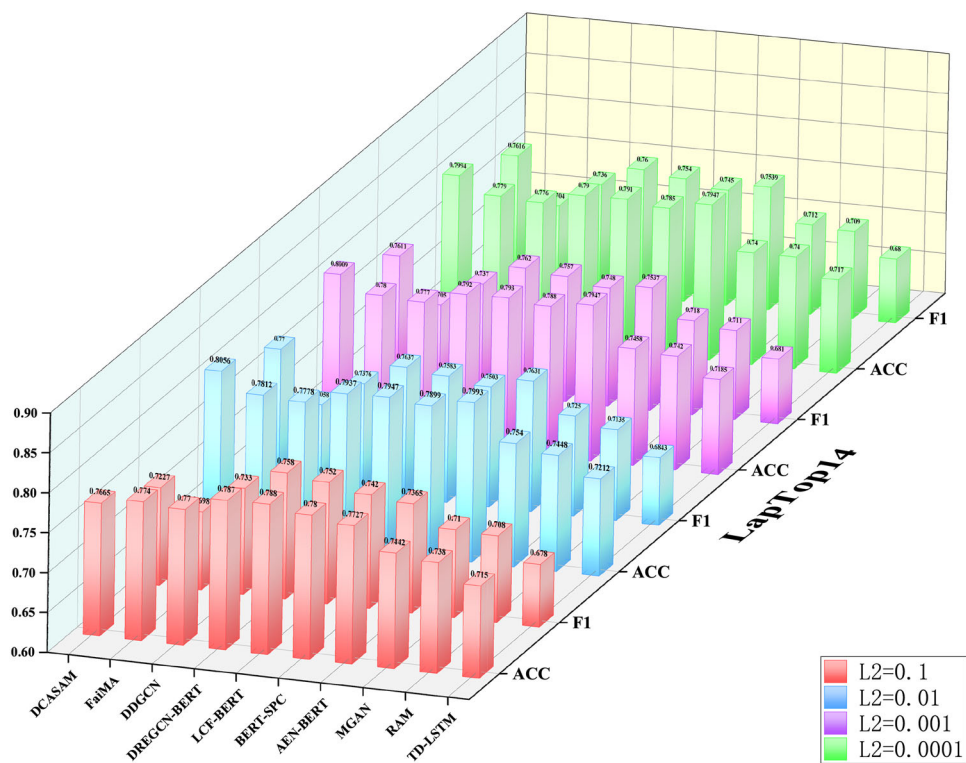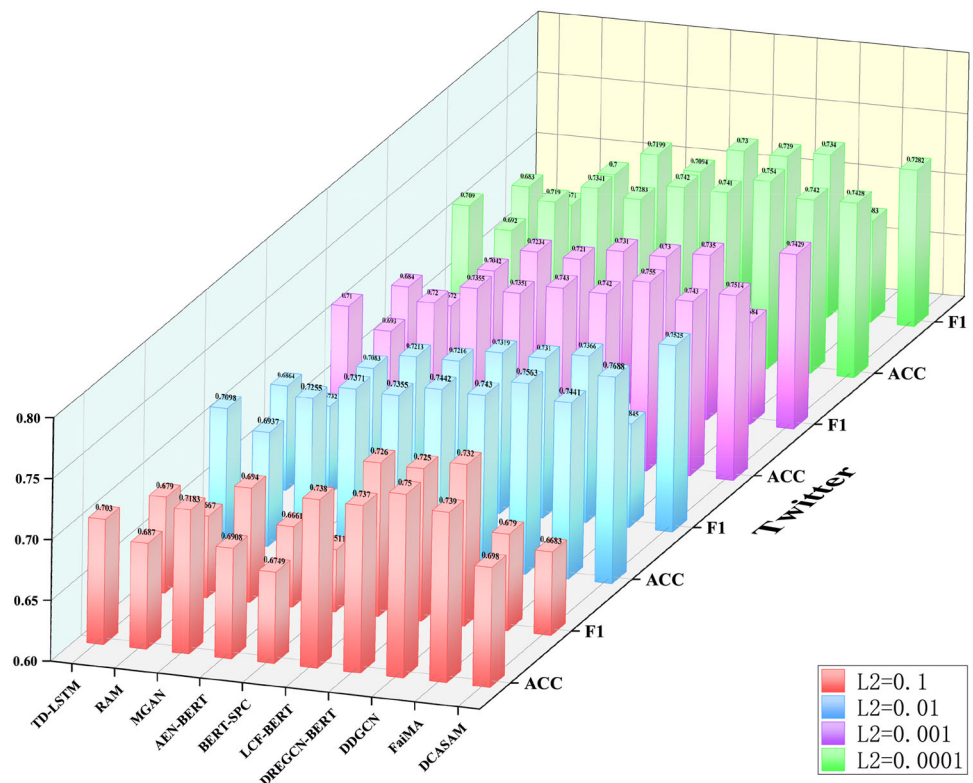**Fig. 13** Performance of different L2 parameters on the LapTop14 dataset

Fig. 14 Performance of different L2 parameters on the Twitter dataset

Concluding remarks

To enhance aspect-level sentiment classification, this manuscript introduces the DCASAM model. Initially, dual BERT pre-trained models are employed to separately distill local and global contextual features. Following that, the CDM layer, which is a part of the local context focus mechanism, works together with the conversational attention mechanism to enhance the local features extracted by BERT. A conversational attention encoder is then utilized to assimilate global contextual features. The local and global features are combined within the feature learning layer and then fed into the DBiLSTM layer, which is designed to probe into more profound semantic interconnections among the contexts. The DBiLSTM's final layer output serves as the input for the compact GCN, which extracts neighborhood and intricate structural data from the dependency tree. The process concludes with the sentiment being classified via the Softmax layer. The model's effectiveness is demonstrated by enhanced accuracy and F1 scores when compared to other models, a fact that is substantiated by experimental analyses conducted on three public datasets. Ablation studies additionally verify the significance of each component within the model.

Considering its advanced design and capabilities, DCASAM could potentially be extended to other domains such as customer reviews or political discourse analysis. These areas often involve complex and nuanced sentiments that require

sophisticated analysis. The ability of DCASAM to leverage contextual cues and adjust for category overlap suggests that it could effectively handle the unique challenges presented by these domains. Future work could explore the application of DCASAM in these fields, further validating its versatility and robustness.

Looking ahead, future research will delve into aspect-based sentiment analysis on Chinese datasets and datasets that have been self-collected, aiming to leverage these findings in practical applications.

Data availability The datasets used and/or analysed during the current study available from the corresponding author on reasonable request.

Declarations

Conflict of interest The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

1. Chvanova MS, Khramova MV, Pitsik EN (2017) Investigation of internet influence on users social needs. In: 2017 International conference "Quality management,transport and information security, information technologies" (IT&QM&IS), pp 652–657. https://doi.org/10.1109/ITMQIS.2017.8085908

2. Nazir A, Rao Y, Wu L, Sun L (2020) Issues and challenges of aspect-based sentiment analysis: a comprehensive survey. IEEE Trans Affect Comput 13(2):845–863

3. Wankhade M, Rao ACS, Kulkarni C (2022) A survey on sentiment analysis methods, applications, and challenges. Artif Intell Rev 55(7):5731–5780

4. Taboada M, Brooke J, Tofiloski M, Voll K, Stede M (2011) Lexicon-based methods for sentiment analysis. Comput Linguist 37(2):267–307

5. Al-Ghuribi SM, Noah SAM, Tiun S (2020) Unsupervised semantic approach of aspect-based sentiment analysis for large-scale user reviews. IEEE Access 8:218592–218613

6. Tian Y, Chen G, Song Y (2021) Aspect-based sentiment analysis with type-aware graph convolutional networks and layer ensemble. In: Proceedings of the 2021 conference of the North American chapter of the association for computational linguistics. Human Language Technologies, pp 2910–2922

7. Prottasha NJ, Sami AA, Kowsher M, Murad SA, Bairagi AK, Masud M, Baz M (2022) Transfer learning for sentiment analysis using BERT based supervised fine-tuning. Sensors 22(11):4157

8. Koroteev M (2021) BERT: a review of applications in natural language processing and understanding. arXiv preprint arXiv:2103.11943

9. Sun S, Luo C, Chen J (2017) A review of natural language processing techniques for opinion mining systems. Inf Fusion 36:10–25

10. Shazeer N, Lan Z, Cheng Y, Ding N, Hou L (2020) Talking-heads attention. arXiv preprint arXiv:2003.02436

11. Zeng B, Yang H, Xu R, Zhou W, Han X (2019) LCF: a local context focus mechanism for aspect-based sentiment classification. Appl Sci 9(16):3389

12. Kaur A, Gupta V (2013) A survey on sentiment analysis and opinion mining techniques. J Emerg Technol Web Intell 5(4):367–371

13. Wang J, Xu B, Zu Y (2021) Deep learning for aspect-based sentiment analysis. In: 2021 International conference on machine learning and intelligent systems engineering (MLISE). IEEE, pp 267–271

14. Brinati D, Campagner A, Ferrari D, Locatelli M, Banfi G, Cabitza F (2020) Detection of COVID-19 infection from routine blood exams with machine learning: a feasibility study. J Med Syst 44:1–12

15. Tutsoy O, Koç GG (2024) Deep self-supervised machine learning algorithms with a novel feature elimination and selection approaches for blood test-based multi-dimensional health risks classification. BMC Bioinform 25(1):103

16. Yan Q, Niu A, Wang C, Dong W, Woźniak M, Zhang Y (2024) KGSR: a kernel guided network for real-world blind super-resolution. Pattern Recognit 147:110095

17. Wang Y, Huang M, Zhu X, Zhao L (2016) Attention-based LSTM for aspect-level sentiment classification. In: Proceedings of the 2016 conference on empirical methods in natural language processing, pp 606–615

18. Woźniak M, Wieczorek M, Siłka J (2023) BiLSTM deep neural network model for imbalanced medical data of IoT systems. Future Gen Comput Syst 141:489–499

19. Zhang Q, Lu R, Wang Q, Zhu Z, Liu P (2019) Interactive multi-head attention networks for aspect-level sentiment classification. IEEE Access 7:160017–160028

20. Jing Y, Si C, Wang J, Wang W, Wang L, Tan T (2020) Pose-guided multi-granularity attention network for text-based person search. In: Proceedings of the AAAI conference on artificial intelligence, vol 34, pp 11189–11196

21. Chen W, Feng F, Wang Q, He X, Song C, Ling G, Zhang Y (2021) CATGCN: graph convolutional networks with categorical node features. IEEE Trans Knowl Data Eng 35(4):3500–3511

22. Phan HT, Nguyen NT, Hwang D (2022) Aspect-level sentiment analysis using CNN over BERT-GCN. IEEE Access 10:110402–110409

23. Xu K, Zhao H, Liu T (2020) Aspect-specific heterogeneous graph convolutional network for aspect-based sentiment classification. IEEE Access 8:139346–139355

24. Cai H, Tu Y, Zhou X, Yu J, Xia R (2020) Aspect-category based sentiment analysis with hierarchical graph convolutional network. In: Proceedings of the 28th international conference on computational linguistics, pp 833–843

25. Ke Q, Jing X, Woźniak M, Xu S, Liang Y, Zheng J (2024) APGVAE: adaptive disentangled representation learning with the graph-based structure information. Inf Sci 657:119903

26. Chen J, Hou H, Ji Y, Gao J (2019) Graph convolutional networks with structural attention model for aspect based sentiment analysis. In: 2019 International joint conference on neural networks (IJCNN). IEEE, pp 1–7

27. Tutsoy O, Barkana DE, Tugal H (2018) Design of a completely model free adaptive control in the presence of parametric, non-parametric uncertainties and random control signal delay. ISA Trans 76:67–77

28. Zhang Q, Lu R (2019) A multi-attention network for aspect-level sentiment analysis. Future Internet 11(7):157

29. Hu Z, Wang Z, Wang Y, Tan A-H (2023) MSRL-Net: A multi-level semantic relation-enhanced learning network for aspect-based sentiment analysis. Expert Syst Appl 217:119492

30. Kovaleva O, Romanov A, Rogers A, Rumshisky A (2019) Revealing the dark secrets of BERT. arXiv preprint arXiv:1908.08593

31. Bello A, Ng S-C, Leung M-F (2023) A BERT framework to sentiment analysis of tweets. Sensors 23(1):506

32. Wang H, Li J, Wu H, Hovy E, Sun Y (2022) Pre-trained language models and their applications. Engineering

33. Kenett YN, Levi E, Anaki D, Faust M (2017) The semantic distance task: quantifying semantic distance with semantic network path length. J Exp Psychol Learn Mem Cognit 43(9):1470

34. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser Ł, Polosukhin I (2017) Attention is all you need. Adva Neural Inf Process Syst 30

35. Bhojanapalli S, Yun C, Rawat AS, Reddi S, Kumar S (2020) Low-rank bottleneck in multi-head attention models. In: International conference on machine learning. PMLR, pp 864–873

36. Xu G, Meng Y, Qiu X, Yu Z, Wu X (2019) Sentiment analysis of comment texts based on BILSTM. IEEE Access 7:51522–51532

37. Kirange D, Deshmukh RR, Kirange M (2014) Aspect based sentiment analysis SemEval-2014 task 4. Asian J Comput Sci Inf Technol (AJCSIT) 4

38. Sahayak V, Shete V, Pathan A (2015) Sentiment analysis on Twitter data. Int J Innov Res Adv Eng (IJIRAE) 2(1):178–183

39. Chen P, Sun Z, Bing L, Yang W (2017) Recurrent attention network on memory for aspect sentiment analysis. In: Proceedings of the 2017 conference on empirical methods in natural language processing, pp 452–461

40. Fan F, Feng Y, Zhao D (2018) Multi-grained attention network for aspect-level sentiment classification. In: Proceedings of the 2018 conference on empirical methods in natural language processing, pp 3433–3442

41. Song Y, Wang J, Jiang T, Liu Z, Rao Y (2019) Attentional encoder network for targeted sentiment classification. arXiv preprint arXiv:1902.09314

42. Zeng B, Yang H, Xu R, Zhou W, Han X (2019) LCF: a local context focus mechanism for aspect-based sentiment classification. Appl Sci 9(16):3389

43. Liang Y, Meng F, Zhang J, Chen Y, Xu J, Zhou J (2021) A dependency syntactic knowledge augmented interactive architecture for end-to-end aspect-based sentiment analysis. Neurocomputing 454:291–302

44. Sun M, Zhang X, Zheng J, Ma G (2022) DDGCN: dual dynamic graph convolutional networks for rumor detection on social media. In: Proceedings of the AAAI conference on artificial intelligence, vol 36, pp 4611–4619

45. Yang S, Jiang X, Zhao H, Zeng W, Liu H, Jia Y (2024) FAIMA: feature-aware in-context learning for multi-domain aspect-based sentiment analysis. arXiv preprint arXiv:2403.01063