

TOPIC DETECTION AND SENTIMENT
ANALYSIS OF SHORT VIDEO TEXT COMMENTS

ZHAO RUOCHUN

UNIVERSITI KEBANGSAAN MALAYSIA

TOPIC DETECTION AND SENTIMENT
ANALYSIS OF SHORT VIDEO TEXT COMMENTS

ZHAO RUOCHUN

PROJECT SUBMITTED IN PARTIAL FULFILMENT FOR THE DEGREE OF
MASTER OF DATA SCIENCE

FACULTY OF INFORMATION SCIENCE AND TECHNOLOGY
UNIVERSITI KEBANGSAAN MALAYSIA
BANGI

2025

PENGESANAN TOPIK DAN ANALISIS
SENTIMEN PENDEK KOMEN TEKS VIDEO

ZHAO RUOCHUN

PROJEK YANG DIKEMUKAKAN UNTUK MEMENUHI SEBAHAGIAN
DARIPADA SYARAT MEMPEROLEH
IJAZAH SARJANA SAINS DATA

FAKULTI TEKNOLOGI DAN SAINS MAKLUMAT
UNIVERSITI KEBANGSAAN MALAYSIA
BANGI
2025

DECLARATION

I hereby declare that the work in this thesis is my own except for quotations and summaries which have been duly acknowledged.

01 February 2025

ZHAO RUOCHUN
P135836

LIBRARY FETSM

ACKNOWLEDGEMENT

Firstly, I would like to extend my deepest gratitude to my supervisor, Associate Professor Dr. Suhaila Zainudin, for her invaluable guidance, support and encouragement throughout my time at UKM.

Additionally, I am also profoundly thankful to the staff at UKM for their unwavering assistance and commitment. Especially grateful to the lecturers and fellow students in the Data Science program under the Faculty of Information Science and Technology, whose collaboration and camaraderie created an inspiring and supportive learning environment.

Lastly, I am truly grateful to my family and friends for their steadfast encouragement and support, which have been a constant source of motivation during this journey.

LIBRARY FTSM

ABSTRAK

Dalam era data besar, analisis sentimen terhadap komen video pendek menghadapi beberapa cabaran utama, termasuk mengenal pasti corak emosi tertentu, menangani ketidakseimbangan data dan mencapai pengesanan topik yang tepat. Pengesanan topik, yang melibatkan pengenalan dan pengkategorian tema dalam data teks, adalah penting untuk memahami sentimen pengguna dan penglibatan di platform video pendek. Pemodelan topik yang tepat membolehkan pemahaman yang lebih mendalam tentang perbincangan asas, membantu untuk mendedahkan trend dan keutamaan pengguna. Kajian ini menangani cabaran pengesanan topik melalui kaedah TF-IDF PCA - Pairwise Similarity Graph Laplacian Regularized - Spectral clustering (TP-PSGR-Spectral). Pendekatan ini memastikan perwakilan yang adil bagi sentimen negatif dan neutral, walaupun set data berat sebelah terhadap sentimen positif. Keputusan eksperimen menunjukkan bahawa TP-PSGR-Spectral-k mengatasi teknik pengelompokan tradisional, dengan mencapai Skor Davis Bouldin sebanyak 0.0423 dan Skor Siluet sebanyak 94.37%. Untuk meningkatkan pengesanan topik, kaedah Latent Dirichlet Allocation (LDA) dan TextRank digunakan untuk mengekstrak kata kunci topik yang relevan, membantu menangkap topik popular dan yang lebih halus dalam kandungan yang kurang dibincangkan. Penggabungan teknik ini meningkatkan keseluruhan proses pengesanan topik, memudahkan pengelompokan dan analisis yang lebih tepat. Selain itu, analisis sentimen dilakukan terhadap peristiwa yang berkaitan dengan topik yang dikesan, memberikan wawasan mengenai keutamaan audiens dan trend emosi di platform video pendek. Akhirnya, model rangkaian neural konvolusional gabungan multiskala adaptif (AFM-CNN) yang baru telah dibangunkan, yang meningkatkan ketepatan analisis sentimen kepada 92.12%, mengatasi model-model lain. Kajian ini juga membangunkan analisis visual kecenderungan emosi peristiwa yang terlibat dalam komen teks video pendek, dengan mengekstrakan kata kunci tertentu sebagai teras, membolehkan pemahaman intuitif terhadap pandangan emosi pengguna mengenai peristiwa yang berkaitan.

ABSTRACT

In the era of big data, sentiment analysis of short video comments faces several key challenges, including identifying specific emotional patterns, addressing data imbalance, and achieving accurate topic detection. Topic detection, which involves identifying and categorizing themes within text data, is essential for understanding user sentiment and engagement in short video platforms. Accurate topic modelling allows for a deeper understanding of the underlying discussions, helping to uncover trends and user preferences. This study addresses the challenge of topic detection through a TF-IDF PCA - Pairwise Similarity Graph Laplacian Regularized - Spectral clustering (TP-PSGR-Spectral) algorithm. This approach ensures fair representation of negative and neutral sentiments, even when the dataset is biased toward positive sentiments. Experimental results demonstrate that the TP-PSGR-Spectral-k method outperforms traditional clustering techniques, achieving a Davis Bouldin Score of 0.0423 and a Silhouette Score of 94.37%. To further enhance topic detection, Latent Dirichlet Allocation (LDA) and TextRank methods were used to extract relevant topic keywords, helping to capture both popular and subtle topics in less discussed content. The integration of these techniques improves the overall topic detection process, facilitating more accurate clustering and analysis. Additionally, sentiment analysis was performed on events related to these detected topics, providing insights into audience preferences and emotional trends on short video platforms. Finally, a novel adaptive fusion multi-scale convolutional neural network (AFM-CNN) model was developed, which significantly improves sentiment analysis accuracy to 92.12%, outperforming other models. Finally, this study develops a visual analysis of the emotional tendency of events involved in short video text comments, with the extraction of specific keywords as the core, enabling intuitive understanding of users' emotional views on related events.

TABLE OF CONTENTS

		Page
DECLARATION		iii
ACKNOWLEDGEMENT		iv
ABSTRAK		v
ABSTRACT		vi
TABLE OF CONTENTS		vii
LIST OF TABLES		x
LIST OF ILLUSTRATIONS		xi
LIST OF ABBREVIATIONS		xiv
CHAPTER I	INTRODUCTION	
1.1	Research Background	1
1.2	Problem Statement	3
1.3	Research Objectives	4
1.4	Research Scope	5
1.5	Research Significance	5
1.6	Thesis Organization	6
CHAPTER II	LITERATURE REVIEW	
2.1	Introduction	8
2.2	Topic Detection Modeling	8
	2.2.1 Text Vectorization and Dimension Reduction	9
	2.2.2 Text Clustering Algorithms	11
	2.2.3 Topic Keywords Extraction Methods	18
	2.2.4 Evaluation Metrics for Text Clustering Experiments	20
2.3	Sentiment Analysis Modeling	22
	2.3.1 Valence Aware Dictionary and Sentiment Reasoner Emotional Polarity Annotation	22
	2.3.2 Sentiment Analysis Models	22
	2.3.3 Word Cloud Maps	32

	2.3.4	Evaluation Metrics for Sentiment Analysis Experiments	32
2.4		Related Work	34
2.5		Conclusion	39
CHAPTER III METHODOLOGY			
3.1		Introduction	40
3.2		Data Preparation	42
	3.2.1	Data Acquisition	42
	3.2.2	Data Integration	45
3.3		Data Processing	47
	3.3.1	Delete Missing Values, Deduplication and Denoising	47
	3.3.2	English Word Segmentation and Removal of Stop Words	48
3.4		Topic Detection Experimental Design	48
	3.4.1	Term Frequency-Inverse Document Frequency Text Vectorization	49
	3.4.2	Principal Component Analysis Dimension Reduction	50
	3.4.3	Text Clustering	51
	3.4.4	Topic Detection	55
3.5		Sentiment Analysis Experimental Design	56
	3.5.1	Data Secondary Processing	56
	3.5.2	Data Secondary Processing	58
	3.5.3	Word Cloud Visualization	62
3.6		Conclusion	62
CHAPTER IV RESULTS AND DISCUSSION			
4.1		Introduction	63
4.2		Dataset	63
4.3		Experimental Setup	64
4.4		Topic Detection Research	64
	4.4.1	Text Comment Preprocessing Results	64
	4.4.2	Experimental Design and Result Analysis	66
4.5		Sentiment Polarity Analysis	79

	4.5.1	Affective Polarity Labelling	79
	4.5.2	Experimental Design and Result Analysis	81
4.6		Conclusion	100
CHAPTER V		CONCLUSION AND FUTURE WORKS	
5.1		Introduction	101
5.2		Discussion of Findings	101
5.3		Achievement of Objectives	102
5.4		Limitation	103
5.5		Future Work	103
5.6		Conclusion	104
REFERENCES			106

LIBRARY FETSM

LIST OF TABLES

Table No.		Page
Table 2.1	Distribution of Cluster Samples Table	20
Table 2.2	Evaluation matrix	32
Table 2.3	Related Work Table	35
Table 3.1	comments.csv File Description	43
Table 3.2	video-stats.csv File Description	44
Table 3.3	Data merge File Description	45
Table 3.4	Data Distribution	46
Table 4.1	User comment text number label example	65
Table 4.2	Initial parameter settings of the clustering algorithms	66
Table 4.3	Text clustering experiment results	74
Table 4.4	LDA topic keyword extraction results(e.g. 9,10,26,55)	76
Table 4.5	TextRank topic keyword extraction results(e.g. 9,10,26,55)	78
Table 4.6	Example of sentiment polarity annotation in review data	80
Table 4.7	Distribution of data sets	81
Table 4.8	Initial parameter settings of the model	82
Table 4.9	Experimental results of emotion polarity classification of dataset	89

LIST OF ILLUSTRATIONS

Figure No.		Page
Figure 2.1	Spectral clustering process	14
Figure 2.2	CNN structural framework diagram	23
Figure 2.3	Basic structure of the LSTM unit	24
Figure 2.4	BiLSTM structural framework diagram	27
Figure 2.5	BiLSTM-AT structural framework diagram	28
Figure 2.6	DPCNN model structure diagram	30
Figure 2.7	AFM structural framework diagram	31
Figure 3.1	Research framework diagram	40
Figure 3.2	Diagram of the data file	42
Figure 3.3	comments.csv File	43
Figure 3.4	video-stats.csv File	44
Figure 3.5	Merge dataset results diagram	45
Figure 3.6	Flow chart of data preprocessing	47
Figure 3.7	Result chart of delete missing values, Data deduplication and Data denoising	47
Figure 3.8	Result chart of English word segmentation and stop word removal	48
Figure 3.9	Flow chart of topic detection experiment design	49
Figure 3.10	TF-IDF vectorization results	49
Figure 3.11	PCA dimensionality reduction result	50
Figure 3.12	TP-PSGR-Spectral clustering process	54
Figure 3.13	The top 10 words with the strongest positive sentiment scores	56
Figure 3.14	VADER dictionary polarizes comments	57
Figure 3.15	VADER dictionary polarizes comments (1,0,-1)	57

Figure 3.16	Training and test sets for emotion categories	58
Figure 3.17	AFM-CNN model structure diagram	61
Figure 4.1	Example of data set	64
Figure 4.2	Example of comment texts	65
Figure 4.3	Code and result of k-means algorithm	68
Figure 4.4	Code and result of DBSCAN algorithm	69
Figure 4.5	Code and result of Birch algorithm	70
Figure 4.6	Code and result of TP-Spectral algorithm (e.g. k-means)	71
Figure 4.7	Code and result of TP-PS-Spectral algorithm (e.g. k-means, gamma=0.5)	72
Figure 4.8	Code and result of TP-PSGR-Spectral algorithm (e.g. k-means, gamma=0.5)	73
Figure 4.9	LDA topic keyword extraction code	76
Figure 4.10	TextRank topic keyword extraction code	77
Figure 4.11	VADER sentiment polarity labeling code	79
Figure 4.12	BiLSTM model code (e.g. batch size=64, text20%)	83
Figure 4.13	BiLSTM-AT model code (e.g. batch size=64, text20%)	84
Figure 4.14	CNN model code (e.g. batch size=64, text20%)	85
Figure 4.15	DPCNN model code (e.g. batch size=64, text20%)	86
Figure 4.16	BiLSTM-AT-DPCNN model code (e.g. batch size=64, text20%)	88
Figure 4.17	AFM-CNN model code (e.g. batch size=64, text20%)	88
Figure 4.18	Compare of sentiment model results	95
Figure 4.19	Overview of the flow of word cloud image generation	96
Figure 4.20	Sentiment word cloud code	97
Figure 4.21	\$456000 Squid Game in Real Li	98
Figure 4.22	NEW FPS Chess Game Model event positive and negative emotion word cloud	98

Figure 4.23	10 ESSENTIAL Easy Reads of Western literature event positive and negative emotion word cloud	98
Figure 4.24	10 Infuriating Ways Video Games Stopped You from Getting 100% event positive and negative emotion word cloud diagram	99
Figure 4.25	Cloud map of positive and negative emotion words for 100 Craziest Animal Fights of All Time 2022	99

LIBRARY FTSM

LIST OF ABBREVIATIONS

AFM	Adaptive Fusion Multiscale
AT	Attention
BERT	Bidirectional Encoder Representations from Transformers
BiLSTM	Bidirectional Long Short-Term Memory
BIRCH	Balanced Iterative Reducing and Clustering using Hierarchies
CNN	Convolutional Neural Network
DBS	Davies-Boulding Score
DBSCN	Dual Branch Super Resolution Convolutional Neural Network
DPCNN	Deep Pyramid Convolutional Neural Networks
GCNs	Graph Convolutional Networks
GNNs	Graph Neural Networks
GR	Graph Laplacian Regularization
LDA	Latent Dirichlet Allocation
LSTM	Long Short-Term Memory
NLP	Natural Language Processing
PCA	Principal Component Analysis
PS	Pairwise Similarity
ReLU	Rectified Linear Unit
SS	Silhouette Score
SVM	Support Vector Machine
SVTCD	Short Video Text Comment Dataset
TD	Topic Detection

TF-IDF	Term Frequency-Inverse Document Frequency
TP	Term Frequency-Inverse Document Frequency Principal Component Analysis
UKM	Universiti Kebangsaan Malaysia
VADER	Valence Aware Dictionary and Sentiment Reasoner
Word2Vec	Word to Vector

LIBRARY FETSM

CHAPTER I

INTRODUCTION

1.1 RESEARCH BACKGROUND

In the digital age, where the internet and big data dominate, the primary source of information acquisition is the internet, with short video content emerging as the most efficient means of dissemination. Brief videos lasting a few seconds can rapidly reach a vast audience, generating significant user comments. Platforms like "YouTube" and "TikTok" serve as pivotal conduits for cultural exchange, offering content that ranges from social trends and entertainment to product promotions and news updates. This diverse spectrum is reshaping lives and attracting more users to short video platforms (Huang, 2024).

According to the latest statistical report released by the China Internet Network Information Centre (CNNIC), as of June 2024, the internet user base in China has expanded to 1.1 billion, with the internet penetration rate rising to 73.5% (CNNIC, 2024). The online video segment, including short videos, reached 1000 million users, representing 94.5% of the total internet population (CNNIC, 2024). As of July 2024, the global internet user base has reached 5.45 billion, accounting for 67.1% of the world's population, with social media users totaling 5.17 billion, or 63.7% of the global population.

Among these, short video users comprise 4.1 billion, or 79% (Statista, 2024). In Malaysia, the internet user population is 28 million, with a penetration rate of 81% (Malaysian Communications and Multimedia Commission (MCMC) Report, 2024). Among these, short video enthusiasts' number 24 million, making up 85.7% of Malaysian internet users, highlighting the significant influence.

This data reflects notable growth in internet users across China, the world, and Malaysia, with short video content increasingly shaping social interactions and cultural trends in all regions.

Topic detection is one of the most basic parts of text mining. Such effective categorization can enable an organization to realize emerging trends, consumer interests, and public discourses (Benedetto et al., 2024). In the health or medical field, topic detection is very instrumental in analyzing patient reviews, medical literature, and discussions in health forums to understand what patients are saying and what their needs are, which can be used to enhance healthcare services and communication (Zhou, Fang et al., 2024). This extraction of relevant topics from such unstructured data helps providers understand patient sentiment better and improve the quality of care. In news media, it is applied to the grouping of news articles under their respective topics so that readers can easily locate those that interest them. This allows media organizations to realize what has been attracting public interest lately, making them aware of what directions their reporting work should take (Wang, 2024).

The significance of sentiment polarity analysis is mainly reflected in the following aspects: by analyzing customer reviews, product evaluations or service feedback, companies can quickly identify users' attitudes towards their products or services and help them improve the quality of their products and services (Ho et al., 2024; Pyate & Srinivasan, 2024). For instance, sentiment analysis can help services uncover comprehensive individual evaluations of particular product and services, and make more targeted improvements (Wankhade et al., 2022). In social and political events, view polarity evaluation is commonly utilized for popular opinion tracking. For example, in situations such as elections, policy releases, emergencies, etc., by analyzing the emotional trends of social media and news reports, governments and institutions can understand public attitudes in a timely manner and make corresponding policy adjustments (Sun et al., 2024).

Numerous contributions in the fields of graph learning and semi-supervised learning has given timely impetus to the development of Laplacian regularization methods. For example, the introduction of Laplacian regularization can smoothly

constrain graph-structured data to ensure that connected nodes share similar values or labels (Zhu & Ghahramani, 2021). With the rise of graph neural networks (GNN), Laplacian regularization has once again attracted great interest. For example, Wang et al. (2023) demonstrated how Laplacian based regularization enhances the generalization ability of graph convolutional networks (GCNs) by preserving the smoothness of node embeddings in the graph. The smoothness induced by the Laplacian matrix has been shown to improve clustering accuracy, especially in large scale and noisy datasets.

Driven by the rapid development of artificial intelligence. One of the most noteworthy areas of development is adaptive mechanisms, which enable systems to adjust their parameters or behavior in response to changing inputs or environments. Adaptive mechanisms enable models to dynamically maximize performance without the demand for constant human intervention (Gui et al., 2024). In natural language processing (NLP), adaptive mechanisms assist models deal with diverse linguistic patterns or dialects by fine-tuning their processing pipelines (Yi et al., 2022). These mechanisms are specifically valuable in real time applications, where data conditions are regularly changing, making sure durable and efficient performance in different tasks.

1.2 PROBLEM STATEMENT

There are several challenges in the research problems of topic detection and sentiment analysis experiments in short video comments, Taking the field of movie comments in short video platforms as an example.

Firstly, sentiment analysis in the field of movie reviews within short video platforms may encounter problems with genre-specific trends, multi-scale review lengths, and data imbalance (Gadekallu et al., 2022). From action to drama to humor, the film industry spans genres and evokes different emotional responses. For example, positive reviews of action movies may emphasize excitement, while reviews of drama movies may emphasize emotional depth (Kakarla et al., 2024.). Traditional emotion models sometimes lose the ability to handle complex emotions, resulting in inconsistent predictions (Liu, Zhou, et al., 2023).

Secondly, the problem of genre variation is particularly obvious in the film industry, and different genres will have different impacts. Traditional sentiment models often misclassify reviews by ignoring specific types of cues (Cherradi & El Haddadi, 2024). Using spectral clustering, reviews can therefore be grouped by type, allowing the model to develop specific sentiment analysis strategies based on the expectations of each type (Zhou, H. et al., 2024). For example, for an action movie, the model focuses on excitement or tension, while for a drama, it emphasizes emotional depth or character development. This approach improves various types of sentiment accuracy.

Finally, data imbalance is also one of the important issues in research, as the most popular movies tend to receive more overwhelmingly positive reviews, which can bias sentiment models by ignoring negative or neutral sentiments (Obiedat et al., 2022). Therefore, sentiment information can be propagated in the review graph through Laplacian regularization to enhance the representativeness of minority categories, such as negative reviews, to mitigate this bias and balance sentiment analysis in a timely manner (Shao et al., 2024). In topic detection, generally positive reviews can obscure subtle themes in independent films. For example, movie reviews may focus on excitement and visual effects, while independent films may emphasize narrative depth or character development (Wang & Fan, 2024). This imbalance limits the model's ability to accurately classify movie genre themes, thereby reducing insights into audience preferences.

1.3 RESEARCH OBJECTIVES

The objectives of this study can be summed up as follows:

1. To propose a new clustering algorithm that integrates spectral clustering with Graph Laplacian Regularization and Pairwise Similarity. Methods like LDA and TextRank are used to extract central keywords.
2. To develop a Multi-scale CNN (Convolutional Neural Networks) - based model for improving sentiment analysis accuracy by using adaptive fusion.

3. To visually represent the relationships between topics and sentiments and insights into public opinion trends for practical applications.

1.4 RESEARCH SCOPE

This study focuses on analyzing text comments from short video platforms, specifically using topic detection and sentiment analysis techniques. The research involves collecting and processing a large amount of short video text data and applying cluster analysis, topic detection and sentiment analysis methods to detect users' opinions and emotions, especially those related to various social events. The findings are intended to guide public opinion, inform business strategy and support government decision-making.

The datasets were downloaded from the Kaggle website, and the download link is: <https://www.kaggle.com/datasets/advaipatil/youtube-statistics>. The dataset contains data like the Video ID, Comments, Sentiment, and keyword.

1.5 RESEARCH SIGNIFICANCE

This research is significant in the context of the burgeoning influence of short video content in the digital landscape.

Firstly, from the perspective of enterprise development, enterprises and marketers can use the insights gained from sentiment analysis to enhance product quality and improve customer service satisfaction. Through this form of understanding the emotions and preferences of various audience groups, companies can more efficiently customize products and further cultivate customer loyalty and satisfaction. All in all, this research can guide companies to further improve market responsiveness (Singgalen, 2024).

Secondly, short video bloggers can use emotional insights to better align content strategies with audience preferences, thereby increasing platform exposure and improving user ratings. At the same time, it can timely identify emotions and themes

that resonate with the audience to better help creators produce more relevant and attractive content (Lyu et al., 2024).

Finally, from the perspective of public governance and social responsibility, public departments or institutions monitor public opinion dynamics during major social events or activities in order to better understand public opinion. Various departments can coordinate and adjust policies and strategies in a timely manner. This approach is intended to achieve more effective communication and actions more aligned with public needs and expectations (Anderson et al., 2024).

1.6 THESIS ORGANIZATION

CHAPTER I first introduces the research background of topic detection and sentiment analysis based on short video platforms, and points out the limitations of existing short video text review research. Secondly, this chapter clarifies the research objectives, scope, and methods and outlines how this study addresses these limitations by integrating advanced clustering algorithms and enhanced sentiment analysis models, laying the foundation for subsequent chapters.

CHAPTER II reviews current related research on topic detection and sentiment analysis. First, the clustering algorithm enhanced by the integration of unsupervised learning and deep learning methods is highlighted. Then related topic detection technologies were discussed, and then sentiment analysis models combined with machine learning, deep learning and other technologies were discussed in depth. Finally, word cloud diagrams for emotion visualization are outlined. Together, these technologies provide a solid foundation for further exploration and research.

CHAPTER III details how to use unsupervised learning and deep learning models for topic detection and sentiment analysis of text comments in short videos. The specific experimental workflow including data processing, text clustering, topic keyword extraction, sentiment analysis neural network model and visual sentiment word cloud graph is outlined. Finally, it is concluded that the main contributions are the construction of the TP-PSGR-Spectral algorithm for enhanced text clustering and the AFM-CNN model for accurate sentiment analysis.

CHAPTER IV provides a detailed overview of the specific experimental process and results of topic detection and sentiment polarity analysis based on short video platforms. First, the proposed TP-PSGR-Spectral clustering algorithm outperforms traditional methods such as BIRCH and DBSCAN in text review clustering. Secondly, compare the topic detection results of LDA and TextRank. Subsequently, the AFM-CNN model consistently achieves the best performance in sentiment analysis. Finally, the interpretability of the results is enhanced by visualizing word cloud diagrams. Overall, the TP-PSGR-Spectral algorithm and AFM-CNN model proved to be effective and reliable.

CHAPTER V provides an overall summary of the research, emphasizing the main content and contributions of the research as well as the limitations and areas for improvement found during the research process. Finally, potential future research directions are proposed, focusing on improving topic detection algorithms and sentiment analysis models. The research results can provide a reference for the government and the media to help them understand and respond to public opinion more effectively.

CHAPTER II

LITERATURE REVIEW

2.1 INTRODUCTION

This chapter introduces the research and application of text processing and sentiment analysis technologies along with related models. It begins by emphasizing the roles of TF-IDF and PCA in text vectorization and dimensionality reduction, which lay the foundation for subsequent analysis. The advantages of the VADER tool in social media sentiment analysis are then explored, particularly its ability to handle sentiment polarity scores accurately. Meanwhile, the chapter examines LDA and TextRank methods for topic detection. Clustering algorithms, including K-means, hierarchical clustering, density-based clustering, and spectral clustering, are reviewed, with a focus on the importance of pairwise similarity and graph Laplacian operators in enhancing clustering performance. Additionally, the structures of CNN, BiLSTM, and DPCNN are analyzed in depth, demonstrating their applications in text classification and sentiment analysis. Special attention is given to improving model performance through deep convolution and adaptive multi-scale feature extraction. Finally, an overview of the relevant evaluation metrics is provided to support subsequent experimental operations.

2.2 TOPIC DETECTION MODELING

Short video text comments have certain similarities and differences compared to other platforms such as Facebook and e-commerce. Compared with Facebook text comments, the two have a high degree of similarity, both of which are comments on text, images, or video content published by personal accounts or official media. The length of the comment text generated varies and some comments may contain various internet slang

popular new words, and small emoticons. Compared with e-commerce review texts, there is no difference in language characteristics.

However, in terms of content, user reviews related to a certain brand's product promotion video in short videos not only focus on the products mentioned in the video, but also directly express their views and emotional tendencies towards other series of products under the same brand or even similar products of other brands based on their past usage experience and experiences. For example, the promotional product in short videos is "a 256GB phone memory of a certain brand sold at a 30% discount." Some reviews may mention phrases like "using a phone of that brand has a slow response and lags, so I won't use brand A anymore" or "I think it's not as good as brand B; there's a big difference." After purchasing a specific product in the e-commerce review, various aspects can be discussed. There are certain differences in experience comments. In view of this, this focus on text comment data based on short video platforms with wider user coverage, larger quantity, and more frequent use.

2.2.1 Text Vectorization and Dimension Reduction

a. Term Frequency-Inverse Document Frequency Text Vectorization

Term Frequency-Inverse Document Frequency (TF-IDF) is mainly used for text feature representation and feature weight calculation. TF in TF-IDF stands for word frequency, that is the number of featured words appearing in a certain document. IDF stands for inverse document frequency, which is the inverse of the number of documents that contain a feature word. Suppose the total number of documents is N , the total number of documents terms is W , N_w is used to represent the documents containing word w , w is used to represent the frequency of feature words in a certain document, then the TF-IDF formula can be expressed as follows (Thirumahal, 2024):

$$TF = \frac{w_i}{W}$$

$$IDF = \log \left(\frac{N}{N_w + 1} \right)$$

$$TF - IDF = TF * IDF$$

(2.1)

The purpose of the denominator +1 in the IDF formula above is to prevent the denominator from being 0. The key idea of TF-IDF is that if a feature word appears more frequently in one document and less frequently in other documents, it is considered to be more representative of the main content of the document (Bafna et al., 2016). Therefore, based on TF-IDF, vectorization representation and feature weight calculation are carried out on the pre-processed text review dataset, which is used as the input of the subsequent PCA algorithm to lay the foundation for the principal component analysis and dimensionality reduction of the data (Thirumahal, 2024).

b. Principal Component Analysis Dimension Reduction

Principal Component Analysis (PCA) is mainly used for data dimensionality reduction, and the core operation is eigenvalue decomposition (Afrad et al., 2024). In a literal sense, PCA is to determine the most important components in the data, so as to replace the original data, to achieve dimensionality reduction while minimizing the extent of information loss. Let the sample size be N , the initial data dimensionality be M , and the dimensionality after dimensionality reduction be M' . The initial sample set is D , and the reduced sample set is $D' = \{x_1', x_2', \dots, x_n'\}$, where x^i represents the i^{th} sample ($1 \leq i \leq N$), and w represents the feature vector corresponding to the feature value. The algorithm flow is as follows (He, 2024):

1. Centralize all the input samples;

$$x^i = x^i - \frac{1}{N} \sum_{j=1}^N x^j$$

(2.2)

2. Compute the covariance matrix X^T of the samples and perform eigenvalue decomposition;

3. Select the M' eigenvectors corresponding to the largest eigenvalues to form the eigenvector matrix $W=\{w_1, w_2, \dots, w_{M'}\}$;
4. Transform each sample x^i into the new sample $x^{i'}$ by $x^{i'}=w^T x^i$, thereby obtaining the reduced sample set $D'=\{x_1, x_2, \dots, x_n\}$.

PCA is an unsupervised learning method that does not rely on data labels, similar to clustering algorithms. It compresses and denoises data through eigenvalue decomposition, using only variables to measure information, with orthogonal principal components eliminating interactions between original data components. This process is simple and easy to implement. PCA is applied to vector data after TF-IDF feature representation and weight calculation for dimensionality reduction (Indasari & Tjahyanto, 2023; Abu-Ghoush, 2024).

2.2.2 Text Clustering Algorithms

The key step in the topic detection task is to cluster text data. Four common clustering algorithms based on unsupervised learning and two special methods based on spectral clustering are as follows (Wahyuningrum et al., 2021):

a. Partition Clustering Algorithm

The partition-based clustering algorithm needs to set the number of clusters in advance while building an iterative process (Pitafi et al., 2023). In a typical way, taking the K-means clustering algorithm as an example, its conceptual steps are as follows (Ikotun et al., 2023):

1. Set the initial number of cluster centers K as the initial K centroids;
2. Calculate the distance from each sample point to the cluster center separately, and find the cluster center closest to that point,
3. Assign the sample points to their corresponding clusters; After all sample points belong to the corresponding cluster, recalculate the centroid (average distance of

each cluster Center), designate it as the new cluster center;

4. Repeat the processes 2 and 3 until the distance of cluster center movement is less than the set threshold or the number of clusters. The algorithm terminates when the maximum number of iterations is reached.

The main drawbacks of k-means are that it is difficult to converge on non convex datasets. The corresponding variant algorithms include k-means ++ for optimizing initial centroid selection (Vardakas & Likas, 2024), Elkan k-means for optimizing distance calculation, and Mini Batch k-means for optimizing large sample processing (Zhang, Li et al., 2024; Haji et al., 2024). Among them, k-means and k-means ++ are more commonly used.

b. Hierarchical Clustering Algorithm

Hierarchical clustering, as the name suggests, is the clustering of data based on the idea of layering (Burger et al., 2024). Taking the balanced iterative specification and clustering using the hierarchical method as an example, this article only describes the generation steps of the most critical clustering feature tree, as other algorithm steps are optional and only optimized for clustering results. The clustering feature tree is referred to as CF Tree (Ran et al., 2023), and the leaf nodes are represented by N. The algorithm process are as follows (Sari et al., 2024):

1. Search down from the root node for the N segment closest to the sample point and the CF segment closest to N's internal distance Point; If the radius of the hypersphere corresponding to the CF node is less than the set threshold after adding new sample points, update all CF triplets on the path and end the insertion process. Otherwise, proceed to step 2;
2. If the number of CF node within the current N is less than the set threshold, create a new CF node. Use point as a new sample point and place it in N, while updating all CF triplets on the path and ending the insertion process. Otherwise, proceed to step 3;

3. Divide the current N into two new N 's and select the hypersphere with the most radius among all CF tuples in the old N 's

The two larger CF tuples function as the initial CF node for the two new N . Add other tuples and new sample tuples to the relevant N based on the distance rule and verify if the parent node requires splitting upwards. If necessary, perform splitting of N (Sari et al., 2024). The Balanced Iterative Reducing and Clustering using Hierarchies (BIRCH) algorithm can choose whether to pre-set the number of clusters K , which is suitable for large sample size clustering tasks, saves memory, and has fast clustering speed. Its disadvantage is that it does not perform well in clustering non convex data sets and high dimensional feature data (Rizalde et al., 2024).

c. Density Clustering

The clustering of density clustering is determined based on the sample density. Using typical density-based clustering algorithms (Bhattacharjee & Mitra, 2021). As an example, its algorithm process is as follows (Bhattacharjee & Mitra, 2021):

1. Consider each sample point as the center of the circle and draw a graph with the set radius parameter as the neighborhood;
2. If the number of samples in the neighborhood is greater than the set density threshold, the center of the circle is the core point; Otherwise, it is the boundary point;
3. If a core point is within the neighborhood of another core point, connect the two core points; If the boundary point within the neighborhood of the core point, connect the boundary point with the nearest core point; If there are samples that are not within the neighborhood of any core point, they are considered as noise points.
4. Traversing all sample points terminates.

DBSCAN can cluster corpus of arbitrary shapes and is insensitive to noise points in the samples. The disadvantage of clustering is that if the density distribution of the data is uneven, the clustering effect is not good (Kulkarni & Burhanpurwala, 2024).

d. Spectral Clustering

The flowchart outline diagram of detailed calculation of spectral clustering based on graph theory is shown in Figure 2.1 (Hasan & Abdulazeez, 2021):

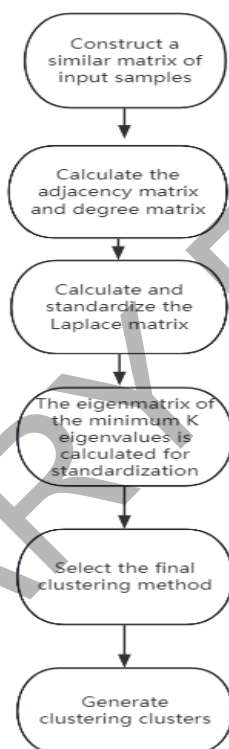


Figure 2.1 Spectral clustering process

Source: Adapted from Hasan and Abdulazeez (2021).

Given a total sample size of N , and a sample set $D = \{x_1, x_2, \dots, x_n\}$, where C represents the cluster labels, it is necessary to reduce the dimensionality of the features to k_1 . Given the number of clusters k_2 , and w represents the feature vector corresponding to the feature values, the algorithm steps are detailed as follows (Hasan & Abdulazeez, 2021):

1. Construct the similarity matrix S for the sample set. Based on the similarity matrix

S , construct its adjacency matrix A , degree matrix B ;

2. Calculate the Laplacian matrix L , $L = B - A$;
3. Normalize L to get L' , $L' = B^{-\frac{1}{2}} * L * B^{-\frac{1}{2}}$, and calculate the feature matrix W composed of the smallest k_1 feature vectors $\{w_1, w_2, \dots, w_{k_1}\}$ of L' ;
4. Standardize W to form an $N * k_1$ dimensional feature matrix W' , where each row of W' is a k_1 -dimensional sample, and then perform clustering using the Discretize or K-means method;
5. Obtain k_2 cluster sets $C = \{c_1, c_2, \dots, c_{k_2}\}$.

The most important step in spectral clustering is the generation of similar matrices, the most common and effective way is based on the generation of Radial Basis Function (RBF), which can map finite dimensional data directly into higher dimensional space, its formula is as follows (Park & Zhao, 2018):

$$R(t, t') = e^{-\frac{\|t-t'\|^2}{2\sigma^2}} \quad (2.3)$$

In the above formula, σ represents the width parameter controlling the local effective range of the RBF. $\|t - t'\|$ represents the Euclidean distance between the two vectors t and t' . In the algorithm steps, k_1 and k_2 are essentially equivalent, i.e., $k_1 = k_2$, representing the same number of dimensions after reduction and the final number of clusters, which unifies the parameters (Park & Zhao, 2018; Hu, F. et al., 2024). Among these, step 5 is the last step of the clustering process, where the clustering method is chosen, either the discretize method or the relatively familiar K-means method (Gao et al., 2024). The K-means method is widely applied and common, while the discretize method is relatively less involved.

The aim of the discretize method is to find a partition matrix (class cluster) that is closest to the eigenvector embedding, where the eigenvector embedding is used to iteratively search the closest discrete partition (Gao et al., 2024). Compared with K-means in spectral clustering algorithm, it is more efficient and robust to random initialization. The general process is as follows (Gupta & Chandra, 2020):

1. The eigenvector is embedded in the block matrix space for normalization, and the optimal discrete block matrix is obtained by multiplying the normalized embedding matrix with the initial rotation matrix;
2. Based on the optimal discrete block matrix, the optimal rotation matrix is calculated, which is a matrix that only changes the size of the vector without changing the direction of the vector after multiplying with the vector;

Repeat steps 1 and 2 until convergence, and finally return a discretized partitioned matrix representing the corresponding class cluster.

e. Clustering Based on Pairwise Similarity

In text clustering, pairwise similarity is a fundamental concept referring to the dimension of similarity between data points. Calculating pairwise similarities helps clustering systems to properly group like products together, hence enhancing the general quality of clustering (Sadeghi & Armanfard, 2024; Tipirneni et al., 2024). Incorporating pairwise resemblance right into clustering techniques enhances their capability to record regional frameworks within the data. As an example, approaches that utilize cosine similarity, Euclidean distance, or Jaccard index make it possible for models to quantify the degree of similarity between texts, resulting in even more systematic clusters (Pawar et al., 2022). Cosine distance has been applied in this study, and the corresponding formulas are as follows (Ghosh & Strehl, 2006):

$$D_{ijs} = \sum_{k=1}^d (x_{ik} * x_{jk}) / \sqrt{\sum_{k=1}^d x_{ik}^2 * \sum_{k=1}^d x_{jk}^2}$$

(2.4)

Cosine distance, a metric for assessing resemblance between two vectors, particularly in between documents i and j . The numerator stands for the dot product of their feature vectors, mirroring their similarity. The denominator normalizes this result and maintains the cosine distance inside the series of 0 to 1, 0 denotes perfect resemblance and 1 shows total dissimilarity. This method is very effective for assessing document similarity.

Overall, using of pairwise similarity measures in graph-based clustering enhances the performance and interpretability of clusters. Spectral clustering among other methods helps models to correctly leverage the links between information points (Hloch et al., 2021). Furthermore, including pairwise similarity into text clustering methods not only improves performance but also advances a better knowledge of the fundamental connections between data points (Tipirneni et al., 2024; Sadeghi & Armanfard, 2024).

f. Clustering Based on Graph Theory

Graph Laplacian regularization has become an effective technique in text clustering, effectively exploiting the inherent structure within the data to enhance clustering performance (Zhang, Yang, et al., 2024). Graph Laplacian regularization is based on graph convolutional networks and effectively utilizes graph structures in text clustering tasks, enabling the model to exploit the relationships between text data points to obtain more accurate and interpretable clustering (Jiang & Lin, 2018; Yang et al., 2021). In addition, Laplacian regularization can also enhance smoothness and capture the local structural characteristics of the data to enhance clustering techniques. For tasks such as text clustering and topic modeling, the preservation of data relationships can lead to more meaningful classification (Jiang & Lin, 2018; Wu et al., 2023). At the same time, Laplacian regularization helps models in deep learning retain relevant information and reduce noise. Traditional clustering techniques may have difficulty detecting significant trends (Jiang, K. et al., 2024). Laplacian regularization helps to solve these problems and thereby improve the efficiency of clustering methods.

Overall, Graph Laplacian Regularization in text-based clustering effectively

improves performance and ensures that relevant data points are grouped according to their natural relationships. Graph Laplacian regularization remains an important research focus for text clustering as the method helps improve accuracy and effectiveness in many different applications through enhanced features (Zhang, Yang et al., 2024; Daneshfar et al., 2024). A typical representative of clustering methods based on graph theory is the spectral clustering algorithm applied in Janani & Vijayarani (2019).

2.2.3 Topic Keywords Extraction Methods

Topic detection plays a pivotal role in understanding the thematic structure of textual data, especially in the context of short video comments. In this study, we employ Latent Dirichlet Allocation (LDA) and TextRank methods to extract topic keywords for each cluster after performing text clustering. These methods are integrated to visually display the results and complete the topic detection task, which is crucial for analyzing large volumes of short video comments (Bonthu et al., 2023). Beyond these conventional methods, recent advancements in topic modeling techniques, such as dynamic topic models (DTM) and hierarchical models, have significantly improved the ability to capture time-evolving topics and hierarchical relationships in complex datasets (Zhang, D. et al., 2023). These advancements in topic modeling allow for a deeper understanding of how topics emerge and evolve, especially in contexts like social media and video platforms where topics shift rapidly.

a. Latent Dirichlet Allocation Method

Latent Dirichlet Allocation (LDA) is a common document topic generation model, which consists of three elements: word, topic and document. Each word in each document selects a specific topic with a certain probability, and selects a keyword in the topic with a certain probability as the final return result. The expression is shown as follows (Hu, N. et al., 2024):

$$P(W|D) = P(W|T) * P(T|D)$$

(2.5)

Where P represents the prior probability, W represents the keyword, D represents the document, and T represents the topic. In order to extract topic keywords more accurately, LDA is used in conjunction with clustering techniques to refine topic modeling results. The number of topics is set to 1 during the experiment, meaning that LDA only extracts keywords from documents under a single dominant topic. Furthermore, recent variations of LDA, such as correlated topic models (CTM) and hierarchical LDA, could further improve topic coherence by accounting for correlations between topics or capturing the hierarchical structure of topics across multiple levels (Zhang, D. et al., 2023).

b. TextRank Method

TextRank is a graph-based ranking algorithm, inspired by Google's PageRank, which is used to extract keywords and identify key phrases in a document. The main idea of TextRank method is to take each word in the data after each segmentation as a node in the network, and all nodes together form a vocabulary network graph (Jiang, Y. et al., 2024). Then, the importance of each word in the network is calculated based on the graph's structure, with higher importance given to words that are more central in the network. The algorithm follows these steps (Zhu et al., 2024), where N represents the number of keywords to be returned:

1. The eigenvector is embedded in the block matrix space for normalization, and the optimal discrete block matrix is obtained by multiplying the normalized embedding matrix with the initial rotation matrix;
2. The network lexical graph is constructed based on the set of candidate keywords. The network nodes are composed of candidate keywords, and the edges are constructed by the co-occurrence relation; The construction condition of the edge: The length of the lexical nodes at both ends appears in the window of the set size;
3. Iteratively calculate the weight value of lexical nodes until convergence;
4. The weight values of nodes are sorted based on the reverse order rule, and the first N terms are taken as the final keywords.

The advantage of TextRank lies in its ability to extract meaningful keywords by considering the global context of words within the document. Recent research has enhanced TextRank's effectiveness by integrating it with other machine learning techniques for even more accurate topic keyword extraction, particularly when dealing with complex and large datasets (Jiang, Y. et al., 2024).

Recent advancements in topic detection have incorporated graph-based message clustering methods to detect communities in social networks, effectively capturing the complex semantic and structural relationships inherent in social media data (Fraj et al., 2024). These methods enable the identification of overlapping topics and communities, which is particularly useful in dynamic, real-time environments like social networks. Furthermore, automated topic categorization has been further enhanced by leveraging techniques such as probabilistic latent semantic analysis (PLSA), which models hidden semantic structures, and multi-view subspace learning, which integrates data from multiple perspectives for more robust topic detection (Zhang, G. Y. et al., 2023). These techniques highlight the increasing sophistication of topic modeling methods, ensuring that topic detection remains accurate and efficient in increasingly complex data environments.

2.2.4 Evaluation Metrics for Text Clustering Experiments

The Davies-Bouldin score (DBS) and Silhouette score (SS) metrics are widely used to evaluate the quality of clustering results (KESSAISSIA, 2024). The details are shown in Table 2.1. Assume that the data has been clustered, with a total of k clusters, the sample set of each cluster is c_i and c_j represents the j^{th} cluster.

Table 2.1 Distribution of Cluster Samples Table

Distribution	c_i	c_j
$D_{i,j}$	σ_i	σ_j
$d_{i,j}$	μ_i	μ_j

Where σ_i is the average distance of samples within the i^{th} cluster, and μ_i is the centroid of the i^{th} cluster; $d_{i,j}$ is the centroid distance between cluster c_i and c_j .

a. Davies-Bouldin Score

Davies-Bouldin Score measures cluster compactness and separation by calculating the ratio of the intra-cluster distance to the inter-cluster distance for each cluster. The lower the score, the better the clustering result. The formula is as follows (Eliza et al., 2024):

$$DBS = \frac{1}{k} \sum_{i=1}^k \max_{j \neq i} \left(\frac{\sigma_i + \sigma_j}{d_{i,j}} \right) \quad (2.6)$$

The formula for the Davies-Bouldin Score (DBS) calculates the average similarity ratio between each cluster and the cluster that is most similar to it. Here, σ_i and σ_j represent the average distance within clusters i and j , while $d_{i,j}$ is the distance between cluster centers i and j .

b. Silhouette Score

Silhouette Score measures how similar a sample is to its own cluster compared to other clusters, with a range of [-1, 1]. The higher the value, the better the clustering effect. The calculation formula is as follows (KESSAISSIA, 2024):

$$S(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}} \quad (2.7)$$

Among them, $a(i)$ represents the mean distance between sample i and other samples in the same cluster, while $b(i)$ denotes the mean distance between sample i and the closest neighboring cluster.

2.3 SENTIMENT ANALYSIS MODELING

Sentiment analysis refers to a method of mining the author's opinions and emotions by analyzing people's text comment data on services, products, events, and topics. Among them, sentiment classification tasks are the most widely used. The sentiment classification of text data is generally divided into two categories: negative and positive (Alslaity & Orji, 2024). The main steps can be summarized as: data acquisition and preprocessing, text vectorization representation, and selection and application of classifiers or neural network models. There are usually rule-based, machine learning based, deep learning based, and pre trained model-based methods for text sentiment polarity classification, among which machine learning based and deep learning-based methods have become the mainstream methods in today's field (Wu et al., 2024).

2.3.1 Valence Aware Dictionary and Sentiment Reasoner Emotional Polarity Annotation

Valence Aware Thesaurus and View Reasoner (VADER) is a view analysis tool particularly designed for social media text, properly taking care of informal expressions such as slang, emojis, and abbreviations (Srivastava et al., 2022). VADER dictionary provides scores for positive, negative, and neutral sentiment, while calculating an overall composite score ranging from very negative to very positive, using a large dictionary of sentiment tags and a specific set of rules to evaluate the sentiment polarity of the text and the overall sentiment trend of the text. This tool is specifically beneficial in areas such as market analysis, public relations, and client service, where it can quickly and effectively analyze and reply to public views in huge datasets (Maulida & Rusydiana, 2022).

2.3.2 Sentiment Analysis Models

a. Basic Structure of Convolutional Neural Network

Convolutional Neural Network (CNN) is a classical deep learning model used to process data with a hierarchical structure, such as images, audio and video (Palanisamy et al., 2020; Zhang, X. et al., 2023). It typically consists of convolutional, pooling, and fully connected layers (Ghosh et al., 2020). The pooling layer down-samples input

features to increase the receptive field, reduce feature dimensions, and decrease the number of parameters for subsequent layers (Zafar et al., 2022; Zhao & Zhang, 2024). The fully connected layer performs a linear transformation of input features, enabling global feature sensing and producing outputs that correspond to classification probabilities. Core of CNN, the convolutional layer generates entire feature maps by means of local operations of the convolution kernel performed across all points from input data (Sakib et al., 2019). Higher-level feature representation (Liu, Yang et al., 2023) is enhanced by increasing the number of convolution kernels, hence facilitating multi-channel feature extracting. The basic network structure is shown in Figure 2.2 (Mohbey et al., 2024):

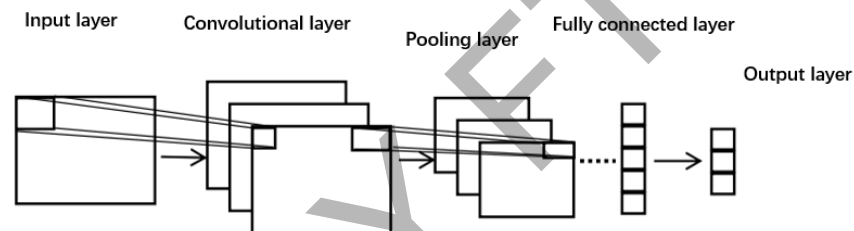


Figure 2.2 CNN structural framework diagram

Source: Adapted from Mohbey et al. (2024).

The model starts with an input layer that gets raw information, such as images or sequential data. The data is processed through convolutional layers making use of kernels to remove regional attributes, adhered to by an activation feature to present non linearity and capture complex patterns. A pooling layer then often max pools and decreases the spatial dimensions of the feature maps, hence lowering computational complexity and improving feature recognition. After flattening the output, it is fed to a fully connected layer incorporating all acquired features for the last prediction or classification. For tasks like image classification (Mohammed et al., 2023), this CNN architecture efficiently learns hierarchical features from raw input, thereby fitting.

CNN network framework, which helps to extract features from input data is based on the mix of convolutional and pooling layers (Sharma et al., 2023; Goumiri et al., 2023). Convolutional and pooling behavior can occur several times and be flexibly

merged with neural network layers, including convolution + convolution + pooling, convolution + convolution, and so on. In addition, the shortcomings of CNNs include slower specification tuning near the input layer when the neural network has way too many layers. The slope descent formula is prone to causing training outcomes merge to local minima (Alnowaiser, 2024) and merging may cause the loss of important info, thereby ignoring the relationship in between global and local information (Gholamalinezhad & Khosravi, 2020).

b. Basic Structure of Long Short-Term Memory Network

In order to more effectively learn the mechanism of long-term dependencies to solve the limitations of sequential data processing, long short-term memory (LSTM) networks are therefore introduced. LSTM storage cells, often called cell states, can retain past information (Sharaff et al., 2023). Three main components define LSTM: input gate, forget gate, and output gate. Together, these three main elements filter and distribute pertinent data (Deng, 2022). Figure 2.3 explains the structure of an LSTM cell (Van Houdt et al., 2020).

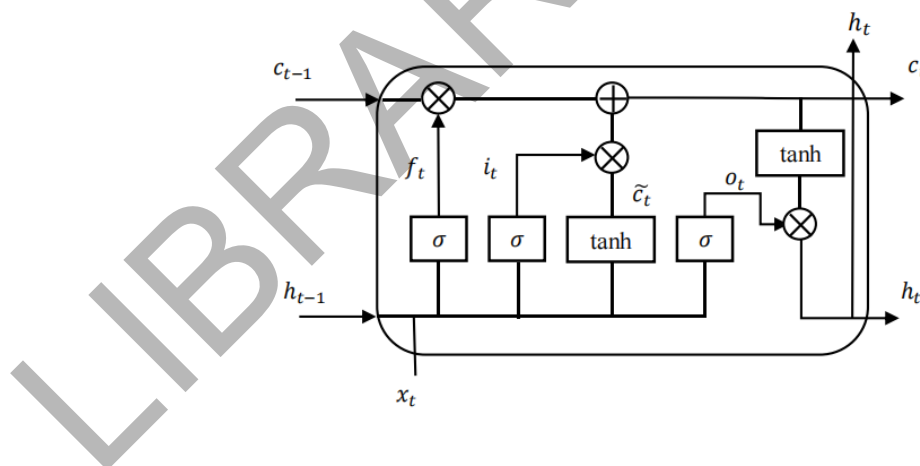


Figure 2.3 Basic structure of the LSTM unit

Source: Adapted from Van Houdt et al. (2020).

In the Figure 2.3, tanh is the activation function, representing the sigmoid activation function. x , and respectively represent the current. Input of time, hidden layer state vector, and cell state. The key processes in the model structure include four parts: forget gate, input gate, cell state and update, and output gate.

1. Forget Gate: Filter out useless information, calculated as follows:

$$f_t = \sigma(W_f * [h_{t-1}, x_t] + b_t) \quad (2.8)$$

The formula $f_t = \sigma(W_f * [h_{t-1}, x_t] + b_t)$ represents the calculation process of the Forgetting Gate in an LSTM unit. f_t is the output of the Forgetting Gate, which controls the proportion of information to forget at the current time step t . σ is the Sigmoid activation function, which converts the input result into a value between 0 and 1, indicating the degree of "forgetting" or "retention." W_f is the weight matrix of the Forgetting Gate, indicating the importance of the previous hidden state h_{t-1} and the current input x_t . $[h_{t-1}, x_t]$ is the concatenated vector of the previous hidden state h_{t-1} and the current input x_t , which is used to decide what information to forget. b_t is the bias term, which allows the model to produce a certain biased output even without input (Van Houdt et al., 2020).

2. Input gate: processes the input of the current sequence position, and the calculation formula is as follows:

$$i_t = \sigma(W_i * [h_{t-1}, x_t] + b_i) \quad (2.9)$$

This formula represents the calculation process of the Input Gate in an LSTM unit. The Input Gate determines the proportion of new information \tilde{c}_t to be added to the cell state at the current time step t . i_t is the output of the Input Gate, with values between 0 and 1, controlling the proportion of new information. σ denotes the Sigmoid activation function, which converts the combined result of weights and input data into a value between 0 and 1. W_i is the weight matrix for the Input Gate, assigning different weights to the previous hidden state h_{t-1} and the current input x_t . $[h_{t-1}, x_t]$ represents the concatenated vector of the hidden state h_{t-1} and the input x_t , providing rich information for the Input Gate to assess. b_i is the bias term for the Input Gate, adjusting the activation of the input (Van Houdt et al., 2020).

3. Cell state and update: Based on the results of the forget gate and input gate, the calculation formula is as follows:

$$c_t = \sigma(W_f * [h_{t-1}, x_t] + b_i)$$

$$c_t = f_t c_{t-1} + i_t \tilde{c}_t$$
(2.10)

These two formulas describe the update process of the cell state c_t in an LSTM unit. The first formula $c_t = \sigma(W_f * [h_{t-1}, x_t] + b_i)$ calculates the candidate value of the cell state, using the Sigmoid activation function σ to compress the weighted combination of the previous hidden state h_{t-1} and the current input x_t into a value between 0 and 1, adjusting the input. The second formula $c_t = f_t c_{t-1} + i_t \tilde{c}_t$ represents the final update of the cell state. It controls the retention of the previous cell state c_{t-1} through the Forget Gate output f_t , and determines the addition of the new candidate value \tilde{c}_t via the Input Gate output i_t . Together, these two formulas allow the LSTM cell state to effectively integrate historical information and current input, achieving memory retention and update (Van Houdt et al., 2020).

4. Output Gate: Output the latest hidden cell state, calculated using the formula:

$$\sigma_t = \sigma(W_o * [h_{t-1}, x_t] + b_o)$$

$$h_t = o_t * \tanh(c_t)$$
(2.11)

These two formulas describe the calculation process of the Output Gate in an LSTM unit. The W series parameters in the above formula represent the weight parameter matrix, while the b series parameters represent the bias term (AI-Selwi et al., 2023). The first formula $\sigma_t = \sigma(W_o * [h_{t-1}, x_t] + b_o)$ calculates the activation value o_t of the Output Gate. Here, σ is the Sigmoid activation function, which compresses the weighted combination of the previous hidden state h_{t-1} and the current input x_t into a value between 0 and 1. W_o is the weight matrix of the Output Gate, and b_o is the bias

term used to adjust the output. The second formula $h_t = o_t * \tanh(c_t)$ represents the calculation of the final hidden state h_t . $\tanh(c_t)$ compresses the current cell state c_t into a range between -1 and 1 using the hyperbolic tangent function, and multiplies it by the Output Gate's activation value o_t , which controls the amount of information to output. This ensures that the LSTM unit only outputs information relevant to the current time step (Van Houdt et al., 2020).

c. Basic Structure of Bidirectional Long Short-Term Memory Network

The unidirectional recurrent neural network only uses the upper data and ignores the lower data, thus limiting its prediction ability (Yao et al., 2024). Integrating the entire data sequence allows the model to make more accurate predictions in real-world applications. The BiLSTM framework provides a bidirectional feature calculation method that is different from the traditional LSTM architecture (Airlangga, 2024). BiLSTM can not only overcome the dependence of long-distance text, but also capture bidirectional contextual semantic information and solve the problems of gradient disappearance and gradient explosion. The basic network structure is shown in Figure 2.4 (Tan et al., 2024).

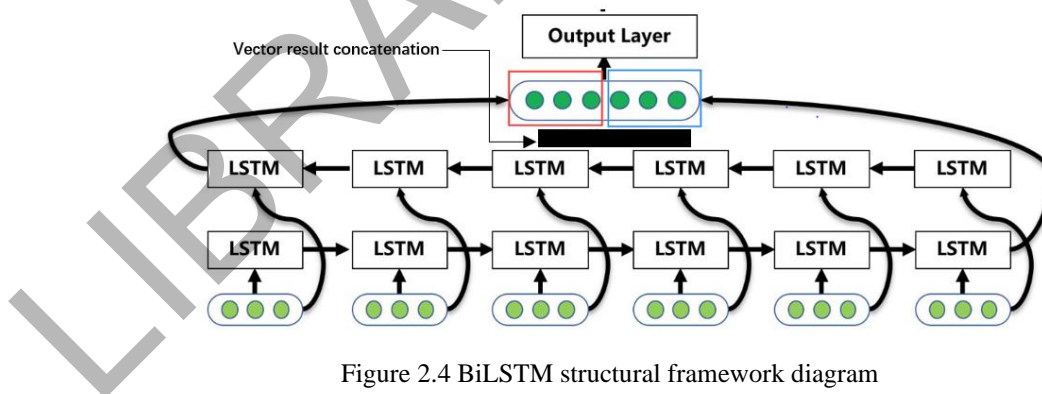


Figure 2.4 BiLSTM structural framework diagram

Source: Adapted from Tan et al. (2024).

BiLSTM (bidirectional long short-term memory) network processes word context data and enhances sentence representation by inputting forward and backward sequence text information. This bidirectional design enables the model to capture semantic and syntactic content more accurately (Devlin et al., 2018). The input sequence is processed simultaneously by forward and backward LSTMs, where the

forward LSTM processes the data from left to right, capturing the dependencies of the earlier parts of the sequence, and conversely, the backward LSTM captures the dependencies of the later parts. Therefore, the combination of forward and backward computations in the BiLSTM model can effectively represent the global context of the sequence, so improving its application to jobs like classification and regression (Tan et al., 2024; Liu & Guo, 2019; Kumar & Yadav, 2023).

d. Basic Structure of Bidirectional Long Short-Term Attention Memory Network

The Bidirectional Long Short-Term Memory with Attention (BiLSTM-AT) network is built on the traditional BiLSTM framework. The integrated attention mechanism not only improves the ability of the BiLSTM model to focus on processing the most task-relevant elements in the input sequence (Huang et al., 2024; Duan & Raga, 2024). Therefore, combining the attention layer to weight the importance of each sequence element enables the model to prioritize and understand important information, thereby improving the performance of tasks such as sentiment analysis and text summarization (Niu et al., 2021). Figure 2.5 illustrates the modified network architecture (Huang et al., 2024).

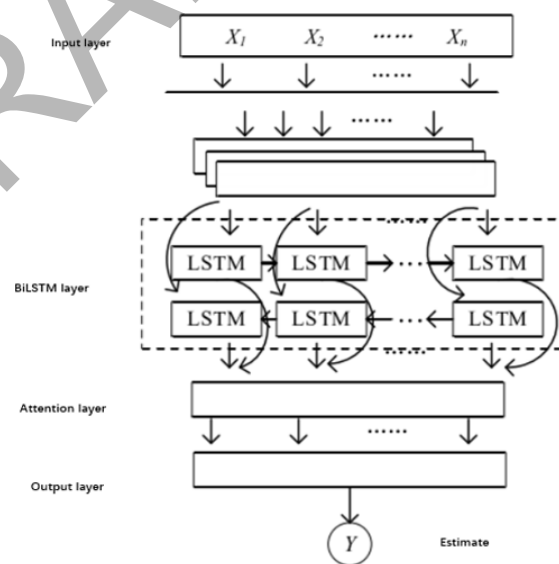


Figure 2.5 BiLSTM-AT structural framework diagram

Source: Adapted from Huang et al. (2024).

The model begins at the input layer, which receives a sequence of text embeddings (X_1, X_2, \dots, X_n) . This data is processed by a BiLSTM layer, capturing contextual information from both forward and reverse sequences, with outputs merged at each time step (Li & Raga, 2023). The attention layer then weights and selects crucial elements of the BiLSTM output, which are passed to the output layer to generate the final prediction Y . This BiLSTM-AT model effectively handles complex NLP tasks, enhancing prediction accuracy and adaptability by combining bidirectional processing with an attention mechanism (Li & Raga, 2023).

The attention mechanism is simply a weight allocation mechanism. The greater the weight, the more critical the characteristics of the vector and the greater the impact of the final emotion polarity classification result. The formula for calculating attention weight is as follows (Ping et al., 2024):

$$L_t = \tanh(W_h * h_t + b_v)$$

$$\alpha_t = \frac{\exp(L_t)}{\sum_t \exp(L_t)}$$

$$R = \sum_t \alpha_t * h_t$$

(2.12)

In the above formulas, h_t is the hidden state vector output of the BiLSTM network layer at time t , W_r is the weight parameter matrix of the attention mechanism layer, and b_v is the bias term. α_t represents the weight calculation for the feature vector, which is normalized using Softmax. R is the output of the attention mechanism layer, obtained by multiplying the feature vector at time t by the corresponding weight coefficients and then applying weighted summation (Li & Raga, 2023; Zamani & Kamaruddin, 2023). This process effectively highlights the key features in the sequence that are most relevant for emotion polarity classification.

e. Basic Structure of Deep Pyramid Convolutional Neural Network

Compared with the traditional CNN, the application of Deep Pyramid Convolutional Neural Network (DPCNN) in text classification tasks mainly overcomes the problem of long-distance dependence that cannot extract text sequences and obtains deeper local feature information through deep convolution (Zhang, M. et al., 2023). Based on its unique pyramid-structure operation mode, DPCNN can halve the calculation time and speed up the training process. The model structure is shown in Figure 2.6 (Huang, W. et al., 2022):

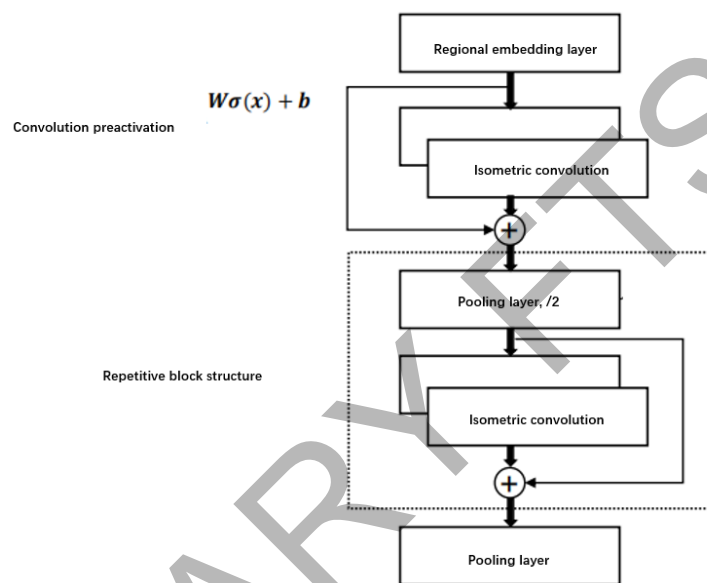


Figure 2.6 DPCNN model structure diagram

Source: Adapted from Huang, W. et al. (2022).

The model starts with a region embedding layer that maps input text into a high dimensional vector space. This is followed by a convolutional layer with kernels parameterized by W to extract local features, with an activation function to introduce non-linearity for capturing complex patterns. In the equal-length convolution layer, input and output lengths are preserved, with residual connections to mitigate gradient vanishing and speed up convergence. Then, a pooling layer down-samples the feature map by half, reducing computational complexity and improving robustness. The model uses a structure of repeated blocks of equal length convolution and pooling, enabling it to extract multi-scale features at deeper levels and enhancing its ability to recognize complex patterns; finally, the final down-sampled feature map is passed to classifiers or other task modules for specific tasks (Zhang, M. et al., 2023). DPCNN, through its

multi-layer convolution and pooling operations, gradually reduces the feature map size while extracting and retaining key features of the input data, making it particularly suitable for text classification tasks by extracting rich contextual information at different scales (Cai & Fu, 2024).

f. Adaptive Fusion Multi-Scale Network Model

The Adaptive Fusion Multiscale (AFM) network enhances traditional deep learning models by integrating a fusion mechanism that adapts to multiple input scales, allowing effective processing across different levels of granularity (Yu et al., 2024). This network combines multi-scale processing, which captures features at various resolutions, with adaptive fusion, which assigns dynamic weights to each scale based on task needs. The fusion mechanism prioritizes relevant features, optimizing the model's ability to recognize complex patterns and improving performance in tasks like image recognition and object detection (Liu, C. et al., 2024; Li et al., 2022). This structure ensures robustness against scale variations, maintaining accuracy across diverse scenarios. Figure 2.7 illustrates the architecture, highlighting adaptive fusion layers of key features at each scale (Yu et al., 2024).

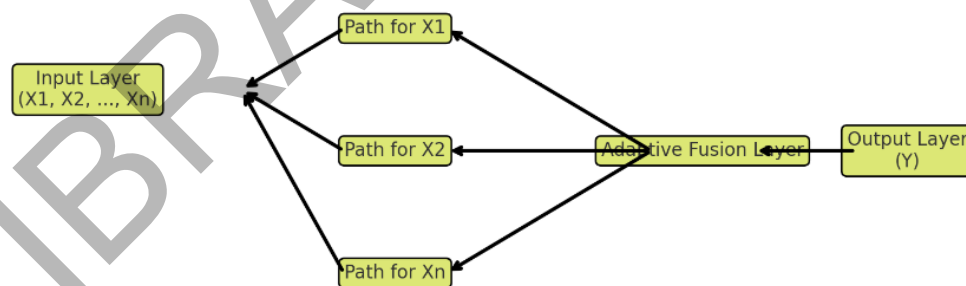


Figure 2.7 AFM structural framework diagram

Source: Adapted from Yu et al. (2024).

The model begins at the input layer, which receives data at multiple scales (represented as X_1, X_2, \dots, X_n). This data is processed through parallel paths, each handling a specific scale to extract relevant features (Zhou & Wang, 2024; Zhou, Y. et al., 2024). Then, the outputs from these paths are fused by the adaptive fusion layer, which dynamically adjusts the contribution of each scale based on the task requirements.

Finally, the output layer transforms the fused output into the final prediction Y (Zhou, T. et al., 2024). By combining multi scale processing with adaptive fusion, this model effectively handles complex tasks requiring a nuanced understanding of varying input resolutions, thereby enhancing prediction accuracy and model robustness (Shang et al., 2020).

2.3.3 Word Cloud Maps

In word clouds maps, entities, relationships and attributes are represented by word sizes, where the size of a word represents the frequency and importance of that element in the graph (AI-Adaileh et al., 2024). Therefore, core entities and relationships are often the most important words, helping users quickly see the main components of the content. In addition, word clouds can further differentiate information types through different colors and clever layouts, such as using different colors to distinguish entities and relationships (Xu et al., 2024). This approach is widely used in a variety of situations, such as optimizing query results in intelligent search engines, providing context for answers in deep question answering systems, and providing customized data in professional domain applications. (Tang et al., 2024).

2.3.4 Evaluation Metrics for Sentiment Analysis Experiments

The evaluation indexes of affective polarity classification in the experiment are Accuracy, F1 value, Precise, Recall and 10-fold validation after macro average (Sarra et al., 2023). Macro average means adding and averaging the values of the corresponding evaluation indexes of each category. The corresponding confusion matrix of the evaluation indexes is shown in Table 2.2:

Table 2.2 Evaluation matrix

Forecast result	Positive sentiment	Negative sentiment
Correct prediction	TP	TN
Prediction error	FP	FN

The true label of TP representing emotional polarity is positive, and the predicted label is positive. TN indicates that the true label is negative, and the predicted label is negative. FP indicates that the true label is positive and the prediction label is negative. FN means the true label is negative and the prediction label is negative. The corresponding formulas of the above four evaluation indicators are as follows (Sarraf et al., 2023):

a. Accuracy

Accuracy rate indicates the percentage of correctly predicted instances out of the total number of instances and is one of the most common classification evaluation metrics for classification problems. The computation formula for accuracy is provided in Equation (2.13) (Sarraf et al., 2023).

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \quad (2.13)$$

b. Precision

Precision measures the proportion of genuinely positive samples among those tagged as such. A higher score indicates that the model produces fewer false positive errors when predicting positive samples. Equation (2.14) shows how precision is calculated (Sarraf et al., 2023).

$$\text{Precision} = \frac{TP}{TP+FP} \quad (2.14)$$

c. Recall

Recall assesses the capacity of the model to identify positive samples, therefore reflecting the proportion of real positive samples accurately detected by the model. The formula for calculating recall is Equation (2.15) (Sarraf et al., 2023).

$$\text{Recall} = \frac{TP}{TP+FN}$$
(2.15)

d. F1-Score

The F1-Score represents the harmonic average of precision and recall. It shows the great performance of the model in producing both accurate and complete positive predictions. The calculation of the F1-Score is Equation (2.16) (Sarra et al., 2023).

$$\text{F1-Score} = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}$$
(2.16)

e. 10-Fold Cross Validation

10-fold cross validation is a model evaluation method that divides datasets into 10 subsets. Once for every subset, validation is conducted; training uses the nine subsets that remain. Every iteration compute accuracy, precision, recall and F1-score. The calculation of the F1-Score is Equation (2.17) (Sarra et al., 2023).

$$F_{1i} = \frac{2 * P_i * R_i}{P_i + R_i}$$
(2.17)

2.4 RELATED WORK

Research on both conventional and hybrid models has been thorough to expose their advantages and drawbacks. With an adaptive multi-scale feature fusion approach to improve object detection in remote sensing photos, Liu Chun et al. (2024) significantly improved datasets including DOTA. Using context extraction and attention methods to hone spatial and semantic features, Shang et al. (2020) presented a multi-scale adaptive feature fusion network (MANet) for semantic segmentation. Comparing SVM to Random Forest, Khan et al. (2024) found SVM somewhat better but confined to traditional techniques. Li et al. (2022) debuted MFEAFN to maximize semantic

segmentation by selective feature fusion. Dong and Fang (2022) also created a sentiment dictionary with an eye toward tourism but lacked cross-domain relevance. Xu et al. (2016) used CLSTM to enhance long-text analysis; although its high processing cost was a disadvantage, although their approach mostly depended on big datasets, Priyadarshini and Cotton (2021) integrated LSTM with CNN to reach great accuracy. Finally, although generalization is still difficult, Huang et al. (2022) improved classification by combining DPCNN and BiLSTM. These studies indicate difficulties in efficiency and flexibility as well as specific tasks as compiled in Table 2.3.

Table 2.3 Related Work Table

Title	Author	Model	Datasets and Performance	Critical Analysis
Sentiment Analysis using Support Vector Machine and Random Forest	Khan et al., (2024)	SVM, Random Forest, TF-IDF, BoW, Preprocessing (tokenization, stop-word removal).	11,997 text samples (split 8:2 for training/testing) SVM Accuracy: 80.39%; Random Forest Accuracy: 78.56%.	SVM performs better in sentiment classification, but both models struggle with handling complex sentence structures.
Cached long short-term memory neural networks for document-level sentiment classification	Xu et al., (2016)	Cached LSTM (CLSTM), Bidirectional CLSTM (B-CLSTM), CIFG-LSTM.	IMDB (84919 reviews, avg. length 394.6 words); Yelp 2013; Yelp 2014. Best accuracy: B-CLSTM: IMDB 46.2%, Yelp 2014 61.9%, Yelp 2013 59.8%.	CLSTM performs well for larger datasets but struggles with long-text accuracy due to its higher computational cost.

to be continued...

...continuation

Attention emotion-enhanced convolutional LSTM for sentiment analysis	Huang et al., (2021)	Emotion-enhanced LSTM (ELSTM), Topic-level Attention Mechanism, Integration with Convolution, Pooling.	IMDB (movie reviews), Amazon Reviews (product reviews). Outperforms state-of-the-art deep learning-based methods.	Significantly improves sentiment classification performance but may overfit on unbalanced datasets. High computational demand.
A novel LSTM-CNN-grid search-based deep neural network for sentiment analysis	Ishani Priyadarshini et al., (2021)	LSTM-CNN architecture with Grid Search hyperparameter optimization.	Dataset 1: Amazon reviews (~4 million reviews); Dataset 2: IMDB (50k movie reviews). Accuracy: 96.4% (Amazon reviews); 97.8% (IMDB).	High accuracy but computationally expensive, unsuitable for smaller datasets or real-time applications.
Sentiment analysis based on Chinese BERT and fused deep neural networks for sentence-level Chinese e-commerce product reviews	Fang Hong et al., (2022)	Chinese-BERT-wwm, CNN, BiLSTM, multi-scale convolution, feature concatenation embeddings.	100,000 sentence-level Chinese e-commerce product reviews (balanced 50% positive, 50% negative). Accuracy: 94.37%; F1 Score: 94.34%	BERT-based methods provide robust performance but require significant computational resources, limiting scalability for smaller systems.

to be continued...

...continuation

Text Sentiment Classification Method Based on DPCNN and BiLSTM	Huang et al., (2022)	DPCNN, BiLSTM, Combination of Local Spatial and Temporal Features.	ChnSentiCorp_hotel (7766 hotel reviews: 6989 training, 777 test, balanced positive/negative) Accuracy: 86.74%.	Strong for spatial and temporal feature extraction, but struggles with highly nuanced sentiment and computational load is high.
BiLSTM model with attention mechanism for sentiment classification on Chinese mixed text comments	Li Xiaoyan et al., (2023)	BiLSTM, Attention Mechanism, Feature Extraction, Data Fusion.	21,091 online shopping comments from Datafountains (8,033 positive, 8,703 negative, 4,355 neutral). Accuracy: 92.80%; 93.58% (long texts).	Performs well on mixed texts, but computationally intensive, requires tuning for domain-specific tasks.
Multiscale adaptive fusion network for hyperspectral image denoising	Pan et al., (2023)	MAFNet (Multiscale Adaptive Fusion Network), Coattention Fusion Module, AIN.	Synthetic: ICVL (31 bands, 1392×1300) and CAVE (31 bands, 512×512); Real: Pavia University, Urban, Indian Pines. Outperforms state-of-the-art methods.	Performs excellently across noise cases but may face difficulty in handling extreme low-light or highly noisy data.

to be continued...

...continuation

Multi-scale adaptive feature fusion network for semantic segmentation in remote sensing images	Shan g et al., (2020)	Multi-scale Adaptive Feature Fusion Network (MANet): Multi-scale Context Extraction Module (MCM), Adaptive Fusion Module (AFM).	Potsdam (38 images, 6000x6000 pixels, RGB + DSM) Vaihingen (33 images, 2500x2000 pixels, IRRG + DSM). Potsdam: OA = 89.4%, Average F1 = 90.4%; Vaihingen: OA = 88.2%, Average F1 = 86.7%.	Outperforms six state-of-the-art models (FCN8s, U-net, APPD, etc.) but still limited by processing speed and adaptation to more complex datasets.
MFEAFN: Multi-scale Feature Enhanced Adaptive Fusion Network for Image Semantic Segmentation.	Li et al., (2022)	Double Spatial Pyramid Module (DSPM), Focusing Selective Fusion Module (FSFM), EfficientNetV2-S backbone.	PASCAL VOC 2012 (10,582 training images), Cityscapes (5000 images with 2975 for training). PASCAL VOC: mIoU = 82.64%; Cityscapes: mIoU = 78.46%.	Outperforms DeepLabv3+ and other state-of-the-art methods, but computational cost is high and may require further optimization for larger-scale applications.
Object Detection in Remote Sensing Images Based on Adaptive Multi-Scale Feature Fusion Method.	Liu Chun et al., (2024)	Adaptive Multi-Scale Feature Enhancement and Fusion Module (ASEM), Atrous Convolutions, Attention Mechanism.	DOTA-v1.0 (2806 images, 15 object categories), HRSC2016 (1061 images for ship detection).mAP: 74.21% (+0.81%) on DOTA-v1.0, 84.90% (+9.2%) on HRSC2016.	High performance in object detection tasks, but computational complexity can be limiting in real-time detection applications.