

# CONTEXTUAL FRAMEWORKS FOR SENTIMENT ANALYSIS

by

**Ashima Yadav**

*A Dissertation submitted to the Delhi Technological University for  
the Award of degree of*

**DOCTOR OF PHILOSOPHY**

IN

**INFORMATION TECHNOLOGY**

Under the Supervision of

**Prof. Dinesh Kumar Vishwakarma**



Department of Information Technology

**Delhi Technological University**

*(Formerly Delhi College of Engineering)*

Delhi, India

**April 2021**



©Delhi Technological University-2021  
All rights reserved.

# DECLARATION

---

I certify that the dissertation titled “*Contextual Frameworks for Sentiment Analysis*” that is being submitted by me for the Doctor of Philosophy degree is my own work and has not been submitted for the award of any degree or diploma to any other University or Institute. The work done in the thesis is original and has been done by me under the supervision of my supervisor.

**Place:** Delhi Technological University, Delhi

**Date:**

This is to certify that the above statements made by the candidate are true.

**Ashima Yadav**  
**2K17/Ph.D./IT/06**

# CERTIFICATE

---

This is to certify that the work contained in the thesis entitled “*Contextual Frameworks for Sentiment Analysis*” submitted by Ms. Ashima Yadav (2K17/Ph.D./IT/06) for the award of the degree of Doctor of Philosophy to Delhi Technological University, India includes original research work carried out by her under my supervision.

She has fulfilled all the requirements as per the required standard for the submission of the thesis. I hereby confirm the originality of the work and certify that the thesis has not formed the basis for the award of any degree or similar title.



**Prof. Dinesh Kumar Vishwakarma**  
Supervisor

Department of Information Technology  
Delhi Technological University

# ACKNOWLEDGEMENT

---

This research journey would not have been possible with the help of many people. First and foremost, I would like to thank my supervisor **Prof. Dinesh Kumar Vishwakarma**, for his constant guidance and motivation. His dedication to research has always encouraged me to work hard and strive for the best. I appreciate all his efforts, time, ideas, patience, and constant feedback on all the research papers. I would also like to thank my colleagues and lab members who have continuously supported and helped me with their valuable suggestions as and when required.

Finally, I would like to thank my parents, **Mr. Ajay Yadav** and **Mrs. Santosh Yadav**, my sister, **Shubhi Yadav**, for their unfailing love and support in all my pursuits. Without them, this journey would have been impossible.

**Place:** Delhi Technological University, Delhi

**Date:**

**Ashima Yadav**  
**2K17/Ph.D./IT/06**

# ABSTRACT

---

Social media is a powerful source of communication among people to share their sentiments, opinions, and views about any topic or article, which results in an enormous amount of unstructured information. Business organizations need to process and study these sentiments to investigate data and to gain business insights. The previous research in sentiment analysis has majorly focused on extracting the sentiments from the textual data only. Thus, various machine learning and natural language processing-based approaches have been used to analyze these sentiments. The text-based sentiment classification suffers from various challenges like domain adaptation, sarcasm detection, multilingual sentiment classification, etc. [1] [2]. However, with the evolution of the web and smartphones, the sentiments can be extracted from varied multimedia content, including text, images, videos, emoticons, GIFs, audios, etc., found on social media networks. These multimodal data like images can detect sarcastic posts by analyzing the facial expressions from the visual data.

Most of the earlier works in this area are based on handcrafted features that fail to explore the high-level semantics of the data. These approaches cannot handle the massive amount of data and require ample time and effort to manually extract the features, impacting the classifier's performance. Hence, the great feats of deep learning, especially in computer vision, have motivated us to apply them in sentiment analysis. These approaches can automatically learn the complex features from the data, thus improving the sentiment analysis process. However, this area still suffers from many challenges. The visual sentiment analysis process is abstract in nature due to the high biasing level in the human recognition process. Similarly, affective video content analysis has emerged as one of the most challenging research tasks as it aims to analyze the emotions elicited by videos automatically. However, little progress has been achieved in this field due to the enigmatic nature of emotions. This widens the gap between the human affective state and the structure of the video. The multimodal sentiment analysis area remains an open problem because each modality has its individual characteristics and is expressed differently by the human cognitive system. Thus, it isn't easy to deal with such heterogeneous content for multimodal analysis.

Hence, our work aims to address the issues mentioned above by designing effective frameworks. Chapter 1 gives the background of sentiment analysis and outlines the motivation behind the research. Chapter 2 is dedicated to the literature review, where the existing state-of-the-arts for sentiment classification are reviewed for textual and visual (images and videos) modalities. The prevalent approaches in each of the modalities are grouped into a taxonomy, which helped in identifying the research gaps in this area. Finally, the research objectives are also briefly addressed. In Chapter 3, we discuss two approaches corresponding to the image and video modalities. We apply transfer learning by utilizing the pre-trained models with visual attention for learning the high-level discriminative features from the images to address the problem in visual sentiment analysis. For affective video classification, we propose a deep affect-based movie genre classification framework that aims to study the relationship between the induced emotions in the movie trailer and its corresponding genre by developing Emotion-based Genre Detection for Bollywood (EmoGDB) Dataset, which helps to create the Emotion-genre based theory.

In order to address the challenges for multimodal sentiment classification, we propose a network in Chapter 4 that generates the discriminative features from the visual images and their textual descriptions by introducing attention at multiple levels. We utilize the channel dimension to generate robust visual features, enhancing the crucial channels in the given image. Further, we extract the essential sentiment words corresponding to the image features by employing semantic attention, which boosts our network's overall performance. One of the significant applications of sentiment analysis is to analyze the opinion of the people. Recently, with the outbreak of the COVID-19 pandemic, enormous amount of sentiments were being generated on Twitter, which could help to assess the people's attitude and behavior related to the pandemic. In Chapter 5, we designed a Multilevel Attention-based Conv-BiGRU Network to classify the opinions of the people posted on Twitter from the countries that were worse-affected by the pandemic so that the analysis can serve as feedback to the government agencies regarding the mitigation plans taken by them.

Finally, in Chapter 6, we summarize the conclusions inferred from our research work and highlight future work in this area.

# LIST OF AUTHOR'S PUBLICATIONS

---

- 1) **Ashima Yadav**, Dinesh Kumar Vishwakarma. “Sentiment analysis using deep learning architectures: a review.” *Artificial Intelligence Review*, 53.6 (2020): 4335-4385 (**Impact Factor: 5.747**). (Pub: Springer).
- 2) **Ashima Yadav**, Dinesh Kumar Vishwakarma. “A unified framework of deep networks for genre classification using movie trailer.” *Applied Soft Computing*, 96 (2020): 106624 (**Impact Factor: 5.472**). (Pub: Elsevier).
- 3) **Ashima Yadav**, Dinesh Kumar Vishwakarma. “A deep learning architecture of RA-DLNet for visual sentiment analysis.” *Multimedia Systems*, 26 (2020): 431-451 (**Impact Factor: 1.563**). (Pub: Springer).
- 4) **Ashima Yadav**, Dinesh Kumar Vishwakarma. “A comparative study on bio-inspired algorithms for sentiment analysis.” *Cluster Computing*, 23 (2020): 2969–2989 (**Impact Factor: 3.458**). (Pub: Springer).
- 5) **Ashima Yadav**, Dinesh Kumar Vishwakarma. “A Language-independent Network to analyze the impact of COVID-19 on the World via Sentiment Analysis” *ACM Transactions on Internet Technology (Major Revision-1)* / (*arXiv:2011.10358*)
- 6) **Ashima Yadav**, Dinesh Kumar Vishwakarma. “A Deep Multi-Level Attentive network for Multimodal Sentiment Analysis.” (*arXiv:2012.08256*).
- 7) Dinesh Kumar Vishwakarma, Deepika Varshney, and **Ashima Yadav** “Detection and veracity analysis of fake news via scrapping and authenticating the web search.” *Cognitive Systems Research*, 58 (2019): 217-229. (**Impact Factor: 1.902**). (Pub: Elsevier).
- 8) **Ashima Yadav**, Dinesh Kumar Vishwakarma. “A Weighted Text Representation framework for Sentiment Analysis of Medical Drug Reviews.” at *IEEE Sixth International Conference on Multimedia Big Data (BigMM)*, IIIT Delhi, 2020.
- 9) **Ashima Yadav**, Dinesh Kumar Vishwakarma. “Multilingual Framework of CNN and Bi-LSTM for Emotion Classification.” at *IEEE 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, IIT Kharagpur, 2020.
- 10) **Ashima Yadav**, Ayush Agarwal, and Dinesh Kumar Vishwakarma. “XRA-net



framework for visual sentiments analysis.” at *IEEE Fifth International Conference on Multimedia Big Data (BigMM)*, National University of Singapore, Singapore, 2019.

- 11) Ayush Agarwal, **Ashima Yadav**, and Dinesh Kumar Vishwakarma. "Multimodal sentiment analysis via RNN variants." at *IEEE International Conference on Big Data, Cloud Computing, Data Science & Engineering (BCD)*, Hawaii, USA, 2019.

# Table of Contents

---

|  |             |
|--|-------------|
| <b>DECLARATION</b> .....                                     | <b>i</b>    |
| <b>CERTIFICATE</b> .....                                     | <b>ii</b>   |
| <b>ACKNOWLEDGEMENT</b> .....                                 | <b>iii</b>  |
| <b>ABSTRACT</b> .....  | <b>iv</b>   |
| <b>LIST OF AUTHOR’S PUBLICATIONS</b> .....                   | <b>vi</b>   |
| <b>Table of Contents</b> .....                               | <b>viii</b> |
| <b>List of Figures</b> .....                                 | <b>x</b>    |
| <b>List of Tables</b> .....                                  | <b>xii</b>  |
| <b>Chapter-1 Introduction</b> .....                          | <b>1</b>    |
| 1.1 Background .....   | 1           |
| 1.2 Motivation.....  | 5           |
| 1.3 Major Challenges .....                                   | 6           |
| 1.4 Problem Statement .....                                  | 7           |
| 1.5 Methods and Results Overview .....                       | 8           |
| 1.6 Significance of the Study .....                          | 9           |
| 1.7 Dissertation Outline .....                               | 10          |
| <b>Chapter-2 Literature Review</b> .....                     | <b>12</b>   |
| 2.1 Textual Sentiment Analysis .....                         | 12          |
| 2.1.1 Sentiment analysis tasks .....                         | 13          |
| 2.1.2 Handcrafted Feature-based Text Sentiment Analysis..... | 17          |
| 2.1.3 Deep based Textual Sentiment Analysis.....             | 19          |
| 2.1.4 Sentiment Analysis in Healthcare.....                  | 22          |
| 2.2 Visual Sentiment Analysis .....                          | 24          |
| 2.2.1 Low-level feature-based approaches.....                | 24          |
| 2.2.2 Mid-level feature-based approaches .....               | 25          |
| 2.2.3 Deep-based feature-based approaches .....              | 26          |
| 2.3 Multimodal Sentiment Analysis .....                      | 27          |
| 2.3.1 Multimodal Fusion.....                                 | 28          |
| 2.4 Research Gaps.....                                       | 29          |
| 2.5 Research Objectives.....                                 | 30          |

|   |            |
|---|------------|
| 2.6 Research Contribution .....   | 31         |
| <b>Chapter-3 Visual Sentiment Analysis and Understanding .....</b>  | <b>33</b>  |
| 3.1 A deep learning architecture of RA-DLNet for visual sentiment analysis .....                                    | 33         |
| 3.1.1 Proposed Methodology .....  | 34         |
| 3.1.2 Experimental Results .....  | 38         |
| 3.2 A unified framework of deep networks for genre classification using movie trailer .....                         | 57         |
| 3.2.1 Proposed EmoGDB Dataset.....  | 57         |
| 3.2.2 Proposed Methodology .....  | 61         |
| 3.2.3 Experiments .....   | 69         |
| 3.3 Significant Outcomes.....   | 78         |
| <b>Chapter-4 Multimodal Sentiment Analysis.....</b>   | <b>81</b>  |
| 4.1 A Deep Multi-Level Attentive network for Multimodal Sentiment Analysis ...                                      | 81         |
| 4.1.1 Proposed Methodology .....  | 81         |
| 4.2.2 Experiments .....   | 86         |
| 4.2 Significant Outcomes.....   | 96         |
| <b>Chapter-5 Application of Sentiment Analysis.....</b>   | <b>98</b>  |
| 5.1 A Deep Language-independent Network to analyze the impact of COVID-19 on the World via Sentiment Analysis ..... | 98         |
| 5.1.1 Proposed Methodology .....  | 98         |
| 5.1.2 Experimental Analysis .....   | 105        |
| 5.2 Significant Outcomes.....   | 117        |
| <b>Chapter-6 Conclusion and Future Work.....</b>  | <b>119</b> |
| 6.1 Conclusion .....  | 119        |
| 6.2 Future Work.....  | 120        |
| <b>References.....</b>  | <b>122</b> |
| <b>Author's Biography .....</b>   | <b>138</b> |

# List of Figures

---

|  |    |
|--|----|
| Figure 1.1 Basic architecture of Sentiment Analysis.....   | 3  |
| Figure 2.1 Taxonomy of Sentiment Analysis .....  | 12 |
| Figure 3.1 Residual Attention Network [135] .....  | 34 |
| Figure 3.2 Proposed RA-DLNet architecture .....  | 36 |
| Figure 3.3 Sample images from (a) Twitter dataset (b) Artphoto dataset .....   | 39 |
| Figure 3.4 Performance comparison of proposed architecture on Artphoto dataset with similar state-of-the-art.....  | 46 |
| Figure 3.5 Confusion Matrix for (a) Artphoto dataset (b) F&I dataset .....   | 46 |
| Figure 3.6 Training and validation curves for each dataset .....   | 51 |
| Figure 3.7 Activation Maximation Plots to visualize each channel generated by different layers of the RA-DLNet architecture (b) Colormaps and corresponding heatmaps of some images to visualize the crucial regions of example images .....   | 54 |
| Figure 3.8 Sample images from EmoGDB dataset (a) 1920: Horror (b) Ae Dil Hai Mushkil : Romance (c) Gameover : Thriller (d) Behen Hogi Teri : Comedy (e) Chhichhore : Drama (f) Baahubali 2_The Conclusion : Action .....                       | 58 |
| Figure 3.9 (a) Folder structure of the proposed dataset (b) Naming format rules for folders and sub-folders.....   | 59 |
| Figure 3.10 Different face orientations and occluded images are captured by our face detection algorithm. ....   | 60 |
| Figure 3.11 Pipeline of the proposed framework.....  | 62 |
| Figure 3.12 Applying 1*1 convolutions for reducing the depth of an image without altering the spatial dimensions.....  | 64 |
| Figure 3.13 Emotion-genre mapping for classifying the movie-trailer into multiple genres .....   | 68 |
| Figure 3.14 Evaluating the performance of different variations of ILNet architecture with (a) Training accuracy + Validation accuracy curves (b) Training Loss + Validation Loss curves.....   | 72 |
| Figure 3.15 Confusion matrix for (a) LMTD-9 (b) MMTF-14K (c) ML-25M dataset generated with model trained on five emotions (d) Confusion matrix for LMTD-9 dataset generated with model trained on six emotions (including neutral emotion) ... | 74 |

|   |            |
|---|------------|
| Figure 3.16 Precision-Recall curves for (a) LMTD-9 (b) MMTF-14K, and (c) ML-25M datasets.....   | <b>76</b>  |
| Figure 3.17 Visualizing the discriminative image regions captured by the model for identifying different emotions in the movie trailers .....                                   | <b>78</b>  |
| Figure 4.1 Block diagram of the proposed DMLANet.....   | <b>82</b>  |
| Figure 4.2 Block diagram explaining the Visual Attention Module.....  | <b>83</b>  |
| Figure 4.3 Experimental results on the datasets (%)......   | <b>89</b>  |
| Figure 4.5 (a) Training and Validation Loss curves (b) Training and Validation accuracy curves on MVSA-Multiple Dataset .....   | <b>90</b>  |
| Figure 4.4 (a) ROC curves (b) PRC curves for the datasets .....   | <b>90</b>  |
| Figure 4.6 Quantitative analysis of DMLANet for (a) Positive image-text pairs (b) Negative image-text pairs on MVSA Dataset.....  | <b>95</b>  |
| Figure 5.1 Block diagram of the proposed Multilevel Attention-based Conv-BiGRU Network (MACBiG-Net).....  | <b>99</b>  |
| Figure 5.2 Sample tweets in different languages from COVID-19 Sentiment Dataset (a) Hindi to English (b) Spanish to English (c) Arabic to English (d) Japanese to English. .... | <b>106</b> |
| Figure 5.3 (a) Sentiment classification results on COVID-19 Sentiment Dataset (%) (b) Confusion Matrix. ....  | <b>108</b> |
| Figure 5.4 Tracing internal performance of MACBiG-Net with (a) Accuracy curve for train and validation set (b) Loss curves for train and validation set. ....                   | <b>108</b> |
| Figure 5.5 Visualizing the sentiments of people over time in top five worst affected countries by COVID-19 pandemic. ....   | <b>109</b> |
| Figure 5.6 Comparison of MACBiG-Net with baseline methods in terms of ROC curve and Area under the curve (AUC). ....  | <b>113</b> |
| Figure 5.7 Visualization the word-level attention weights from proposed dataset: (a)-(b) Positive Sentiment (c)-(d) Negative Sentiment.....                                     | <b>114</b> |
| Figure 5.8 Identifying the hidden information in document through LDAvis visualization for (a) Positive and (b) Negative sentiment.....   | <b>115</b> |

# List of Tables

---

|  |    |
|--|----|
| Table 2.1 List of handcrafted features .....   | 19 |
| Table 3.1 Twitter Dataset.....   | 39 |
| Table 3.2 ArtPhoto Dataset Details .....   | 40 |
| Table 3.3 Classification results (%) of RA-DLNet on benchmark datasets .....   | 42 |
| Table 3.4 Performance comparison of proposed architecture on Twitter I (%) .....   | 44 |
| Table 3.5 Performance comparison of proposed architecture on Twitter II dataset (%)<br>.....   | 45 |
| Table 3.6 Performance comparison of proposed architecture on CMU-MOSI dataset<br>(%).....  | 47 |
| Table 3.7 Performance comparison of proposed architecture on CMU-MOSEI dataset<br>(%).....   | 47 |
| Table 3.8 Performance comparison (ACC) of proposed architecture on Flickr and<br>Instagram Sentiment Datasets (%).....               | 49 |
| Table 3.9 Performance comparison of proposed architecture on F&I Emotion Dataset<br>(%).....   | 50 |
| Table 3.10 ACC (%) on all the datasets with (80:10:10) split.....  | 51 |
| Table 3.11 ACC results (%) for layer ablation study on Flickr dataset.....   | 53 |
| Table 3.12 Summary of CNN architectures.....   | 55 |
| Table 3.13 Five-cross validation results (mean $\pm$ std) on other popular architectures   | 55 |
| Table 3.14 Training efficiency of all architectures (Time/Epoch) (sec).....  | 56 |
| Table 3.15 List of Abbreviations with their meanings .....   | 60 |
| Table 3.16 List of symbols with their meanings.....  | 61 |
| Table 3.17 Parameters of ILDNet architecture .....   | 63 |
| Table 3.18 Classification results of ILDNet on LMTD-9, MMTF-14K, and ML-25M<br>datasets (P: Precision, R: Recall, F1: F1 score)..... | 73 |
| Table 3.19 Comparison of ILDNet architecture with previous works using micro-<br>average AU(PRC) metric.....                         | 75 |
| Table 3.20 Time complexity of movie genre classification methods .....   | 77 |
| Table 4.1 Overall Statistics of each Dataset .....   | 88 |

|   |     |
|---|-----|
| Table 4.2 Comparison Results of different methods for MVSA Datasets (%).....              | 91  |
| Table 4.3 Comparison Results of different methods for Flickr and Getty images (%)92       |     |
| Table 4.4 Ablation studies on MVSA-Multiple and Flickr Datasets.....                      | 93  |
| Table 5.1 Layer description of Step 2: Word-level Encoding and Attention .....            | 103 |
| Table 5.2 Layer description of Step 3: Sentence-level Encoding and Attention .....        | 104 |
| Table 5.3 Country-wise details of COVID -19 Sentiment Dataset.....                        | 107 |
| Table 5.4 Comparative Results of different methods on COVID-19 Sentiment Dataset (%)..... | 113 |

# Chapter-1 Introduction

---

With the boom in digital technology and the proliferation of smartphones, social media has witnessed an exponential increase in online data being generated by users. According to [3], Facebook now has around 2.7 billion active users and has the biggest social network. Social media provides a platform for users to express themselves through diverse modalities, which include: text, images, videos, emoticons, GIFs, audios, etc. [4] [5] [6] [7]. Such an information-rich social media repository offers an excellent opportunity to study the behavior of the users. Moreover, the decision-making process of commercial companies and organizations is highly affected by the opinion of these online users. This produces a need for building automated sentiment analysis systems, which could effectively and efficiently capture the opinions or sentiments from different modalities to make optimum decisions. This thesis aims to design contextual sentiment analysis frameworks to investigate the sentiments expressed from the wide-range of contextual input data in the form of images, videos, and text.

This chapter focuses on developing an elementary background for sentiment analysis by discussing the fundamentals, motivation, and prominent challenges in this field. Based on these challenges, problem statements for the research is developed, followed by an outline of the dissertation.

## 1.1 Background

Sentiment Analysis (SA), popularly known as opinion mining, belongs to the domain of Natural Language Processing (NLP). It is the field of study which identifies and extracts the opinions, sentiments, emotions, views, attitude towards any target entity [8]. The previous research in this area has majorly focused on extracting the sentiments from the textual data only. However, with the evolution of the web and smartphones, the target entity can now belong to the varied multimedia content found on social media networks. Therefore, this cross-disciplinary field of sentiment analysis shows the intersection of NLP with the popular area of computer vision. This further boosts the widespread application of SA, which includes: Healthcare [9], Business review



analytics [10] [11], Human-agent interaction [12], Politics [13] [14], Financial market prediction [15] [16], Crime prediction [17], Demonetization [18], Disaster management [19] [20] [21], Affective image retrieval [22], Entertainment [23].

Despite the popularity of this field, it faces several challenges in modeling multimedia content. The conventional text-based SA approach suffers from the problems of implicit sentiment detection, irony and sarcasm detection, handling multilingual data, word ambiguity, understanding the semantic and syntactic structure of data, negation handling, and length limitation of the text [163] [164]. However, the emerging area of visual sentiment analysis (VSA) overcomes these issues to a great extent. The vision-based SA aims to capture the sentiments from the visual multimedia (like images or videos), which serves as the non-verbal expressions for sentiments. It aims to automatically extract the sentiments evoked from the facial expressions or body gestures or from other visual content like an image of a pleasant garden conveying positive sentiment.

While this area of automatic object recognition is well-defined, the same cannot be alleged for VSA. This is because of the abstract nature of visual sentiments, which leads to a high level of subjectivity in the human recognition process. The human cognition process highly drives the emotional content expressed by the images. Object recognition identifies the objects in the image, whereas VSA recognizes the objects or scenes along with their emotional context [24]. Moreover, this area suffers from the problem of high inter-class and intra-class similarity. For instance, the images of “*Happy Boy*” and “*Sad Boy*” belong to the same Boy class. Still, the sentiment represented by them is the opposite. One signifies a positive sentiment with the *happy* keyword, and the other signifies a negative sentiment with the *sad* keyword. Thus, the vision-based sentiment classification models need to learn the mapping from low-level visual features depicted in the form of raw pixels or motion to the high-level sentiment labels.

Emotion recognition belongs to the domain of affective computing. The field of affective computing and SA are highly interdependent as the former deals in emotion recognition, and the latter focuses on polarity detection [25]. Thus, many models like

Hourglass of emotions [26] have classified the sentiments based on the emotions expressed by the data. Hence, we can easily conclude that the popular domain of SA is an amalgamation of several popular areas like affective computing, natural language processing, and computer vision. This thesis focuses on extracting sentiments and emotions from different modalities, which include: images uploaded by users from popular social networking websites, identifying the popular emotions evoked in the movie video trailers, extracting sentiments from text-based tweets posted on Twitter, and from various multimodal data (images and text).

The fundamental architecture of SA, as depicted in Figure 1.1, includes the following steps: Data preprocessing, Feature extraction, and classification. As discussed above, the input for the SA system is available in various forms: image, text, video, etc. Hence, different preprocessing and feature extraction techniques are applied to different modalities of data. For textual data, some of the popular preprocessing methods include lowercase conversion, removing special characters and stopwords, number to word conversion, stemming, and lemmatization. For visual data, image resizing, rotation, denoising, rescaling, zooming, cropping are some of the preprocessing techniques.

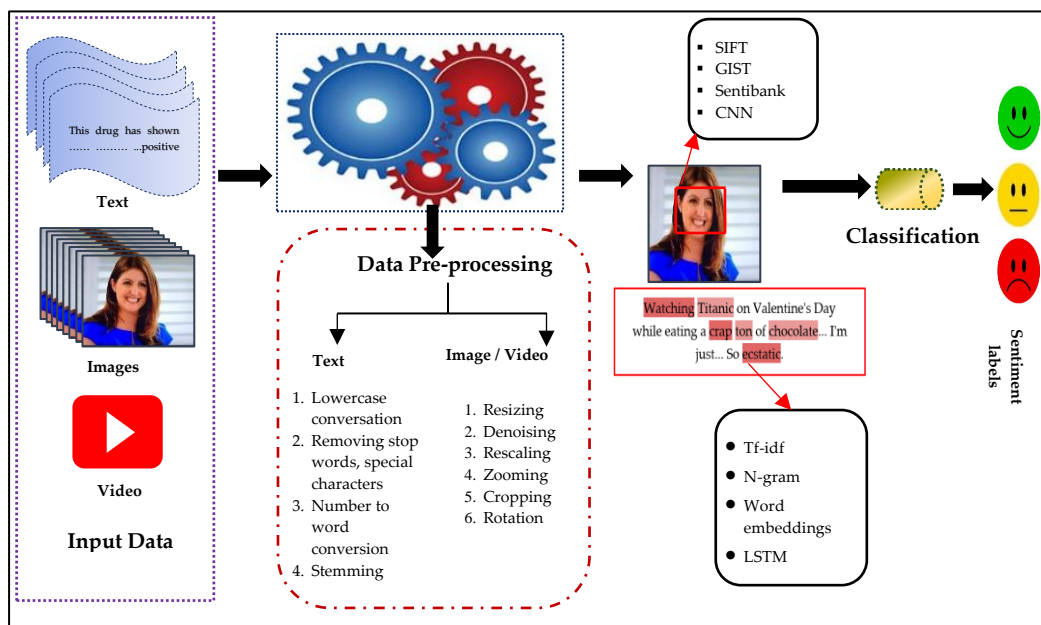


Figure 1.1 Basic architecture of Sentiment Analysis

Feature engineering is one of the most crucial step as it directly impacts the

performance of the system. Feature extraction helps in capturing the relevant features from the input data to increase the predictive power of the systems. This step is categorized into two parts: traditional handcrafted features and Deep features. The conventional approach for feature extraction has focused on designing handcrafted features, which was a very time-consuming and tedious process. According to Twitter, approximately 6000 Tweets are generated per second. With such a huge amount of data, the traditional handcrafted based approaches fail to perform. With the great feats of deep learning, especially in computer vision, researchers are motivated to apply them to SA. The significant advantage of deep based models includes handling the massive amount of data and learning the relevant features from them automatically. Since the area of VSA has a strong relationship with computer vision, the deep based pre-trained models, which have shown great success in computer vision, can also be applied for VSA. This can be achieved by fine-tuning them on the target domain.

The traditional handcrafted features for text include lexicons based features to compute the word polarity from the dictionary or lexicons, Term-Frequency Inverse Document Frequency (tf-idf) extracts the essential words from the vocabulary, fetching words based on their frequency or count, and the presence of n-grams like bi-grams or tri-grams. In images or videos, handcrafted low-level features include local or global color histograms, scale-invariant feature transform (SIFT) based on bag of visual words, and mid-level approaches that provide a mid-level description for the images like SentiBank, SentiBank. On the other hand, the deep based features can capture the essential and dynamic features from the input data automatically. The pre-trained word embeddings like Glove, Word2vec are highly popular for text-based processing. Deep based models like convolutional neural networks (CNN), Long short-term memory networks (LSTM), Bi-directional Long-short term memory (Bi-LSTM), Gated Recurrent Units (GRU) are widely applied for SA. The promising development of CNN based pre-trained models for image processing is extensively being explored in the VSA domain. Finally, classification is performed using machine learning classifiers like Support Vector Machine (SVM), Naïve Bayes, K-Nearest Neighbor (K-NN), and softmax.

## **1.2 Motivation**

The significant motivation for SA can be attributed to its wide range of applications in different fields. One of the noteworthy application involves extracting opinions or sentiments from the medical narratives in the health domain. In medical texts, SA plays a crucial role by diagnosing a particular medical disease, justifying the status of the patient's health, or evaluating the effectiveness of the treatment. The health status of a person may be reflected in terms of sentiment, which can be bad, good, or normal at a given point in time [9]. According to the PwC Health Research Institute social media consumer survey, people are using social media for a variety of healthcare-related activities. Moreover, they are actively sharing negative and positive health experiences via social media, which includes experience with medication or treatment received, care received at the hospital, expenses incurred at medical facility or health insurance coverage, etc.

The growing multimedia content on social media in the form of images and videos creates promising avenues for vision-based SA. With the rapid development of smartphones, people are now exposed to this new channel to express themselves by uploading their real-time photos or videos on the web. They feel more convenient to upload their pictures for expressing themselves, rather than typing the simple plain text. Mining such information becomes crucial for developing personal recommendation systems and understanding human behavior.

The rise of video content on social media has seen a tremendous increase over the years. The videos are referred to as the “king” or the most popular modality on social platforms. This is because people are spending a considerable amount of time watching videos. According to the Wibbitz survey, around 74% of US users watch video content on YouTube, approx 60% on Facebook, and 26% on Instagram. Facebook data reveals that 100 million hours of videos are being watched on social media every day. These figures indicate how videos and images create more impact on the human cognitive system.

All these applications motivate us to design a robust SA framework that can

identify and extract the vital sentiments expressed in the data. The earlier research in this area has majorly focused on text-based SA. Much of the progress has already been made in this area. However, the domain of vision-based SA remains unexplored, and its potential applications inspire us to explore this field. Therefore, we design frameworks for extracting and analyzing the sentiments from different modalities like images posted on social media, movie trailer videos, text-based tweets on Twitter, and multimodal content in the form of images and text found on various social media websites.

### **1.3 Major Challenges**

The popular area of SA faces several challenges while designing robust systems. For text-based SA, the medical domain has not yet achieved significant results as compared to the field of non-medical social media. The prime reason is that the sentiments are expressed in terms of (mainly) adjectives on social media, whereas medical texts contain less opinionated words [9]. The clinical documents like discharge summary, radiology report comprises of implicit terms that describe the health condition or symptoms of the patients. Thus, context-specific cues must be designed to handle these elusive and implicit sentiments expressed in the texts. Negation handling and developing domain-specific lexicons remains another challenging task in this field. Further, the lack of medical datasets poses yet another challenge in this area.

On the other hand, the vision-based SA is more challenging as compared to the text-based analysis. This is because there exists a huge “semantic gap” between the low-level visual features of the image and the high-level sentiment labels. The human brain is highly subjective in the visual recognition process. Past studies have shown that many researchers have built successful models to reduce this semantic gap [27] [28]. They have used CNN based models to capture the sentiments effectively. However, visual sentiments can sometimes be expressed only from specific regions of the images. But deep based models like CNN learns the discriminative features from the entire image [29]. Therefore, it becomes necessary to design a mechanism that could successfully identify the prominent sentiment-rich regions in the image. The redundant

areas could sometimes induce unnecessary noise for the classification process.

Another challenge lies in the fact that the pictures uploaded by the users on social media are not only restricted to facial or body expressions but also include several objects like people uploading the view of their garden or image of a house fire, etc. Further, there exists a close relationship between emotions and sentiments [30]. The field of SA and affective analysis are strongly related to each other [25]. Emotions can be used to drive the SA process. The pictures on the web also convey strong emotions. However, a proper distinction must be made between the emotion intended by the publisher (induced emotion) and the emotion invoked on the viewer [31]. Hence, robust features must be designed, which could address the above problems by taking advantage of the interconnections between emotions and sentiments.

Finally, all the above image-related challenges also drill down to the video-based SA, which is also not explored much in the literature for extracting the sentiments and emotions. Since social media users also express themselves through multiple modalities (Eg: Posting an image with a caption on Instagram), SA is now becoming famous for analyzing the multimodal data (images and text) simultaneously. However, each modality has its individual characteristics and is expressed differently by the human cognitive system [7]. Hence, it isn't easy to deal with such heterogeneous content for multimodal analysis. Therefore, the correlations between the image and text descriptions need to be captured effectively to bridge this gap for multimodal sentiment classification.

## **1.4 Problem Statement**

The above challenges motivated us to develop effective frameworks to extract the sentiments and emotions from visual data (images and videos) and textual data and advance from unimodal analysis to multimodal analysis by applying the popular deep-based methods to address the practical issues in them. Hence, we frame the following problem statements to accomplish the practical challenges:

- ✓ An effective framework is required to handle the high level of abstraction and

subjectivity in the human recognition process. This framework would help in extracting the visual sentiments and emotions from the real-world image datasets.

- ✓ Since movie genres are abstract and elusive in nature and a trailer can be associated with multiple genres, a novel affect-based framework is required to handle the enigmatic nature of emotions. The framework would utilize the different emotions to classify a movie trailer into multiple genres.
- ✓ Multimodal sentiment classification is a challenging area that explores the fine-grained correlation between different modalities to generate the final sentiment class. Hence, a multilevel attention mechanism can be employed to enhance the multimodal learning process to extract sentiments from crucial multimodal data regions.
- ✓ Finally, the application sentiment analysis in the medical domain needs to be explored to demonstrate how sentiments can reflect the behavior patterns of people from all over the world towards the outbreak of any disease or a pandemic.

## 1.5 Methods and Results Overview

The algorithms developed for solving SA challenges are evaluated on popular real-world datasets from various social media platforms like Twitter, Flickr, Getty images, and Instagram for visual and multimodal SA. For affective-video classification, we make considerable efforts in developing EmoGDB dataset, which contains 100 Bollywood movie trailers in six popular and distinct genres: Action, Comedy, Drama, Horror, Romance, Thriller. The entire dataset is labeled with six induced emotions: Anger, Fear, Happy, Neutral, Sad, Surprise corresponding to every movie genre. Similarly, to analyze the impact of COVID-19 on people's lives, we developed the *COVID-19 Sentiment dataset* by crawling and downloading the tweets from the famous microblogging site, Twitter, and labeling them according to the sentiments belonging to the positive, negative, and neutral sentiment classes. We download and analyze the tweets of the people belonging to the USA, Brazil, India, Russia, and South Africa, starting from January 1, 2020, to June 7, 2020. The experimental results are as follows:

- ✓ Transfer learning has shown great results for extracting visual sentiments and emotions. The CNN helps in learning the spatial hierarchies of image features, and attention networks focus on crucial sentiment-rich, local regions of the image.
- ✓ The combination of Inception V4, Bi-LSTM, and LSTM layers is utilized to obtain the discriminative and comprehensive high-level features from the movie trailers to classify them into multiple genres and map the human emotions to the structure of the movie trailers effectively.
- ✓ The attention mechanism introduced at multiple levels in the form of a bi-attentive visual map, semantic attention, and self-attention helps in extracting the crucial features. The image-level features are used to highlight the sentimental words, which enhances the performance for multimodal sentiment classification.
- ✓ Experimental results validated that word level and sentence level attention can capture the word structure from a sentence and sentence structure from a document to extract the sentiments from text-based tweets posted on Twitter.

## **1.6 Significance of the Study**

SA is being used for commercial purposes, politics, healthcare, human-machine interactions, multimedia analytics, etc. This makes SA a popular field having ample applications. Our study aims to contribute to this field by applying the vital outcomes of the research to real-world problems. We proposed a novel idea in affect-based video classification, where we contribute by successfully establishing and validating the relationship between psychology and cinematography. This is achieved by designing a novel framework that could automatically classify the movie trailer of different cinemas like Hollywood, Bollywood, Cinema of Japan, Denmark, South Korea, etc., into multiple genres, efficiently eliminating the need to watch them explicitly.

During our research work, the outbreak of the COVID-19 pandemic (popularly referred to as coronavirus) badly hit the entire world and infected millions of people worldwide. Since SA is popularly applied to extract the opinions of people on social



media; we use it to study the sentiments of the people from the top five worst affected countries by the virus, namely the USA, Brazil, India, Russia, and South Africa. We draw several conclusions related to the social and economic condition of the people. This analysis can serve as feedback to the government agencies regarding the mitigation plans taken by them. Further, it may also guide the future planning of the public health agencies in case of any such outbreak.

## **1.7 Dissertation Outline**

The dissertation consists of five chapters, which are organized as follows:

Chapter 2 is dedicated to the literature review, where the existing state-of-the-arts for sentiment classification are reviewed for textual and visual (images and videos) modalities. The prevalent approaches in each of the modalities are grouped into a taxonomy, which helped in identifying the research gaps in this area. Finally, the research objectives are also briefly addressed.

Chapter 3 focuses on understanding the sentiments and emotions from the visual data. The first model is built for extracting the sentiments from the images posted by the users on various social media sites by applying the CNN architecture, followed by a residual attention block, which focuses on crucial sentiment-rich, local image regions. The second model aims to study the emotions induced while watching a movie trailer and utilize them for performing movie genre classification. We apply the unique combination of Inception V4, Bi-LSTM, and LSTM layers to obtain discriminative and comprehensive high-level features from the trailers of the movie. We also developed EmoGDB dataset, which helped to model the relationship between emotions induced in the movie trailer and the movie genre.

Chapter 4 aims to extract multimodal features by applying a multi-level attention model in the form of channel attention, region attention, semantic attention, and self-attention to explore the correlation between image and text modalities for improving multimodal learning. The bi-attentive visual features are obtained by exploiting the channel attention and spatial attention on the visual data.

In Chapter 5, we discuss the application of sentiment analysis for analyzing the opinion of people posted on Twitter about the COVID-19 pandemic from the top five worst affected countries by the virus. We develop a language-independent framework that explores the hierarchical structure of a document by extracting the word-level and sentence-level features for sentiment classification.

Finally, in Chapter 6, we summarize the conclusions inferred from our research work and highlight the potential future work in this area.

# Chapter-2 Literature Review

This chapter discusses the previous work and state-of-the-art in the field of SA. The research in SA began in early 2000. However, the term *sentiment analysis* first appeared in 2003 by [32] [33]. Hence, from the past 20 years, extensive researches have been done in this area. To analyze its progress over the years and study the significant developments achieved so far, the literature is broadly classified into handcrafted and automatic generated features corresponding to the different modalities. Hence, we discuss the significant works in textual and visual modalities and advance from the unimodal analysis to multimodal analysis. Figure 2.1 shows the taxonomy of SA.

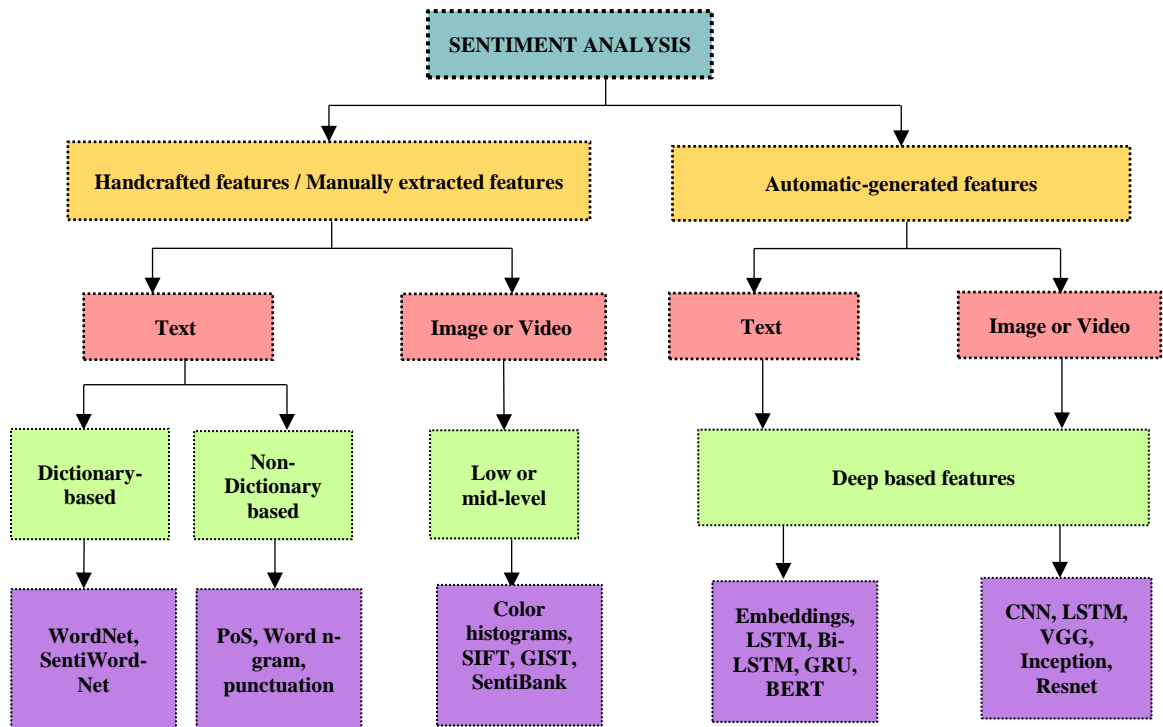


Figure 2.1 Taxonomy of Sentiment Analysis

## 2.1 Textual Sentiment Analysis

The early works in the area of SA have majorly focused on textual modality. With the advent and progressive increase of web technology and the exchange of public ideas on the Internet, research for textual SA increased exponentially [34]. Several data processing approaches were proposed to extract the sentiment of people from reviews [35], blogs [36], and micro-blogging sites [37]. Text-based SA approaches can be

classified into handcrafted feature-based and deep-based methods.

### **2.1.1 Sentiment analysis tasks**

The SA tasks can be categorized into two parts [8]: core (or major) task referred to as the basic SA tasks, and the sub-tasks called sub-categories of the major tasks. The core SA tasks include document-level sentiment classification, sentence-level sentiment classification, aspect-level sentiment classification, and sub-tasks, includes multi-domain sentiment classification and multimodal sentiment classification [38]. The various tasks and sub-tasks of SA are discussed in the subsequent sections.

#### **2.1.1.1 Document-level sentiment classification**

This classification level takes the whole document as a primary unit of information focusing on one topic or object. The document is further categorized into positive polarity or negative polarity. Thus, an overall sentiment of text can be generated. Yang *et al.* [39] proposed a hierarchical attention network model that focuses on vital content for constructing the document representation. Experimental results on six popular text-based reviews demonstrate that the proposed model outperformed the state-of-the-art results by a significant margin. It can capture insights into the structure of the document. A major challenge in document-level sentiment classification is to model longer texts for generating semantic relations between sentences. This problem was handled by Rao *et al.* [40], by developing an LSTM model with sentence representation (SR-LSTM) in which the first layer learns the sentence vectors, and the second layer encodes the relations between the sentences.

A hybrid approach of Restricted Boltzmann machine (RBM) and Probabilistic Neural Network (PNN) was proposed by Ghosh *et al.* [41] in which RBM was used for dimensionality reduction, and PNN performs sentiment classification. The experiment was conducted in four steps: Initially, multi-domain data was collected containing reviews on Movies, Books, DVDs, Electronics, and Kitchen appliances. Next, the data was pre-processed using tokenization, stemming, and stop-word removal. The third step includes dimensionality reduction on the dataset using RBM, and finally, PNN was

used for binary sentiment classification. The proposed approach gave better results on the five datasets compared to the state-of-the-art methods.

### **2.1.1.2 Sentence-level sentiment classification**

The disadvantage of document-level sentiment classification is that it is difficult to extract the different polarity or sentiment about distinct entities separately. Hence, in sentence-level sentiment classification, a sentence is classified into subjective or objective type. A subjective statement expresses an opinion towards an entity. For example, “*I got a beautiful bag*”, signifies positive polarity about the bag. Hence, it is considered as a subjective statement that can be further classified into different polarities. On the other hand, factual statements are termed as objective statements. A statement like “*The bottle is blue in color*”, displays no sentiment, so it is categorized as an objective statement.

Zhao *et al.* [42] proposed a framework called Weakly-supervised Deep Embedding (WDE), which employs review ratings to train a sentiment classifier. They used CNN for constructing WDE-CNN and LSTM for constructing WDE-LSTM to extract feature vectors from review sentences. The model was evaluated on the Amazon dataset from three domains: digital cameras, cell phones, and laptops. The accuracy obtained on the WDE-CNN model was 87.7%, and on WDE-LSTM model was 87.9%, which shows that deep learning models give the highest accuracy compared to baseline models. Xiong *et al.* [43] developed a model called Multi-level Sentiment-enriched Word Embedding (MSWE), which used a Multi-layer perceptron (MLP) to model word-level sentiment information and CNN to model tweet level sentiment information. The model also learned sentiment-specific word embeddings and SVM for the classification. It was evaluated on the SemEval2013 dataset and Context-Sensitive Twitter (CST) dataset, the benchmark datasets for sentiment classification.

### **2.1.1.3 Aspect-level sentiment classification**

Aspect level SA is commonly called feature-based SA or entity-based SA. This SA task includes identifying features or aspects in a sentence (which is a user-generated review

of an entity) and categorizing the features as positive or negative. The sentiment-target pairs were first identified, then they are classified into different polarities, and finally, sentiment values for every aspect is clubbed. Peng *et al.* [44] studied the Chinese aspect targets at three granularity levels: *radical*, *character*, and *word* level by proposing a model called Aspect Target Sequence Model for Single Granularity (ATSM-S). The previous work was related to processing only one aspect at a time, so they addressed this issue and presented an approach to process two aspects by focusing on the aspect target itself. Recently, attention-based LSTM mechanisms are being used for aspect-based SA. Wang *et al.* [45] proposed an attention-based LSTM model, focusing on different parts of a sentence when various aspects are concerned. The attention weights were computed by concatenating the aspect vector into the sentence hidden representation (AE-LSTM model) or by appending aspect vector embedding into each word input vector (ATAE-LSTM model). Experimental results demonstrate that both the proposed models achieved superior performance over the baseline models, which shows that attention-based LSTM models boost the performance of aspect-based SA models.

Yu *et al.* [46] proposed a framework using Bi-LSTM and multi-layer attention networks for aspect and opinion terms extraction. Al-Smadi *et al.* [47] proposed a bi-LSTM with CRF model for aspect opinion target expressions (OTEs) extraction, along with aspect-based LSTM, where the aspect OTEs are treated as attention expressions for aspect sentiment polarity classification. Ma *et al.* [48] proposed a two-step attention architecture that attends the target expression and the whole sentence. The author also applied extended LSTM to utilize external knowledge for developing a common-sense system for target aspect-based SA. The initial systems could not model different aspects in a sentence and do not explore the explicit position context of words. Hence, Ma *et al.* [49] developed a two-stage approach that can handle the above problems. In Stage-1, the position attention model was introduced for modeling the aspects and its neighboring context words. In Stage-2, multiple aspect terms within a sentence were modeled simultaneously. The most recent approach was proposed by Yang *et al.* [50], who replaced the conventional attention models with coattention mechanism by introducing a Coattention-LSTM network. The network could simultaneously model

the context-level and target-level attention by learning the non-linear representations of the target and context simultaneously. Thus, the proposed model can extract more effective sentiment features for aspect-based SA.

#### **2.1.1.4 Multi-domain sentiment classification**

The word domain is referred to as a set of documents that are related to a specific topic. Multi-domain sentiment classification focuses on transferring information from one domain to the next domain. The models are first trained in the source domain; the knowledge is then transferred and explored in another domain. Dragoni *et al.* [1] incorporated word embeddings with a deep learning model for implementing a NeuroSent tool to build a multi-domain sentiment model. Yuan *et al.* [51] proposed a Domain Attention Model (DAM) for modeling the feature-level tasks using attention mechanisms for multi-domain sentiment classification. DAM is composed of two modules: domain module and sentiment module. The domain module predicts the domain in which the text belongs using Bi-LSTM layer, and sentiment module selects the critical features related to the domain using another Bi-LSTM layer with an attention mechanism. The vector thus obtained from the sentiment module is fed into a softmax classifier to predict the polarity of the text. The author used Amazon multi-domain dataset containing reviews from four domains, and Sanders Twitter Sentiment dataset containing tweets about four different IT companies. The proposed model was compared with traditional machine learning approaches, and results show that the model performed well for multi-domain sentiment classification.

#### **2.1.1.5 Multimodal sentiment classification**

Different people express their sentiments or opinions in different ways. Earlier, the text was considered as the primary medium to express an opinion. This is known as a unimodal approach. With the advancement of technology and science, people are now shifting towards visual (videos, images, or clips) and audio (speech) modalities to express their sentiments. Combining or fusing more than one modality for detecting the opinion is known as multimodal SA. Hence, researchers are now focusing on this direction for improving the sentiment classification process.

Chen et al. [52] proposed a Weakly-Supervised Multi-modal Deep Learning (WS-MDL) model to predict multimodal sentiments for tweets. The model applied CNN and Dynamic CNN (DCNN) to calculate multimodal prediction scores and sentiment consistency scores. Due to the enormous data available on social media in different forms like videos, audios, photos for expressing sentiment on social media platforms, the conventional approach for text-based SA was progressed into compound models of multimodal SA. Hence, mining the opinions expressed in different modalities became a crucial approach. Poria *et al.* [53] proposed a novel methodology for merging the affective information extracted from audio, visual, and textual modalities. They discussed how different modalities were combined together to improve the overall SA process. Poria *et al.* [54] explored three deep learning architectures for unimodal, bimodal, and multimodal (trimodal) sentiment classification. The experimental results showed that bimodal and tri-modal models had shown better accuracy than unimodal models, which shows the importance of using features from all the modalities for enhancing the performance of SA models.

### **2.1.2 Handcrafted Feature-based Text Sentiment Analysis**

Research in the field of SA is taking place for several years. Initially, handcrafted features were used for various classification tasks. Some examples of handcrafted features are shown in Table 2.1. Lexicon based methods use handcrafted features and depend on sentiment lexicons, which are the collection of lexical units and their sentiment orientation. On the other hand, machine-learned features can be categorized into traditional machine learning-based approaches and deep learning-based approaches. Machine learning-based methods include Support Vector Machine (SVM), Naïve Bayes (NB), Maximum Entropy (ME), Decision tree learning, and Random Forests. They are further categorized into supervised and unsupervised learning methods. A significant goal of SA is to classify and analyze the reviews related to products, hotels, online booking sites, e-commerce sites, social media, etc. Haque *et al.* [55] used Amazon product reviews in three domains: ‘cell phone and accessories’, ‘musical’, and ‘electronics’. They have classified the sentiments via Linear SVM, Multinomial Naïve Bayes, Stochastic Gradient Descent, Random Forest, Logistic



regression, and Decision Tree. The best classification results were obtained by SVM with an accuracy of 94.02% on the musical domain. Singla *et al.* [56] have performed SA on Amazon mobile phone reviews, and in their study, they have categorized text into positive and negative polarity and have also included sentiments of anger, anticipation, fear, joy, sadness, disgust, surprise, and trust. The classification is done through SVM resulting in an accuracy of 84.85%. Moreover, the Samsung brand received the most positive feedback from customers. These results are useful for manufacturers as they can work on the feedback to improve the quality of a product.

Singh *et al.* [57] proposed an approach to evaluate the effect of demonetization on people worldwide. They applied Valence Aware Dictionary and Sentiment Reasoner (VADER), for SA on tweets extracted from Twitter. They also performed a retweet analysis in which they predicted the tweet, which could be retweeted again by using various machine learning algorithms. Experimental results show that SVM obtained finest accuracy in the case of unigram. In contrast, unigram-bigram feature extraction and Classification and Regression Trees (CART) gave good results for bigram feature extraction. Stock market investment constitutes a dominant part of the economy of any country. Hence, a detailed market analysis becomes a crucial part of investing money in stock market. Bhardwaj *et al.* [58], developed a system that fetches Sensex and Nifty's live data values, which serves as an essential stock market indicator. Pre-processing and feature selection are also applied to the data, followed by the SA task to get stock market status. Rao *et al.* [59] analyzed the relationship between the sentiment of tweets from DJIA (Dow Jones Industrial Average), NASDAQ-100, and 13 other companies and its impact on market performance. They have used Naïve Bayesian classifier for sentiment classification. The results show that the polarity of sentiments had a substantial effect on the stock price movement. Moreover, the previous week's sentiments strongly impact the coming week's opening and closing stock values. Xu *et al.* [60] proposed a method in which SVM was used for the two-stage sentiment classification process, and the dataset consisted of Tweets from the StockTwits website. In the first stage, neutral and polarized classification was performed, followed by binary (positive and negative) classification. Then a schema was designed to analyze the labeled tweets. The experimental results showed that the user's overnight activity on

StockTwits had a positive correlation to the next day's stock trading volume. It was also found that the collective sentiment of after-hours (4:00 pm–9:30 am) influenced the stock movement direction of the following day.

**Table 2.1 List of handcrafted features**

| Handcrafted features                         | Description   |
|--|---|
| PoS (Part-of-speech) [61] [62] [63]          | PoS reads the input text and label each word with parts of speech such as nouns, adverbs, verbs, adjectives, etc.   |
| Word n-gram, character n-gram [61] [64] [65] | Word n-grams is a combination of a contiguous sequence of $n$ words in the input text sequence. They are usually categorized into unigram (size 1), bigram (size 2), trigram (size 3). Similarly, character $n$ -grams are the combination of a continuous sequence of $n$ characters in the input text sequence. |
| Hashtags [43] [63]                           | They are majorly used for Twitter SA by counting the positive and negative hashtag tokens in the tweet.   |
| Emoticons [43] [63]                          | They are majorly used for Twitter SA by counting the positive and negative emoticons in the tweet.  |
| Punctuation [62] [66]                        | Occurrence of punctuation marks e.g., question mark (?), exclamation mark (!), colon (::) are considered as features.   |
| Negations [62] [67] [68]                     | Negation words like don't, didn't, not, etc. can change the polarity of the sentence from positive to negative and vice versa   |
| Sentiment lexicon [62] [67]                  | Sentiment lexicon contains a list of words which expresses a positive or negative sentiment. The final sentiment is calculated by averaging the #positive tokens and #negative tokens. e.g., MPQA lexicon [69], SentiwordNet 3.0 [70], Sentiment140 lexicon [69], NRC Hashtag Sentiment Lexicon [71] etc.         |

### 2.1.3 Deep based Textual Sentiment Analysis

In the past years, these traditional approaches have achieved good results. However, they use handcrafted features in the feature-engineering process, which is a time-consuming and tedious process. Further, these features are not able to generalize well on other domains or areas. Later, the researchers realize that extracting the sentiments from today's data requires an in-depth understanding of the text. Hence, deep-based approaches were used to learn the complex features from the data, thus enhancing the SA process. It was observed that the deep-based methods were able to outperform the traditional methods significantly. However, these methods require lots of data and computing resources for training to ensure that they deliver the desired results.

Deep networks have shown tremendous success in computer vision [72], speech recognition [73] and human-computer interaction [74] over the past. However, in natural language processing, deep learning models need word embeddings as input

features. Word embeddings is a popular technique that is developed to represent the vocabulary of a document. They map the word into its equivalent dense, low-dimensionality vector representations. Earlier, frequency-based embeddings like Count Vectorizer [75], Bag-of-Words [76], and Term Frequency-Inverse Document Frequency [77], were applied for sentiment classification of textual data. However, these techniques provide limited word representations and failed to capture the contextual information from data. Hence, prediction-based approaches were popularly applied for sentiment classification. They would map the semantic similar words closer to each other in the word embedding space, thus preserving the relationship among words.

Word2Vec [78] applies the shallow neural network to generate the word embeddings using CBOW (Continuous Bag of Words) and Skip-gram model. The former predicts the target word from the surrounding context words, whereas the latter predicts the neighboring context words from the given word. Severyn *et al.* [79] applied word2vec model for training word embeddings on 50M tweets and used them as an input for the deep learning model. Fu *et al.* [80] applied Word2Vec to learn the vector representation for phrases and sentences on the Wikipedia dataset for English and Chinese languages. The vectors were used as inputs for Recursive autoencoders for sentence-level sentiment classification. However, in order to get useful embeddings for a word, the model must be trained on a vast corpus, which is a time-consuming process. Thus, pre-trained word embedding models were developed, which uses previous knowledge of a task to solve new tasks.

Kim *et al.* [81] applied pre-trained Word2Vec, trained on 100 billion words from Google News as an input to the CNNs. The results tell that these vectors can serve as “universal” feature extractors for NLP tasks using deep learning. Pennington *et al.* [82] developed Glove, which stands for “Global Vectors” to incorporate the global statistics for generating the word vectors. The prime advantage of Glove vectors is that they require less time for training, thus faster than Word2Vec. On the other hand, instead of learning the embedding of the full word, [83] developed char2vec, which gives the embedding of n-grams by learning the correlations between the string of characters and

the meaning of a word. This helps in generating the embeddings for rare words or words which does not belong to the vocabulary. Therefore, word representation for unseen words can progress drastically, improving the overall sentiment classification process.

One of the most prominent works which used CNN for sentiment classification is [84]. The authors performed a sensitivity analysis on a one-layer CNN model to find the optimal hyperparameters for classification. They explored how the performance of a model can be affected by changing its configuration (hyperparameters, filter size, regularization parameters, etc.). The input sentence of length  $s$  was tokenized and converted into a sentence matrix by following the work of [85], which applied a look-up table concept to generate the matrix. The rows of the matrix are the vectorized representation obtained using the word2vec method. The dimensionality of the matrix is  $s * d$ , where  $d$  represents the dimensionality of word vectors. Hence, the sentence matrix can now be treated as an input image on which convolution is performed using linear filters to generate the feature maps. The height of the filter is referred to as the region size of the filter. For pooling operation, 1-max pooling is performed on each feature map, followed by softmax classifier, which gave the probabilistic distribution over the number of output classes.

Recursive Neural Network (RecNN) belongs to the category of the deep networks, which takes the structural representation of a sentence in the form of a parse tree with word vector representations at leaves, and recursively generates parent representations in a bottom-up manner. Socher *et al.* [86] discussed an approach for RecNNs by introducing a new corpus called Stanford Sentiment Treebank (SSTb), which will help in analyzing the compositional effects of sentiments expressed in any language. They applied the Recursive Neural Tensor Network (RNTN) to capture these compositional effects and increase interactions between the input vectors. Apart from these models, Recurrent Neural Networks (RNN), which is a variant of RecNN, have shown great success for modeling the sequential data. The primary difference between RNN and RecNN is that, unlike RecNN, RNN considers the time factor for processing the elements in a sequence. Thus, output in RNNs depends not only on the present input but also on the output computed from the previously hidden state of a network. RNNs

stores the internal states of the inputs by processing each word in a sentence recurrently. Hence, to predict the next word in a sentence, RNN will store all the previous words and the relations between them. The commonly used variants of RNNs are LSTM, GRU, bi-directional RNNs, and deep RNNs. LSTM is one of the most popular variants of RNN, which possesses the capability to handle the vanishing gradient problem in standard RNN and can catch long-term dependencies. This makes them more powerful and flexible.

Researchers are also combining various deep learning techniques for sentiment classification. Hassan *et al.* [87] proposed an architecture called ConvLstm, which again combined CNN and LSTM for classifying short texts on the top of word2vec. GRUs are considered as LSTM without an output gate. They deal with two kinds of gate: update gate and reset gate. Attention-based GRU networks are also applied to various SA tasks like target based sentiment classification [88]. Another category of deep networks includes deep belief networks (DBNs), which emanate under the type of unsupervised pre-trained networks, and includes autoencoders and Generative Adversarial Networks. DBNs are composed of multiple layers of unsupervised networks like Restricted Boltzmann Machines (RBMs) or autoencoders [89] [90]. The traditional RNN approaches capture irrelevant information in the piece of information-rich text. Hence, the attention mechanism was introduced, which was inspired by the visual attention mechanism found in humans. It decides which part of the text should be focused on, rather than encoding the full sentence length. SA using attention-based networks was applied in [51] [67] [91] [92].

#### **2.1.4 Sentiment Analysis in Healthcare**

The facets of sentiments in medicine are broadly categorized into [9]: capturing a patient's health status (good/bad/no change), diagnosing the certainty or severity of a disease, and the effectiveness of the treatment or drug. The early work in this area focuses on in-built lexicons or machine learning-based approaches for extracting the sentiments. Yang *et al.* [93] proposed a framework based on the weighted scheme Latent Dirichlet Allocation (LDA) to analyze the health-related information regarding

breast cancer. The medical topics were clustered together to perform SA using the AFINN lexicon. Each topic was associated with positive and negative sentiment terms. The framework can benefit other users to seek advice or take necessary precautions in case they experience similar symptoms. Sabra *et al.* [94] proposed a framework based on semantic extraction and sentiment assessment of the risk factors to predict the diagnosis of Venous thromboembolism disorder. The clinical narratives were parsed to extract the risk factors with the help of MetaMap. These risk factors were stored in a dictionary, and semantic-rich rules were derived from weighting the risk factors according to the severity. Finally, the polarity of the adjectives and adverbs were extracted with the help of SentiWordNet lexicon. The framework achieved 54.5% precision and 85.7% recall to identify a patient's likelihood to develop VTE early and start the treatment as soon as possible.

Several clinical trials are necessary before releasing any drug to the market. Still, discovering all the side effects of the drugs is a challenging task. Moh *et al.* [95] developed an architecture to study the side effects of five crucial drugs by capturing the tweets from Twitter. The sentiment scores were extracted using several lexicons like SentiWordNet, AFINN, etc. Similarly, Jiménez-Zafra *et al.* [96] applied machine learning classifiers and lexicon-based approaches to compute the sentiment polarities for the physicians and the drugs. The reviews related to physicians contain informal language, while discussions about drugs contain the combination of colloquial language and specific terms like adverse effects or name of the drug. Korkontzelos *et al.* [97] identified the adverse drug reactions on social media platforms by integrating the sentiment feature. Experimental results show that sentiment features improved the F-score from 72.14% to 73.22%.

Recently, deep based approaches are being explored to extract the health-care sentiments expressed by the people. Limsopatham *et al.* [98] applied CNN and word embeddings on social media messages to medical concepts. Chen *et al.* [99] extracted the emotion features based on the LSTM network for screening the users on perinatal depression. The results show that the method improves the early screening process for perinatal depression. Grisstte *et al.* [100] applied the Bi-LSTM model for identifying

the adverse drug reactions from social media text. Talpada *et al.* [101] studied the impact of Telemedicine on heart attacks and epilepsy. They compared the lexical and semantic methods with the deep based methods. The lexical and semantic methods were based on VADER and TextBlob, whereas LSTM was employed for deep based classification. Experimental results signify that sentiments can be used to obtain the demographic distribution of Telemedicine for heart attacks and epilepsy. Thus, sentiment in the context of medicine shows the impact of a disease or drug on the life of a person over a period of time. This can provide feedback to assess the effectiveness of the treatment or measures taken to combat a disease or a crucial event.

## 2.2 Visual Sentiment Analysis

The majority of research in sentiment analysis has focused on text-based modality. With the increasing number of user-generated images and videos on Twitter, Facebook, Flickr, etc, VSA has extended the traditional text-based sentiment classification process. The VSA aims to extract the sentiments for images and videos. Moreover, the sentiment expressed by visual modalities like emoticons, images, and videos can boost the sentiment extracted by the text modality [102]. Nowadays, visual media are rising as users feel more convenient to express their opinion in the form of emoticons or images. The area of VSA is quite challenging. The work done in this area can be divided into low-level features, mid-level features, and deep based approaches.

### 2.2.1 Low-level feature-based approaches

Earlier efforts for VSA focused on several low-level features. Siersdorfer *et al.* [103] used SIFT-based visual terms with RGB Histogram to get the prominent features from Flickr images. The text metadata embedded in the image expressed the sentiment conveyed by them. A sentiment value was allotted to the image on the basis of textual metadata, which was combined with low-level features for the final classification with SVM. Vonikakis *et al.* [104] proposed a method to generate a slideshow of the family photos by concentrating on the colors of the picture, Gist of the scene, emotions, and the clicked time. Jia *et al.* [105] used Flickr images to analyze the features which reflect





presented a novel approach to convey the affects intended by a publisher from the visuals of images. They applied Sentibank to extract the 1200 Publisher Affect Concepts (PACs) from images and various data mining methods for discovering 446 Viewer Affect Concepts (VACs) from the comments. All the experiments were conducted on Flickr.

### **2.2.3 Deep-based feature-based approaches**

The deep learning-based architectures are increasingly being applied in VSA. You *et al.* [112] fine-tuned CNN on Flickr images. Chen *et al.* [52] used multiple modalities in the form of text, image, and emoticon from microblogs and proposed a weak supervised deep learning model called WS-MDL to focus on numerous modalities. Yang *et al.* [113] employed CNN for learning the sentiments by extracting the global features along with local features from the image. The objectness score was calculated by creating a bounding box around every image. The objectness score, along with the sentiment score, was joined to generate the affective regions. Xiong *et al.* [114] designed a framework based on the pre-trained VGGNet model and group sparse regularization (R-CNNGSR) to obtain an initial sentiment prediction. The sentiment regions were detected by combining the sentimental features with the full image to estimate the image's final sentiment score. This approach can also be applied in the future for multi-label sentiment classification.

#### **2.2.3.1 Visual attention**

Sometimes, images may be noisy with lots of immaterial information that could hinder the model's performance. Attention-based systems were widely used for these problems. Li *et al.* [115] applied a 3D attention model to extract the spatial and temporal features of a video and get the total highlight score for video highlight detection. Zhao *et al.* [116] exploited the LSTM-based attention network for fine-grained classification. The proposed recurrent soft attention framework focused on crucial regions of the image. Recent studies on SA apply attention mechanisms for affective analysis. She *et al.* [117] proposed a weakly supervised coupled network to detect sentiments using the detection branch and classify them using the classification branch. The former

generated the regions which evoke certain emotions. The soft attention map gives the probability of emotion for each part of the image. The final sentiment maps highlight the important part of the image, which serves as crucial information for the sentiment classification process. This information was combined with deep features to fetch the semantic features of the image. Experimental results on several data sets show the efficacy of soft attention systems for classification. However, their approach gave a low precision value in the case of images with smaller objects.

Fan *et al.* [118] developed a novel dataset (EMOd) for capturing the relation between the emotional content and the visual attention mechanism of humans by including eye tracking data and annotating the image context related to scenes, objects, emotions, and semantics. Furthermore, they developed a CNN model that computes 1024-dimensional feature weights of each image. This helped to select the relevant semantic features of each image. The experimental results demonstrate that emotional content in the images strongly attracts human visual attention. Similarly, [119] applied the VGG-19 model along with saliency maps to represent the visual attention in the images. The results show that the negative sentiments were influenced by focal information, whereas the positive sentiments by focal and context-based information. Lee *et al.* [120] proposed two subnetworks that extracted facial features and contextual features separately. The features from both the networks were fused to build a context-based emotion recognition system. Similarly, Bawa *et al.* [121] developed a technique based on two models. The first model extracted the global features, and the second model learned the emotional state of different subjects in an image. The features from both the model were learned using the LSTM model.

## 2.3 Multimodal Sentiment Analysis

With the rapid popularity of smartphone devices, a huge variety of data is generated on social media. This makes SA on multiple modalities a popular field of research. The early works majorly focuses on feature selection based approaches. Baccchi *et al.* [122] applied the continuous bag-of-words (CBOW) model for extracting textual information and denoising autoencoder for getting the robust visual features on Twitter short

messages. Fang *et al.* [123] proposed a probabilistic graphical model to capture the correlation between the textual and visual data of Flickr. Ji *et al.* [7] proposed a hypergraph learning framework that computes the relevance among the textual, visual, and emoticon modalities on Sina Weibo microblogs. Dai *et al.* [124] constructed a structured forest to generate the bag of affective words, which reduces the gap between the low-level features and affective descriptors on the Multilingual Visual Sentiment Ontology dataset.

Due to the powerful performance of deep learning-based approaches, these techniques are increasingly being applied for multimodal SA. Xu *et al.* [125] applied word-level and sentence-level attention for modeling the textual data and the CNN-LSTM approach for extracting the semantic information in images. Chen *et al.* [126] used emoticons as weak labels and learned joint features from the image and textual modalities using CNN and dynamic CNN. A probabilistic graphical model is applied to infer the correlation among the predicted labels of various modalities. Zhao *et al.* [127] experimented on five pre-trained CNN models for extracting the features from images, and word2vec was applied for textual feature extraction. Cosine similarity was used to measure the consistency between the features from both modalities. Finally, the features were merged for the classification. Yu *et al.* [128] proposed a network for entity-level multimodal sentiment classification. They extracted and represented the target entity using the LSTM network, followed by capturing the contextual information using the attention mechanism. Bilinear pooling is applied to capture the interactions among the different modalities.

### **2.3.1 Multimodal Fusion**

Multimodal fusion [129] integrates the features from multiple data sources to predict the final class value. There are mainly three types of fusion strategies: early fusion, late fusion, and intermediate fusion.

Early fusion combines data from the input features of multiple modalities to obtain a single feature vector. Poria *et al.* [130] extracted the visual and textual features through deep based networks and fused them using multiple kernel learning classifier

for sentiment classification. However, early fusion cannot capture the time-synchronicity of different modalities and often results in a high dimensional redundant feature vector. Late fusion refers to a combination of decisions from multiple classifiers, where each classifier is trained on a separate modality. However, late fusion ignores the low-level interaction of the modalities. Xu *et al.* [131] developed a bi-directional attention model, which exploits the correlation between the visual and textual contents simultaneously to fuse the attended visual features with attended textual features via late fusion. In Xu *et al.* [132] the cross-modal relation among the images, text, and social links was explored through multi-level LSTMs. A joint relationship was obtained to learn the inter-modal correlations at different levels.

In deep learning models, intermediate fusion is commonly employed as a deep multimodal fusion strategy, where the input data is changed into a higher-level representation through multiple layers. Huang *et al.* [133] proposed multimodal attentive fusion, which focuses separately on the visual attention model and semantic attention model, followed by intermediate-fusion-based multimodal attention. In intermediate fusion, since the different representations are fused at different depths, this may lead to overfitting where the network fails to model the relationship between each modality. Hence, a careful design approach needs to be followed.

## 2.4 Research Gaps

Based on the literature review, we identify the following research gaps for SA:

- ✓ In VSA, different sentiments can be expressed from the objects of the same class, and different objects can convey the same sentiment (either positive or negative), resulting in high interclass and intraclass variance. Hence, an effective framework must be designed to identify the positive and negative sentiments along with the different types of emotions in visual data on real-world datasets.
- ✓ Affective video content analysis has emerged as one of the most challenging and essential research tasks as it aims to automatically analyze the emotions elicited by videos (Eg: movie trailers). The enigmatic nature of emotions, and the difficulty in bridging the gap between the human affective state and structure of the video, poses

yet another challenge in this area. Therefore, a novel framework is needed to model the relationship between the induced emotions and the movie trailers by obtaining the discriminative and comprehensive high-level features from the movie trailers for classifying them into multiple genres.

- ✓ It has been observed that multimodal SA has attracted increasing attention with broad application prospects. In multimodal learning, most works have focused on simply combining the two modalities without exploring the complicated correlations between them. This resulted in dissatisfying performance for multimodal sentiment classification. Thus, a Deep attention-based network can be designed, which exploits the correlation between image and text modalities to improve multimodal learning using real-world datasets.
- ✓ Recently, the world experienced an outbreak of novel coronavirus, which infected millions of people worldwide. Since sentiments are popularly being used in the medical domain; hence, a language-independent framework can be designed to extract and study people's opinions from different parts of the world. This analysis can serve as feedback to the government agencies regarding the mitigation plans taken by them. Further, it may also guide the future planning of the public health agencies in case of any such outbreak.

## **2.5 Research Objectives**

To address the above research gaps, we have framed the following research objectives:

- ✓ To develop an effective framework for extracting the visual sentiments and emotions from real-world image datasets by learning the spatial hierarchies of image features and focusing on the crucial sentiment-rich, local regions in the image.
- ✓ To design a novel deep affect-based video classification framework for classifying the movie trailers (videos) into multiple genres by reducing the “affective gap” between the low-level visual features and high-level affective contents and successfully establishing a relationship between the field of psychology and cinematography.

- ✓ To propose a deep Multilevel Attention-based framework for exploiting the fine-grained correlations between the image and text modalities from large-scale, real-world multimodal datasets and enhancing the multimodal learning process by applying attention mechanisms at various levels.
- ✓ To apply SA in the medical domain for analyzing the impact of the novel coronavirus on the people by extracting and analyzing the sentiments of the top five most affected countries with the virus through a language-independent framework.

## 2.6 Research Contribution

The overall research contribution of our work can be summarised as follows:

- ✓ An effective residual attention-based deep learning network (RA-DLNet) is designed using the CNN and Residual attention model to classify the visual sentiments. The proposed architecture is a unique integration of CNN for learning the spatial hierarchies of image features and residual attention, which generates attention-aware features that change adaptively as layers go deeper and deeper. Hence, the model can capture different types of attention extensively, which refines the feature maps gradually, thus guiding the feature learning process.
- ✓ A novel deep affect-based movie genre classification framework is developed to obtain discriminative and comprehensive high-level features with a unique combination of Inception V4, Bi-LSTM, and LSTM layers. The literature lacks a proper dataset that could strongly annotate the frames of the movie trailer corresponding to the emotion evoked by it. This would reduce the “affective gap” between the low-level visual features and high-level affective content like human feelings or emotions. Hence, we develop an EmoGDB *dataset*, which focuses on studying the relationship between psychology and cinematography.
- ✓ A Deep Multi-level Attentive network (DMLANet) is proposed, which generates the discriminative features from the visual and textual descriptions. The attention mechanism is applied at multiple image levels to generate the channel attended and spatial attended features in order to get a bi-attentive feature map. The network

captures the complicated correlation between the images and text by developing a joint multimodal learning strategy that focuses on the crucial text-based features based on the attended visual features.

- ✓ A deep Multilevel Attention-based ConvBiGRU (MACBiG-Net) framework is presented for classifying the opinions of the people from the top five worst affected countries by the coronavirus. The framework is language-independent and can classify the tweets belonging to multiple languages like Hindi, Japanese, Arabic, Spanish, Urdu, etc. The *COVID-19 Sentiment dataset* is also developed by crawling and downloading the tweets from the famous Twitter and labeling them according to the sentiments belonging to the positive, negative, and neutral sentiment classes.

# Chapter-3 Visual Sentiment Analysis and Understanding

---

This chapter discusses two approaches based on images and video modalities. The first approach handles the challenges faced in VSA by performing extensive experiments on challenging real-world datasets. The second approach performs affect-based movie genre classification by utilizing the emotions evoked while watching a movie trailer. For this, we create an EmoGDB dataset, primarily focused on Bollywood movie trailers and annotated with popular movie genres and emotions.

## 3.1 A deep learning architecture of RA-DLNet for visual sentiment analysis

Visual media has become one of the most potent means of conveying opinions or sentiments on the web. Millions of photos are being uploaded by the people on famous social networking sites for expressing themselves. The area of VSA is abstract in nature due to the high level of biasing in the human recognition process. This work proposes a residual attention-based deep learning network (RA-DLNet), which examines the problem of VSA. We aim to learn the spatial hierarchies of image features using CNN. Since the local regions in the image convey significant sentiments, we apply residual attention model, which focuses on crucial sentiment-rich, local regions in the image. The significant contribution of this work also includes an exhaustive analysis of seven popular CNN-based architectures such as VGG-16, VGG-19, Inception-Resnet-V2, Inception-V3, ResNet-50, Xception, and NASNet. The impact of fine-tuning on these CNN variants is demonstrated in VSA domain. The performance of the proposed RA-DLNet architecture is evaluated by performing extensive experiments on eight publically available benchmark datasets: Twitter I (3-agree, 4-agree, 5-agree), Twitter II, Multimodal Corpus of Sentiment Intensity (CMU-MOSI), CMU Multimodal Opinion Sentiment and Emotion Intensity (CMU-MOSEI), Flickr, and Instagram for binary sentiment classification, and on Artphoto, Flickr and Instagram (F&I) dataset for multi-class emotion classification in terms of accuracy and F1 scores. The



comparison of accuracy with similar state-of-the-art exhibits the superiority of the proposed work.

### 3.1.1 Proposed Methodology

Inspired by the fine-tuning process, which is prevalent for image classification area, we have used fine-tuning for our VSA task. Our work is based on the popular architecture of Neural Architecture Search Network (NASNet) [134]. The proposed attention mechanism is described in Section 3.1.1.1, and the overall architecture is discussed in Section 3.1.1.2.

#### 3.1.1.1 Proposed attention mechanism

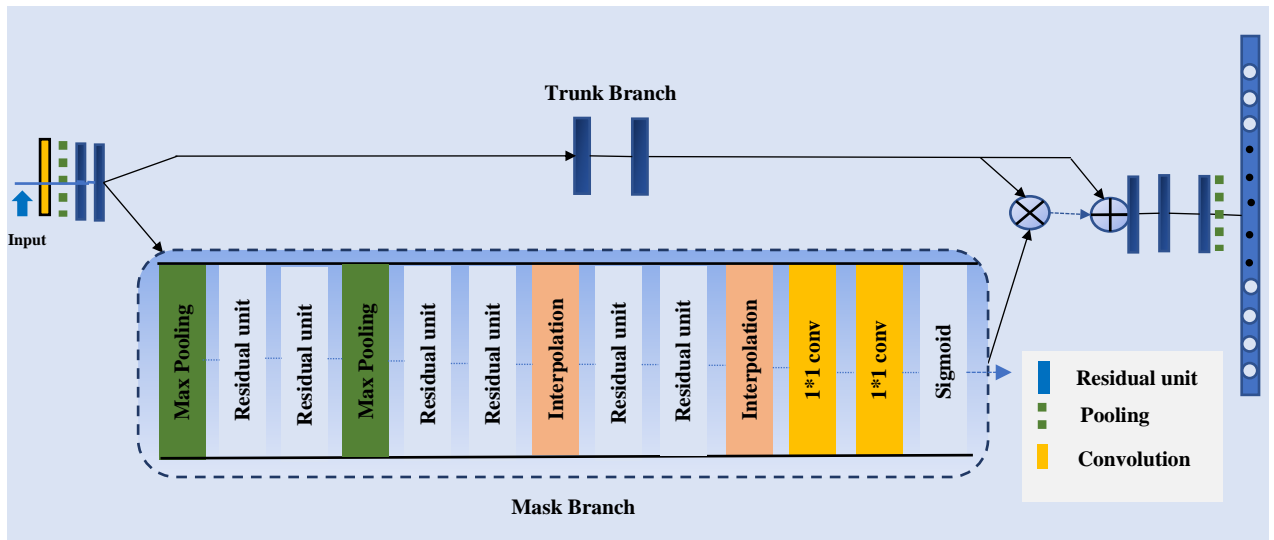


Figure 3.1 Residual Attention Network [135]

This work applies the Attention mechanism, which is inspired by the Residual Attention network [135]. In [135], the author has developed a system that was built by assembling multiple attention modules in a feedforward structure, which is different from the previous work of formulating attention as a sequential process. The proposed architecture applies a single attention module to concentrate on the most significant regions of the image.

Figure 3.1 shows a complete view of the attention module. Each module is separated into two branches. The first branch, known as the *trunk branch*, helps in the

feature engineering process. The second branch, called a *mask branch*, helps in picking the relevant features of the image to focus and learn the attention-based features. They prevent the trunk branch parameters from getting updated with the wrong gradients computed from noisy labels. The mask branch consists of a sequence of residual units in-between the pooling layers. Bi-linear interpolation is applied for upsampling the output. Finally, the sigmoid layer is added after two convolutional layers to normalize the output in  $[0, 1]$ . The output of the Attention module is calculated using Eq. (3.1):

$$H_{i,c}(\mathbf{x}) = A_{i,c}(\mathbf{x}) * T_{i,c}(\mathbf{x}) \quad (3.1)$$

Here,  $A_{i,c}(\mathbf{x})$  denotes the attention mask,  $c \in \{1, 2, \dots, C\}$  is the channel's index,  $i$  varies over the spatial positions,  $T_{i,c}(\mathbf{x})$  represents the original features generated by the trunk branch,  $*$  denotes element-wise product (dot product) between trunk branch features and mask branch features. The gradient of the mask for input features is calculated using Eq. (3.2):

$$\frac{\partial A(x,\theta) T(x,\emptyset)}{\partial \emptyset} = A(x,\theta) \frac{\partial T(x,\emptyset)}{\partial \emptyset} \quad (3.2)$$

Where  $\theta$  denotes the parameters of the mask branch and  $\emptyset$  denotes the parameters of the trunk branch. However, the repeated dot product between trunk branch features and mask branch features leads to a significant performance drop for useful and useless information. Hence, Eq. (3.1) is changed as follows:

$$H_{i,c}(x) = (1 + A_{i,c}(x)) * T_{i,c}(x) \quad (3.3)$$

$$\text{or } H_{i,c}(x) = T_{i,c}(x) + (A_{i,c}(x) * T_{i,c}(x)) \quad (3.4)$$

From Eq. (3.4), it can be seen that an element-wise product of trunk branch features and mask branch features are performed followed by element-wise addition with trunk branch features. Thus, when  $A_{i,c}(x) = 0$ , the  $H_{i,c}(x)$  will approximate to the original features  $T_{i,c}(x)$ . This process is called residual attention learning, where the soft mask unit is constructed as the same mapping. Hence, the mask branch tries to improve the trunk branch features by refining the feature maps, which makes the

network robust to noisy inputs.

### 3.1.1.2 RA-DLNet Architecture

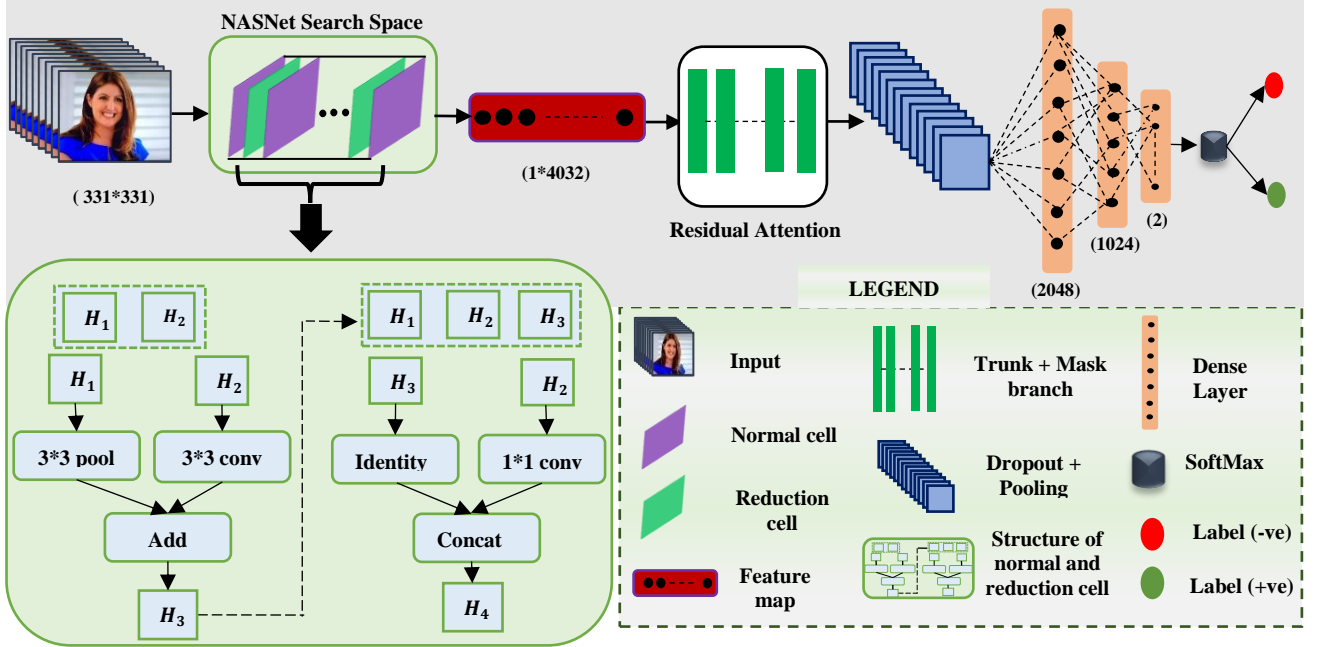


Figure 3.2 Proposed RA-DLNet architecture

The architecture of the proposed RA-DLNet framework for VSA is shown in Figure 3.2. The proposed architecture is divided into three main components: The first part consists of applying a pre-trained model. We have implemented the NASNet architecture, which is inspired by Neural Architecture Search (NAS) framework [136]. In the NAS framework, a controller RNN recursively trains a child network on a validation set for finding good architectures. The child network will generate different accuracies, which helps to calculate the gradient for updating the controller. Hence, the controller will eventually improve its searching process for learning the most appropriate architectures. The controller RNN used in our work is the LSTM model with 100 hidden units. The Proximal Policy optimization [137] is used for updating the parameters of the controller RNN. The objective function of PPO is defined as in Eq. (3.5) below:

$$P^{CPI}(\theta) = \hat{E}_t[\min(\mathfrak{r}_t(\theta) \hat{A}_t, \text{clip}(\mathfrak{r}_t(\theta), 1-\epsilon, 1+\epsilon) \hat{A}_t)] \quad (3.5)$$

The *CPI* superscript denotes the conservative policy iteration constraint, which controls

the excessive policy updates. The  $\hat{A}_t$  denotes the advantage function that estimates the relative value of the selected action in the current state. The  $\mathfrak{r}_t(\theta)$  defines the ratio between a newly updated policy output and the old policy output of the network, which is shown in Eq. (3.6) below:

$$\mathfrak{r}_t(\theta) = \frac{\Pi_{\theta}(a_t | s_t)}{\Pi_{old}(a_t | s_t)} \quad (3.6)$$

Here,  $\Pi_{\theta}(a_t | s_t)$  signifies the policy that takes observed states from the environment as input and suggests actions to take as output. Hence, in Eq. (3.5), the term  $\mathfrak{r}_t(\theta) \hat{A}_t$  pushes the policy towards actions that yield a high positive advantage over the baseline. The second term contains the truncated version of  $\mathfrak{r}_t(\theta)$  by applying a clipping operation between  $1-\epsilon$  and  $1+\epsilon$ .

The architecture of NASNet is composed of different convolutional cells, called *normal cell* and *reduction cell*. As discussed above, the controller RNN continuously searches for the best architectures of these cells. Each of the cells is made up of  $B$  (*In our case,  $B=5$* ) blocks where each block performs the operations which are described in Algorithm 1 and further relate to the different outputs of the softmax layer. Since this architecture has attained state-of-the-art on prevalent ImageNet classification task, hence we are motivated to analyze its performance for capturing the visual sentiments from the images. As shown in Figure 3.2, the input image of size  $331 \times 331$  passes through a series of *reduction cells* and *normal cells*. The former returns a feature map whose height and width are reduced by two, and the latter returns a feature map with the same dimension. Both the cell shared the same structure with different weights, as shown in Figure 3.2.

We extract the final feature maps of size  $1*4032$  and pass it to the attention mechanism module, which was discussed in Section 3.1.1.1. As discussed, the attention mechanism concentrates on the different regions of the input image to get a finer sentiment prediction. The final feature vector is passed into a dropout layer with 0.5 probability, followed by an average pooling layer. We have used three dense layers for softer dimensionality reduction. The first layer contains 2048 neurons, and the second

layer comprises 1024 neurons. The final dense layer is composed of two neurons in the case of binary sentiment classification and eight neurons in the case of emotion detection with a softmax classifier for the final classification. Algorithm 2 describes the training of RA-DLNet architecture.

---

**Algorithm 1** Working of controller RNN for generating the structure of convolutional cells in NASNet

---

**For**  $i = 1$  to 5 **do**

1. Select two hidden states  $H_1, H_2$  from the set of previously created hidden states.
2. Select an operation (identity, convolution, and pooling) to be applied on both the hidden states to generate a new output state  $H_{o1}$  and  $H_{o2}$ .
3. Combine  $H_{o1}$  and  $H_{o2}$  by using element-wise addition or concatenation to create a new hidden state  $H_3$ .
4. Append the state  $H_3$  to the existing list of hidden states for making it as potential input for the next blocks.

**end For**

**return** the concatenated value of the final cell output

---

**Algorithm 2** Training of RA-DLNet architecture for VSA

---

**Input:** Training model ( $M$ ),  $X = \{x_1, x_2, \dots, x_n\}$  are the set of training images of size 331\*331,  $Y = \{y_1, y_2, \dots, y_n\}$  are the set of sentiment labels of  $X$ .

1. Partition  $X$  and  $Y$  randomly into five groups:  $\{(X_1, Y_1), (X_2, Y_2), \dots, (X_5, Y_5)\}$ .
  2. **For**  $i = 1$  to 5 **do**
  3. Fine-tune the model  $M$  using NASNet Architecture (Algorithm 1) to obtain the feature map  $F_i$
  4. Calculate the attention over the obtained feature map  $F_i$  of  $M$
  5. Apply dropout with 0.5 probability and global average pooling layer.
  6. Add fully connected layers to  $M$  and apply the softmax classifier to calculate the sentiment probabilities.
  7. Calculate the performance of  $M$  on the set of testing images.
  8. **end for**
  9. **return** the average performance of  $M$  on the set of testing images.
- 

### 3.1.2 Experimental Results

To assess the performance of the proposed architecture, we experimented on widely used benchmark datasets for sentiment and emotion classification. We have separated the datasets into two categories: *small-scale* datasets and *large-scale* datasets according to the number of image samples. The evaluation parameters such as Precision, Recall, F1 score, and Accuracy are measured on these datasets. The upcoming sub-sections give the complete details of these datasets.

#### 3.1.2.1 Small Scale Datasets

The images on Twitter are related to the current trending topics. The users are most likely to upload the images which are related to their personal experiences. Hence, we

experimented on the four most popular Twitter datasets for binary sentiment prediction: Twitter I [112] image dataset, which itself is composed of three sub-datasets, namely 3-agree, 4-agree, 5-agree dataset, and Twitter II [108] image dataset. Amazon Mechanical Turk (AMT) annotation was used for creating the sentiment label for the images. For Twitter I dataset, five AMT annotators labeled the images of the dataset. Thus, “5-agree” indicates that all the annotators gave identical sentiment response for a given image. Twitter II dataset is composed of images from 21 different hashtags. Three different AMT annotators were asked to label the images. Finally, the tweets with unanimous labels were included in the released dataset. Table 3.1 gives the complete details about the dataset.

Table 3.1 Twitter Dataset

| Classes               | 5- Agree | At least 4- Agree | At least 3- Agree | Twitter II |
|-----------------------|----------|-------------------|-------------------|------------|
| # of Positive samples | 581      | 689               | 769               | 470        |
| # of Negative samples | 301      | 427               | 500               | 133        |
| Total                 | 882      | 1116              | 1269              | 603        |



Figure 3.3 Sample images from (a) Twitter dataset (b) Artphoto dataset

The major challenge with these datasets is that the images are noisy. Some of the images uploaded by the users are not captured clearly. Moreover, some of the images are quite challenging for the classifier as they contain multiple images within a single image or contains either text only or a combination of text + image both. Figure 3.3 (a) shows some sample images from the Twitter dataset.

For multi-class emotion detection, we have work with ArtPhoto [138] dataset

which consists of 806 artistic photos that are collected from some of the art sharing sites and categorized into eight discrete emotions namely, *Amusement*, *Anger*, *Awe*, *Contentment*, *Disgust*, *Excitement*, *Fear*, *Sadness*. The complete details of the dataset are given in Table 3.2.

**Table 3.2 ArtPhoto Dataset Details**

| Emotion      | Amusement | Anger | Awe | Contentment | Disgust | Excitement | Fear | Sadness | Total |
|--------------|-----------|-------|-----|-------------|---------|------------|------|---------|-------|
| # of Samples | 101       | 77    | 102 | 70          | 70      | 105        | 115  | 166     | 806   |

Some of the sample images from this dataset is shown in Figure 3.3 (b). The photos evoke a certain emotion by combining and manipulating specific visual characteristics like colors, image position, etc.

### 3.1.2.2 Large scale Datasets

We assess the performance of our model on several large-scale datasets, which include CMU-MOSI [139], CMU-MOSEI [140], Flickr [141], Instagram [141] for binary sentiment prediction, and F&I [142] for multi-label emotion classification. CMU-MOSI consists of 2199 opinionated video clips, and CMU-MOSEI contains 17,859 clips. Both the datasets were annotated with sentiment classes in the range of  $[-3,3]$ . The distribution of samples for binary sentiment classification is same as in [143], in which for CMU-MOSI, frames  $[-3,0)$  were taken as negative and frames  $[0,3]$  as positive and for CMU-MOSEI, frames with  $[-3,0)$  were taken as negative and frames  $(0,3]$  as positive. Since both these datasets have primarily been used for multimodal sentiment classification and the purpose of our work is to focus on visual sentiments, hence we have only focused on extracting the sentiments from videos (ignoring the audio and textual modality). The raw features were obtained from CMU Multimodal SDK [144]. The Flickr and Instagram sentiment datasets contain 60,745 (48,139 positive and 12606 negative) and 42,856 (33,076 positive and 9,780 negative) images respectively, as some of the images were not available because they were removed by the users. The Flickr & Instagram (F&I) emotion dataset contains 23,308 images annotated with eight discrete emotions, namely, Amusement (4942), Anger (1266), Awe (3151), Contentment (5374), Disgust (1658), Excitement (2963), Fear (1032), Sadness (2922). These are by far the largest datasets in this domain.

### 3.1.2.3 Implementation Details

This section describes the evaluation settings, classification results, baseline comparison, layer ablation study, visualizations, and an empirical study of NASNet with other CNN architectures.

#### A. Evaluation Settings

All the experiments were performed on a 64-bit Windows 10 machine with 128 GB RAM on NVIDIA Titan RTX GPUs. Keras framework was used to build the network using Python 3. Data augmentation techniques like zooming, shearing, horizontal flipping, rescaling were applied to avoid overfitting. We have used Adam as an optimizer with an initial learning rate of 0.001. Xavier uniform initializer is used to set the weights of the new layers, and the weights of the pre-trained model were used to train the remaining layers.

#### B. Classification Results

We have employed Precision (P), Recall (R), F1 score (F1), and Accuracy (ACC) metrics to measure the performance of the RA-DLNet on all the datasets. Five-fold cross-validation is used to represent each class equally across the individual test fold and assess the model's performance. The final results were computed by averaging the values of each fold. The classification results are shown in Table 3.3.

- For the Twitter I and Twitter II dataset, we have taken 80% samples for training and 20% for testing with a batch size of 32. The model was trained with binary cross-entropy loss for 50 epochs.
- For Artphoto dataset, since we have an unbalanced number of images per emotion category, we followed *one vs. all* setup strategy, where we separate each category against the rest of the others in turn. The model was trained with categorical cross-entropy loss for 50 epochs.
- For the CMU-MOSI dataset, the training, validation, and testing set consists of 1284 clips, 229 clips, 686 clips respectively. Similarly, the training, validation, and testing set for CMU-MOSEI dataset include 12787, 3634, and 1438 clips



respectively. The model was trained with binary cross-entropy for 100 epochs with a batch size of 128.

- For Flickr and Instagram datasets, we have taken 90% as training samples and 10% as testing samples, as specified in [141]. The model was trained with binary cross-entropy loss for 100 epochs with a batch size of 128.
- For F&I datasets, we have used 80% as training samples, 5% as validation, and 15% as testing samples, as specified in [142]. The model was trained with categorical cross-entropy loss for 100 epochs with a batch size of 128.

**Table 3.3 Classification results (%) of RA-DLNet on benchmark datasets**

| Dataset    | P           | R           | F1          | ACC         |
|------------|-------------|-------------|-------------|-------------|
| 3-agree    | 88.2        | 78.9        | 83.2        | 81.3        |
| 4-agree    | 86.7        | 85.1        | 85.8        | 83.2        |
| 5-agree    | <b>92.5</b> | 95.7        | <b>93.8</b> | <b>89.1</b> |
| Twitter II | 80.7        | <b>99.5</b> | 89.1        | 81.2        |
| ART Photo  | 76.3        | 86.5        | 80.9        | 71.8        |
| CMU-MOSI   | 64.7        | 73.3        | 68.7        | 67.7        |
| CMU-MOSEI  | 82.1        | 70.5        | 75.9        | 77.2        |
| Flickr     | 88.6        | 90.6        | 89.6        | 83.4        |
| Instagram  | 84.7        | 96.6        | 90.2        | 82.7        |
| F&I        | 74.7        | 72.8        | 73.7        | 72.8        |

From Table 3.3, we observe that in the Twitter domain, 5-agree dataset gives the best accuracy rate of 89.1% and an F1 score of 93.8%, whereas the Twitter II dataset gives an accuracy of 81.2% with an F1 score of 89.1%. The intuition behind this behavior is that the Twitter II dataset contains noisier images (blur images) and challenging images in the form of multiple images within a single image or images with either text or a combination of both text + image, as discussed in Section 3.1.2.1. However, when we compare our results with the earlier state-of-the-art methods in the upcoming sections, we observe that our proposed architecture performs better than state-of-the-art methods. In the Artphoto dataset, the proposed architecture obtained an accuracy of 71.8% for multiclass emotion classification.

For large-scale datasets, CMU-MOSI and CMU-MOSEI have shown an accuracy of 67.7% and 77.2%, respectively, whereas in Flickr and Instagram sentiment datasets the accuracy obtained is 83.4% and 82.7%, respectively. Hence, the proposed model has effectively characterized the facial expressions in the images. Although these

datasets were cleaner as compared to Twitter I and Twitter II datasets, yet the model misclassified some cases where text or letters were embedded in the images. For the F&I emotion dataset, the model has achieved an accuracy of 72.8%. Due to more number of samples corresponding to each emotion, the network could learn more and hence improves the classification performance as compared to the Artphoto dataset. Deeper insights into the performance of each emotion class are shown in the next section.

### C. Baseline Comparison

This section compares the proposed RA-DLNet architecture with state-of-the-art works on all eight datasets.

#### 1) Twitter Dataset:

The results of RA-DLNet on Twitter I and Twitter II datasets are compared with state-of-the-art in Table 3.4 and Table 3.5, respectively. *Bold* values represent our results. We have used the following baselines for comparison.

- **Low-Level feature-based approaches:** Siersdorfer *et al.* [103] used Global Color Histogram (GCH), Local Color Histogram (LCH), and SIFT-based bag of visual terms. The GCH based approach selects 4\*4\*4 RGB histogram for representing the features, whereas the LCH based approach first distributes each image into 16 different blocks to compute the 64-bit RGB histogram for each block. Finally, they applied SIFT descriptors to learn the visual word dictionary and used a bag of visual word features (BoW) for every image. A combination of GCH+BoW and LCH+BoW was also employed for combining the features of an image.
- **Mid-level feature-based approaches:** SentiBank [108] is a visual sentiment ontology comprising 1,200 ANPs to generate 1,200 responses for any test image. Sentibute [109] applies semantic features using scene-based attributes instead of using ANPs. Finally, the two mid-level features were classified by SVM. [145] fused the sentiment specific value with ANP values to compute the

final classification score.

- **Deep Learning-based approaches:** The deep learning approaches based on transfer learning for VSA are used for comparison with the proposed architecture. They are as follows:
  - (i) You *et al.* [112] designed CNN based architecture consisting of two convolutional layers with several dense layers for VSA. Progressive CNN (PCNN) was used to fine-tune the network.
  - (ii) Campos *et al.* [146], [147] fine-tuned CaffeNet, with five convolutional layers and three dense layers for VSA. They also used oversampling to remove the dataset bias.
  - (iii) Wang *et al.* [148] proposed DCAN (Deep Coupled Adjective and Noun) network to extract the descriptiveness and objectiveness features. They applied transfer learning with pre-trained Alexnet and Rectified Kullback-Leibler loss (ReKL) to build a scalable network.
  - (iv) Song *et al.* [149] applied VGGNet based saliency detection mechanism for VSA.
  - (v) Islam *et al.* [150] used transfer learning by applying the weights and biases from GoogleNet, which was pre-trained for VSA.
  - (vi) Fan *et al.* [119] applied the VGG-19 network and devoted a full channel for saliency maps or focal object masks.

**Table 3.4 Performance comparison of proposed architecture on Twitter I (%)**

| Ref.  | Method                             | 3-agree |      |      |      | 4-agree |      |      |      | 5-agree |      |      |      |
|-------|------------------------------------|---------|------|------|------|---------|------|------|------|---------|------|------|------|
|       |                                    | P       | R    | F1   | ACC  | P       | R    | F1   | ACC  | P       | R    | F1   | ACC  |
| [103] | GCH                                | 67.8    | 83.6 | 74.9 | 66.0 | 68.7    | 84.0 | 75.6 | 66.5 | 70.8    | 88.8 | 78.7 | 68.4 |
|       | LCH                                | 71.6    | 73.7 | 72.6 | 66.4 | 72.5    | 75.3 | 73.9 | 67.1 | 76.4    | 80.9 | 78.6 | 71.0 |
|       | GCH + BoW                          | 68.3    | 83.5 | 75.1 | 66.5 | 70.3    | 84.9 | 76.9 | 68.5 | 72.4    | 90.4 | 80.4 | 71.0 |
|       | LCH + BoW                          | 72.2    | 72.6 | 72.3 | 66.4 | 75.1    | 76.2 | 75.6 | 69.7 | 77.1    | 81.1 | 79.0 | 71.7 |
| [108] | SentiBank                          | 72.0    | 72.3 | 72.1 | 66.2 | 74.2    | 72.7 | 73.4 | 67.5 | 78.5    | 76.8 | 77.6 | 70.9 |
| [109] | Sentribute                         | 73.3    | 78.3 | 75.7 | 69.6 | 75.0    | 79.2 | 77.1 | 70.9 | 78.9    | 82.3 | 80.5 | 73.8 |
| [112] | CNN                                | 73.4    | 83.2 | 77.9 | 71.5 | 77.3    | 85.5 | 81.1 | 75.5 | 79.5    | 90.5 | 84.6 | 78.3 |
| [112] | PCNN                               | 75.5    | 80.5 | 77.8 | 72.3 | 78.6    | 84.2 | 81.1 | 75.9 | 79.7    | 88.1 | 83.6 | 77.3 |
| [147] | Fine-tuned CaffeNet + oversampling | -       | -    | -    | -    | -       | -    | -    | -    | -       | -    | -    | 83.0 |
| [148] | DCAN +ReKL                         | -       | -    | -    | -    | -       | -    | -    | -    | -       | -    | -    | 83.8 |
| [146] | Fine-tuned CaffeNet + oversampling | -       | -    | -    | 74.9 | -       | -    | -    | 78.7 | -       | -    | -    | 83.0 |
| [145] | Hybrid SentiBank + Late fusion     | 73.9    | 80.9 | 77.2 | 71.1 | 76.5    | 82.3 | 79.2 | 73.4 | 80.4    | 86.4 | 83.3 | 77.2 |

|                 |                                 |             |             |             |             |             |             |             |             |             |             |             |             |
|-----------------|---------------------------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|
| [119]           | VGG19 + saliency map            | -           | -           | -           | -           | -           | -           | -           | -           | 91.0        | 89.0        | 90.0        | 87.0        |
| [149]           | SentiNet-A                      | 82.0        | 80.0        | 81.0        | 77.7        | 85.0        | 83.0        | 84.0        | 80.7        | 89.0        | 87.0        | 88.0        | 85.1        |
| [150]           | GoogLeNet                       | 85.1        | 78.4        | 81.6        | 78.7        | 84.9        | 83.0        | 83.9        | 80.7        | 92.6        | 86.9        | 89.7        | 86.1        |
| [151]           | CNN                             | -           | -           | -           | 67.77       | -           | -           | -           | -           | -           | -           | -           | -           |
| [113]           | VGGNet + Fusion                 | -           | -           | -           | 81.0        | -           | -           | -           | 85.1        | -           | -           | -           | 88.6        |
| <b>RA-DLNet</b> | <b>CNN + Residual Attention</b> | <b>88.2</b> | <b>78.9</b> | <b>83.2</b> | <b>81.3</b> | <b>86.7</b> | <b>85.1</b> | <b>85.8</b> | <b>83.2</b> | <b>92.5</b> | <b>95.7</b> | <b>93.8</b> | <b>89.1</b> |

The experimental results show that the proposed architecture has outperformed the baseline results on all the evaluation metrics. The results reveal that deep learning-based approaches have outperformed the best performing low-level and mid-level approaches like SentiBank [109]. The layers of NASNet extract the local regions of the input image, whereas the attention mechanism helps to filter out the irrelevant regions and capture the global informative areas in the images. Moreover, attention can also deal with noise in the dataset. Hence, the residual attention has further complimented the deep learning CNN based architecture (NASNet) to enhance the VSA process. This shows the importance of attention mechanisms for extracting visual sentiments from the images.

**Table 3.5 Performance comparison of proposed architecture on Twitter II dataset (%)**

| Ref.            | Method                          | P           | R           | F1          | ACC         |
|-----------------|---------------------------------|-------------|-------------|-------------|-------------|
| [108]           | SentiBank                       | -           | -           | -           | 70.0        |
| [145]           | Hybrid SentiBank + Late fusion  | 78.1        | 100         | 87.6        | 78.1        |
| [113]           | VGGNet + Fusion                 | -           | -           | -           | 80.4        |
| [117]           | WCSNet                          | -           | -           | -           | 81.3        |
| <b>RA-DLNet</b> | <b>CNN + Residual Attention</b> | <b>80.7</b> | <b>99.5</b> | <b>89.1</b> | <b>81.2</b> |

## 2) Artphoto Dataset

Since this dataset suffers from unequal data distribution problems per emotion category, hence we calculate the true positive rate per class averaged over positive and negative samples, as discussed in [138]. The following baselines are used for comparison:

- Machajdik *et al.* [138] extracted low-level features, which include color, texture, composition, content from the images.
- Zhao *et al.* [110] extracted emotion features that were derived from different aspects like harmony, balance, proportion, variety, etc.
- Rao *et al.* [152] [153] used probabilistic latent semantic with the bag of visual words for generating a mid-level representation from the images.

- Wang *et al.* [154] extracted aesthetic features like color difference, shape, composition, etc. for affective image analysis.
- Liu *et al.* [155] used pre-trained deep CNN for generating semantic orientation from the images.

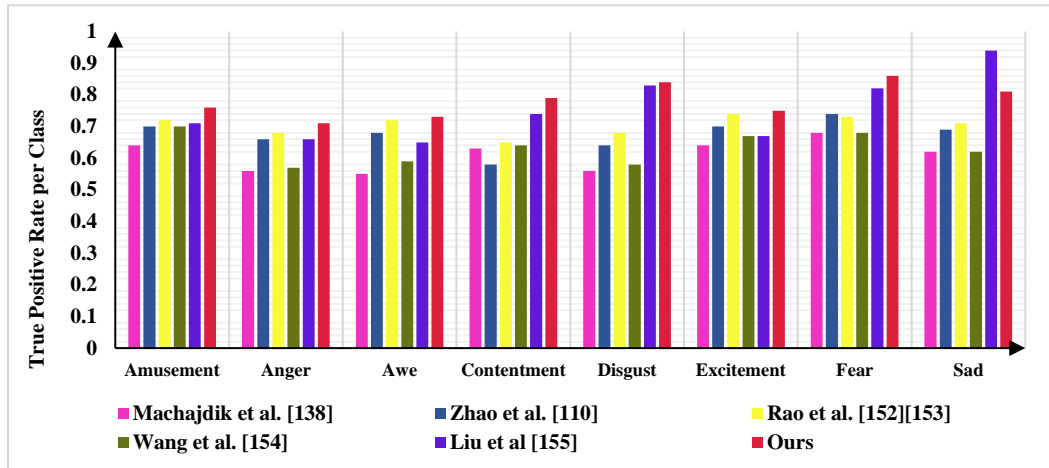


Figure 3.5 Performance comparison of proposed architecture on Artphoto dataset with similar state-of-the-art

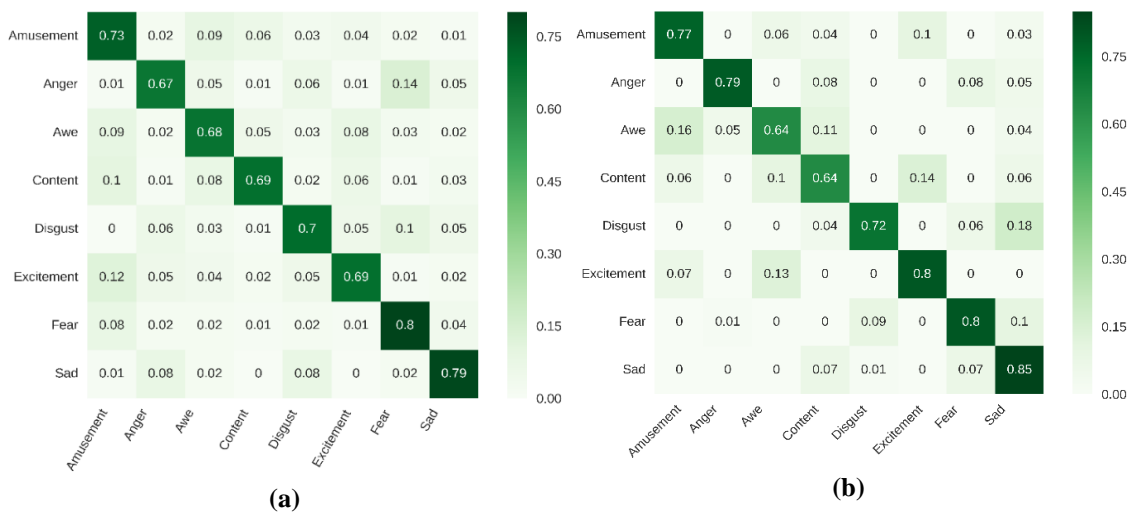


Figure 3.4 Confusion Matrix for (a) Artphoto dataset (b) F&I dataset

From Figure 3.4, we see that our approach has outperformed the state-of-the-art results majorly on each of the emotion types. This improvement may be credited to the fact that the residual attention mechanism can capture the different attention-aware features from the other regions of the image. However, for *sad* emotion, the work done by Liu *et al.* [155] architecture has shown better results. Figure 3.5 (a) shows the confusion matrix for eight classes on Artphoto dataset. As seen in the figure, some of the emotions are difficult to classify, such as *anger* and *sad*. This behavior is intuitive because the

images in both categories are dull with dark colors.

### 3) CMU-MOSI & MOSIE Dataset

As discussed, we have focused on extracting the visual sentiments from the videos. Hence we have compared our results with similar state-of-the-art works on video modality only. Table 3.6 and Table 3.7 show the comparative results of the proposed RA-DLNet model on CMU-MOSI & MOSEI datasets, respectively. Our results are highlighted in Bold.

**Table 3.6 Performance comparison of proposed architecture on CMU-MOSI dataset (%)**

| Ref.            | Method                          | ACC         | F1          |
|-----------------|---------------------------------|-------------|-------------|
| [156]           | TFN                             | 69.4        | 71.4        |
| [157]           | GME-LSTM-A                      | 52.3        | 57.3        |
| [158]           | Bi-LSTM                         | 53.8        | 53.5        |
| [159]           | DCC-CAE                         | 59.4        | 59.3        |
| [160]           | CIA based RNN                   | 51.31       | 50.48       |
| [143]           | DCCA                            | 44.0        | 44.5        |
| [161]           | ICCN                            | 48.10       | 49.06       |
| [162]           | HvnLBP-TOP                      | 63.9        | 65.9        |
| <b>RA-DLNet</b> | <b>CNN + Residual Attention</b> | <b>67.7</b> | <b>68.7</b> |

**Table 3.7 Performance comparison of proposed architecture on CMU-MOSEI dataset (%)**

| Ref.            | Method                          | ACC         | F1          |
|-----------------|---------------------------------|-------------|-------------|
| [158]           | Bi-LSTM                         | 59.6        | 59.5        |
| [159]           | DCC-CAE                         | 67.3        | 67.5        |
| [160]           | CIA based RNN                   | 75.04       | 69.13       |
| [163]           | GRU based attention             | 76.5        | 69.1        |
| [143]           | DCCA                            | 58.75       | 59.23       |
| [161]           | ICCN                            | 59.25       | 59.90       |
| <b>RA-DLNet</b> | <b>CNN + Residual Attention</b> | <b>77.2</b> | <b>75.9</b> |

The following baselines are used for comparison:

- Dumpala *et al.* [158] extracted visual features using FACET framework and OpenFace toolkit. They applied the Bi-directional LSTM-RNN classifier for capturing the contextual features.
- Dumpala *et al.* [159] proposed a deep canonically correlated cross-modal autoencoder (DCC-CAE) model that combines deep canonical correlation analysis along with cross-modal encoder.
- Chauhan *et al.* [160] proposed Context aware Interactive Attention (CIA) based RNN framework to capture the interaction between different modalities using a

contextual attention model.

- Akhtar *et al.* [163] developed a GRU based framework based on context-based inter-modal attention framework.
- Sun *et al.* [143] developed Deep Canonical Correlation Analysis (DCCA) to learn the embeddings of data for combining audio, visual, and text modality and improve the SA process.
- Sun *et al.* [161] proposed Interaction Canonical Correlation Network (ICCN), which learns the hidden relationships between all three modalities.
- Zadeh *et al.* [156] developed Tensor Fusion Network (TFN), which models the interactions between different modalities by computing a 3-fold cartesian product from the embeddings of the modality.
- Chen *et al.* [157] proposed Gated Multimodal embedding LSTM with temporal attention (GME-LSTM-A), which tackles the noisy modalities and attention focus on the most crucial time steps.
- Li *et al.* [162] proposed the HvnLBP-TOP approach for combining handcrafted features on three orthogonal planes followed by bi-LSTM for final sentiment classification.

From the above table, it can be seen that our proposed algorithms achieved significantly good results as compared with several baselines on both datasets. We have reported only the results of video modality for the baseline comparison. The accuracy reported on CMU-MOSI and CMU-MOSIE datasets is 67.7% and 77.2%, respectively, which shows that the model has outperformed the baseline results.

#### 4) Flickr and Instagram Sentiment Datasets

Table 3.8 compares the accuracy of our work with earlier baseline results on Flickr and Instagram sentiment datasets. The following methods are used for baseline comparison.

- Borth *et al.* [108] provided ANPs, which were used as mid-level features for every image and fed into the SVM and logistic regression classifiers.
- Chen *et al.* [164] introduced Sentibank 2.0 called DeepSentiBank, which used CNN-based deep networks for visual sentiment classification.

- Katsurai *et al.* [141] applied latent correlations among multiple views (visual, textual, and sentiment) using SentiWordNet. They obtained a latent embedding space to maximize the correlations among multiple views.
- He *et al.* [165] proposed Multi-Attentive Pyramidal model (MAP), which gives the association between local features of the image and the sentiment information represented by them.
- Yang *et al.* [166] and She *et al.* [117] developed weakly supervised coupled networks for detecting and classifying the sentiments using the detection branch and classification branch, respectively. The final sentiment maps highlight the important part of the image, which serves as crucial information for the sentiment classification process. However, the network fails to capture the sentiments in complex scenes, and no interest regions are also detected in many images.

In Table 3.8, we see that the deep learning approaches are increasingly being applied for visual sentiment classification as they have outperformed the low-level and mid-level strategies. However, the proposed RA-DLNet has outperformed the state-of-the-art deep learning methods for Flickr and Instagram sentiment datasets by learning more discriminative features for this task.

**Table 3.8 Performance comparison (ACC) of proposed architecture on Flickr and Instagram Sentiment Datasets (%)**

| Ref.            | Method                             | Flickr      | Instagram   |
|-----------------|------------------------------------|-------------|-------------|
| [108]           | SentiBank                          | 69.2        | 66.5        |
| [164]           | DeepSentiBank                      | 70.1        | 67.1        |
| [141]           | Latent correlations                | 74.7        | 73.6        |
| [165]           | Multi- Attentive Pyramidal model   | 80.9        | 81.9        |
| [166]           | Weakly Supervised Coupled Networks | 81.3        | 81.8        |
| [117]           | WSCNet                             | 81.3        | 81.8        |
| <b>RA-DLNet</b> | <b>CNN + Residual Attention</b>    | <b>83.4</b> | <b>82.7</b> |

### 5) F&I emotion dataset

Table 3.9 compares the accuracy of our work with earlier baseline results. We have included the following methods for baseline comparison:

- Yang *et al.* [167] developed multi-task CNN based on VGGNet to establish the relationship between multiple emotion categories for visual emotion prediction.



- Zhu *et al.* [168] proposed a unified CNN-RNN based framework to exploit the dependency between the features.
- Yang *et al.* [169] explored the relationship between the emotion labels by applying CNN to optimize the classification loss and sentiment loss jointly.
- Zhao *et al.* [170] proposed an unsupervised approach that uses an adversarial model called CycleEmotionGan to have a similar distribution of images in the source and target domain.
- Zhang *et al.* [171] developed a novel method to integrate low-level visual features in the image like color, shape with high-level information extracted by layers of CNN.

**Table 3.9 Performance comparison of proposed architecture on F&I Emotion Dataset (%)**

| Ref.            | Method                          | ACC         |
|-----------------|---------------------------------|-------------|
| [108]           | Sentibank                       | 49.2        |
| [164]           | DeepSentibank                   | 51.5        |
| [167]           | VGG_Net                         | 67.4        |
| [168]           | CNN-RNN                         | 73.0        |
| [169]           | Deep metric learning            | 67.6        |
| [170]           | CycleEmotionGAN                 | 66.8        |
| [117]           | WSCNet                          | 70.0        |
| [171]           | CNN                             | 71.7        |
| <b>RA-DLNet</b> | <b>CNN + Residual Attention</b> | <b>72.8</b> |

The results in Table 3.9 show that the proposed method has outperformed all the baseline results. This shows that applying attention mechanism for capturing the local information has significantly improved the performance for image emotion classification. To get more insights into the classification results for each emotion, Figure 3.5 (b) shows the confusion matrix for the F&I dataset. Our approach shows a balanced performance, especially for negative emotions like anger, sad, and fear which is the most common problem for visual classifiers as these emotions are hard to be discriminated against by the classifiers.

#### D. Further Evaluation Results

The proposed architecture is further evaluated by training on 80% of the samples, validating on 10% samples, and finally testing on the remaining 10% samples. The final results with this split (80:10:10) are given in Table 3.10.

Table 3.10 ACC (%) on all the datasets with (80:10:10) split

| Dataset    | # of Training Samples | # of Validation Samples | # of Testing Samples | Accuracy     |
|------------|-----------------------|-------------------------|----------------------|--------------|
| 3-agree    | 1015                  | 127                     | 127                  | <b>82.8</b>  |
| 4-agree    | 892                   | 112                     | 112                  | <b>84.14</b> |
| 5-agree    | 705                   | 88                      | 89                   | <b>91.01</b> |
| Twitter II | 493                   | 52                      | 58                   | <b>81.6</b>  |
| ArtPhoto   | 645                   | 80                      | 81                   | <b>62.4</b>  |
| CMU-MOSI   | 1759                  | 219                     | 221                  | <b>73.1</b>  |
| CMU-MOSEI  | 14287                 | 1785                    | 1787                 | <b>79.7</b>  |
| Flickr     | 48596                 | 6074                    | 6075                 | <b>87.35</b> |
| Instagram  | 34285                 | 4286                    | 4285                 | <b>85.10</b> |
| F&I        | 18646                 | 2330                    | 2330                 | <b>80.98</b> |

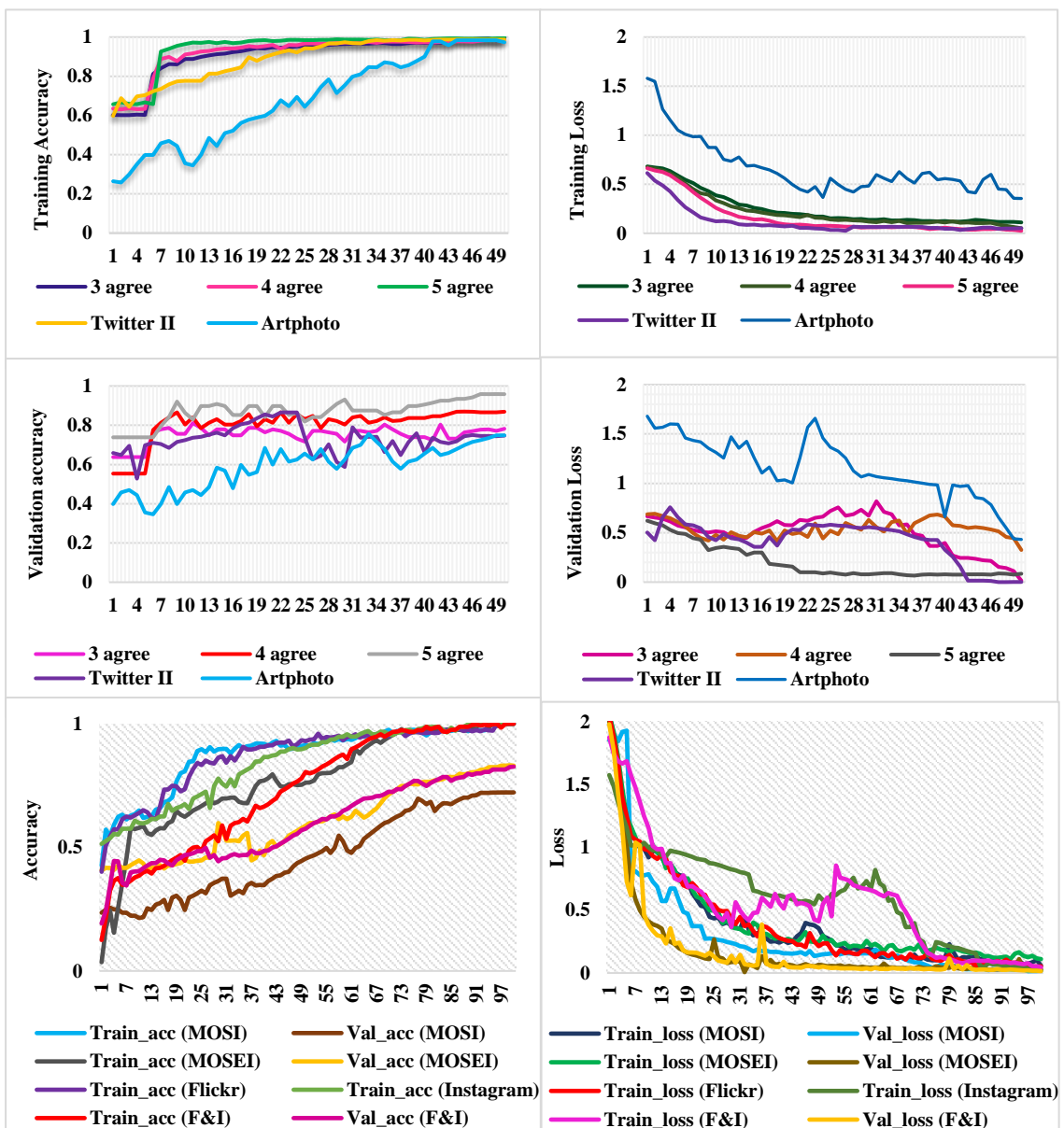


Figure 3.6 Training and validation curves for each dataset

From Table 3.10, we see that with this split (80:10:10), the proposed RA-DLNet is performing even better and have crossed the results obtained by the five-fold cross-validation accuracy. However, for Artphoto dataset the accuracy is dropped from 71.8% to 62.4%. This may be articulated by the fact that for multi-label emotion classification, the model didn't get enough training images per emotion category for learning. In CMU-MOSI and CMU-MOSEI dataset, the accuracy increases by 6% and 2%, respectively, and for Flickr and Instagram sentiment datasets, the accuracy improves by 4% and 3%, respectively. In F&I dataset, the accuracy considerably increases from 72.8% to 80.98%. The training accuracy, validation accuracy, training loss, and validation loss curves for all the datasets can be visualized by the graph plots shown in Figure 3.6. As discussed, the network is trained for 50 epochs on small scale datasets and for 100 epochs on large scale datasets using 80:10:10 split as described above. From the accuracy plots, we infer that the training accuracy has reached almost 100% for every dataset. Moreover, as the model gets trained, the validation accuracy also increases with every epoch. The loss curves show a decreasing trend. This denotes that the model has a comparable performance on the training and validation sets and does not overfit on the data.

### **E. Layer Ablation study and Visualizations**

To show the effectiveness of our proposed method, we perform an ablation study and provide some visualizations by showing the *activation maximization plots* and *colormaps* for some sample images from the datasets.

- **Layer Ablation Study**

In this section, we perform layer ablation study to quantify the effect of some vital layers of the RA-DLNet architecture for VSA. We re-train our model by ablating the following components on the Flickr sentiment dataset. The results are reported in Table 3.11.

We conclude several important observations from the study. When we removed the trunk branch and soft mask branch of our attention module and extracted the spatial features from the images, we achieved only 45.1% accuracy. This verifies the

effectiveness of our attention module for extracting the visual sentiments and proves that the attention module is necessary for more accurate sentiment prediction. However, if only the soft mask branch is ablated, we observe that the model reaches the lowest accuracy. The reason behind this behavior is that when the network is trained with noisy data, the wrong gradients computed from noisy labels might update the trunk branch parameters, which further deteriorates the learning of our model. Hence, the soft mask branch was added to prevent the trunk branch from getting updated with gradients caused by label error.

**Table 3.11 ACC results (%) for layer ablation study on Flickr dataset**

| NasNet CNN | Trunk branch | Soft mask branch | Dropout | Pooling | Dense_1 (2048) | Dense_2 (1024) | Dense_3 (2) | SoftMax     |
|------------|--------------|------------------|---------|---------|----------------|----------------|-------------|-------------|
| ✓          |              |                  | ✓       | ✓       | ✓              | ✓              | ✓           | 45.1        |
| ✓          | ✓            |                  | ✓       | ✓       | ✓              | ✓              | ✓           | 42.2        |
| ✓          | ✓            | ✓                |         | ✓       | ✓              | ✓              | ✓           | 64.1        |
| ✓          | ✓            | ✓                | ✓       | ✓       |                | ✓              | ✓           | 71.3        |
| ✓          | ✓            | ✓                | ✓       | ✓       | ✓              |                | ✓           | 76.6        |
| ✓          | ✓            | ✓                | ✓       | ✓       | ✓              | ✓              | ✓           | <b>83.4</b> |

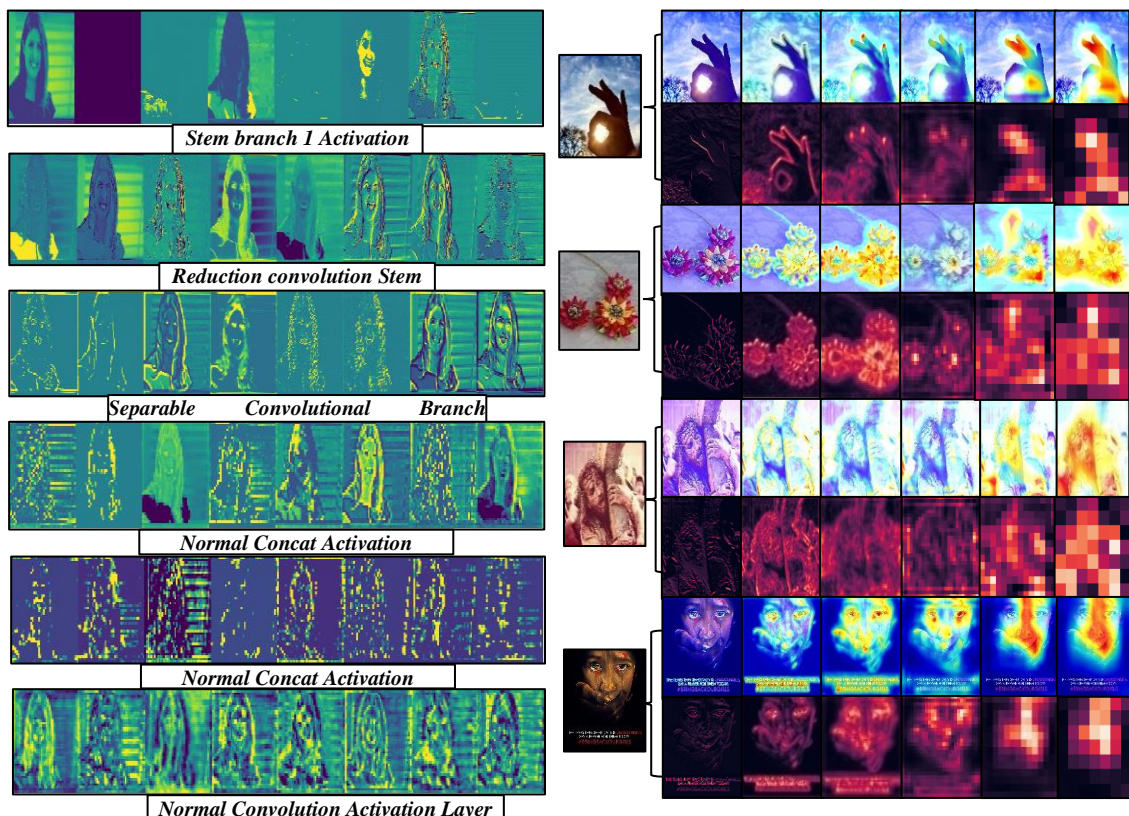
When *Dense\_1* (2048) and *Dense\_2* (1024) layers are ablated, accuracy decreases to 71.3% and 76.6%, respectively. Since the decrease in the number of parameters from the pooling layer to *Dense\_2* (1024) is sudden, the model's efficiency is profoundly affected. However, the reduction in the number of parameters from *Dense\_1* (2048) to *Dense\_3* (2) is quite smooth, which has slightly less impact on the accuracy. The intuition behind this is that softer dimensionality reduction proves beneficial for the architecture which was provided by intermediate fully connected layers. The evolving accuracy at each layer shows how the features are learned effectively along with the network.

### ▪ Visualizations

In order to visualize what information is captured by different layers of the proposed model, we show the *activation maximization plots* in Figure 3.7 (a). From the figure, we observe that as we go deeper and deeper in the layers, the input image becomes less visually interpretable. This is because initially, all the filters of the layer are not activated, hence most of the information of the image is retained. But as we go deeper into the layers, more and more filters get activated to get a higher representation of the image. These representations carry less information about the visuals of the image and

more details on the class of the image. Finally, a point comes when the layers are not activating any filters, and the network has nothing more to learn. At this point, the training of the model gets stopped. Hence, these plots explain how the particular filters are activated in each layer from the input pattern.

To provide more visualizations on what the CNN learns from the visual patterns of an image, we show the color maps and the corresponding heatmaps of some images in Figure 3.7 (b). These maps highlight the crucial regions of the image, which helps CNN to make the final prediction. This shows what features CNN relies on by focusing on that region. As seen in the figure, the first input image corresponds to a human hand. The initial layers are not exactly concentrating on the hand, but as we reach towards the final layer, the network starts focusing on the hand region of the input image. Accordingly, in the corresponding heatmap of the image, the pixels of the hand regions are getting activated towards the final layer. This shows that the model is learning to localize the crucial visual features of the image.



**Figure 3.7** Activation Maximization Plots to visualize each channel generated by different layers of the RA-DLNet architecture (b) Colormaps and corresponding heatmaps of some images to visualize the crucial regions of example images

## F. Comparison of NASNet with other CNN architectures

Table 3.12 Summary of CNN architectures

| Model              | Input size | Top-1 accuracy (%) | Top-5 accuracy (%) | Parameters (millions) | Depth | Size (MB) |
|--------------------|------------|--------------------|--------------------|-----------------------|-------|-----------|
| VGG-16             | 224*224    | 71.3               | 90.1               | 138                   | 23    | 528       |
| VGG-19             | 224*224    | 71.3               | 90.0               | 143                   | 26    | 549       |
| ResNet-50          | 224*224    | 74.9               | 92.1               | 25                    | -     | 98        |
| Inception-V3       | 299*299    | 78.8               | 94.4               | 23                    | 159   | 92        |
| InceptionResnet-V2 | 299*299    | 80.1               | 95.1               | 55                    | 572   | 215       |
| Xception           | 299*299    | 79.0               | 94.5               | 22                    | 126   | 88        |
| NASNet             | 331*331    | 82.7               | 96.2               | 88                    | -     | 343       |

To further examine the competence of the proposed architecture, we have compared the performance of the NASNet with six popular CNN architectures: VGG-16 [172], VGG-19 [172], Inception-Resnet-V2 [173], Inception-V3 [174], ResNet-50 [175], and Xception [176] by replacing the NASNet layer in our proposed RA-DLNet architecture with the above six architectures and training the model end-to-end for each architecture. The experimental settings for every architecture are the same as described in Section 3.1.2.3 (A).

Table 3.13 Five-cross validation results (mean  $\pm$  std) on other popular architectures

| Dataset    | VGG-16          | VGG-19          | ResNet – 50     | Inception-V3    | Inception-Resnet V2 | Xception        | Nasnet          |
|------------|-----------------|-----------------|-----------------|-----------------|---------------------|-----------------|-----------------|
| 3 agree    | 67.6 $\pm$ 0.67 | 67.2 $\pm$ 0.42 | 52.3 $\pm$ 2.12 | 75.7 $\pm$ 3.07 | 76.8 $\pm$ 3.86     | 77.4 $\pm$ 0.54 | 81.3 $\pm$ 0.70 |
| 4 agree    | 72.4 $\pm$ 0.82 | 70.0 $\pm$ 2.92 | 38.2 $\pm$ 2.78 | 76.6 $\pm$ 1.45 | 78.1 $\pm$ 0.72     | 78.8 $\pm$ 3.84 | 83.2 $\pm$ 0.84 |
| 5 agree    | 77.4 $\pm$ 0.02 | 75.2 $\pm$ 0.75 | 34.4 $\pm$ 1.85 | 79.6 $\pm$ 1.29 | 84.5 $\pm$ 2.22     | 84.8 $\pm$ 3.13 | 89.1 $\pm$ 0.01 |
| Twitter II | 76.1 $\pm$ 3.16 | 75.5 $\pm$ 2.59 | 77.9 $\pm$ 2.07 | 75.4 $\pm$ 0.24 | 77.9 $\pm$ 0.94     | 74.3 $\pm$ 2.75 | 81.2 $\pm$ 1.18 |
| ArtPhoto   | 43.1 $\pm$ 2.85 | 45.3 $\pm$ 2.16 | 31.6 $\pm$ 2.41 | 40.2 $\pm$ 2.82 | 63.4 $\pm$ 8.29     | 53.1 $\pm$ 2.16 | 71.8 $\pm$ 0.22 |
| CMU-MOSI   | 50.7 $\pm$ 1.99 | 51.7 $\pm$ 0.87 | 39.0 $\pm$ 2.77 | 41.5 $\pm$ 1.77 | 49.8 $\pm$ 0.77     | 61.0 $\pm$ 1.11 | 67.7 $\pm$ 1.87 |
| CMU-MOSEI  | 53.9 $\pm$ 2.74 | 54.0 $\pm$ 1.68 | 44.4 $\pm$ 2.01 | 52.6 $\pm$ 0.74 | 67.7 $\pm$ 0.04     | 64.6 $\pm$ 0.22 | 77.2 $\pm$ 2.60 |
| Flickr     | 70.4 $\pm$ 1.88 | 69.3 $\pm$ 2.74 | 50.9 $\pm$ 3.11 | 68.8 $\pm$ 1.12 | 76.1 $\pm$ 0.94     | 80.7 $\pm$ 1.76 | 83.4 $\pm$ 1.23 |
| Instagram  | 65.5 $\pm$ 1.01 | 60.0 $\pm$ 1.34 | 45.0 $\pm$ 2.00 | 67.4 $\pm$ 1.26 | 72.3 $\pm$ 0.78     | 78.0 $\pm$ 2.44 | 82.7 $\pm$ 0.91 |
| F&I        | 56.7 $\pm$ 1.58 | 61.7 $\pm$ 1.11 | 53.4 $\pm$ 2.78 | 60.1 $\pm$ 0.57 | 63.8 $\pm$ 1.44     | 68.9 $\pm$ 0.94 | 72.8 $\pm$ 0.06 |

Table 3.12 shows the complete summary of these architectures. The top-1 accuracy and top-5 accuracy denote the performance of the model on the ImageNet dataset. Table 3.13 shows a comparison between different models based on five-fold cross-validation accuracy on all the visual sentiment datasets. To reduce the variance, each experiment is performed five times. The *mean  $\pm$  standard deviation* of the accuracies are reported in Table 3.13.

As evident from the Table 3.13, the top-performing CNN for the VSA task is NASNet. However, in Table 3.12, we see that ResNet-50 has shown better results than VGG-16 and VGG-19 architectures on the ImageNet dataset. Surprisingly, the results obtained for VSA doesn't convey the same. Despite being more sophisticated than VGG-16 and VGG-19 architectures, ResNet-50 has shown the worst performance on visual sentiment datasets. In Twitter I dataset, the maximum accuracy achieved by ResNet-50 is around 53%, whereas the minimum accuracy on VGG-16 and VGG-19 is 68%. The same behavior is observed on CMU-MOSI, Flickr, Instagram, and F&I datasets. However, in ArtPhoto and CMU-MOSEI dataset, VGG-16 and VGG-19 perform better than Inception-V3 as that latter overfits the data and gives less accurate results. For further analysis, we have compared the training efficiency of different architectures on all the datasets. Table 3.14 shows the results in *seconds*. NASNet has taken the highest training time on every dataset, and ResNet-50 takes the minimum time, whereas, in F&I emotion dataset, InceptionV3 takes the minimum time. However, as seen in Table 3.13, ResNet-50 has performed poorly for visual sentiment classification. Thus, it is crucial to study the trade-off between accuracy and training time for selecting the best architecture.

**Table 3.14 Training efficiency of all architectures (Time/Epoch) (sec)**

| Dataset    | VGG-16 | VGG-19 | ResNet-50 | Inception-V3 | Inception-Resnet-V2 | Xception | NASNet |
|------------|--------|--------|-----------|--------------|---------------------|----------|--------|
| 3-agree    | 516    | 533    | 485       | 742          | 796                 | 759      | 1031   |
| 4-agree    | 501    | 566    | 483       | 723          | 795                 | 770      | 1041   |
| 5-agree    | 544    | 500    | 488       | 710          | 767                 | 764      | 1046   |
| Twitter II | 498    | 512    | 511       | 718          | 787                 | 1030     | 1098   |
| Artphoto   | 336    | 418    | 195       | 224          | 411                 | 409      | 1100   |
| CMU-MOSI   | 490    | 471    | 544       | 658          | 704                 | 958      | 1177   |
| CMU-MOSEI  | 511    | 504    | 497       | 610          | 621                 | 744      | 1244   |
| Flickr     | 710    | 729    | 702       | 854          | 942                 | 1024     | 1300   |
| Instagram  | 628    | 714    | 647       | 904          | 1011                | 1297     | 1845   |
| F&I        | 604    | 611    | 694       | 684          | 710                 | 984      | 1475   |

The experimental results suggest that the deep learning-based methods can outperform low-level and mid-level based approaches for VSA. The advantage of using a pre-trained model is that the simplicity of the model makes it easy to learn and train according to the requirements. Moreover, by combining the local and global features in the images, the final discriminative features are captured by the RA-DLNet architecture, which helps in VSA.

## 3.2 A unified framework of deep networks for genre classification using movie trailer

We propose a novel deep affect-based movie trailer classification framework. We also develop an EmoGDB dataset, which contains 100 Bollywood movie trailers annotated with popular movie genres: Action, Comedy, Drama, Horror, Romance, Thriller, and six different types of induced emotions: Anger, Fear, Happy, Neutral, Sad, Surprise. The affect-based features are learned via ILDNet architecture trained on the EmoGDB dataset. Our work aims to analyze the relationship between the emotions elicited by the movie trailers and how they contribute in solving the multi-label genre classification problem. The proposed novel framework is validated by performing cross-dataset testing on three large scale datasets, namely LMTD-9, MMTF-14K, and ML-25M datasets. Extensive experiments show that the proposed algorithm outperforms all the state-of-the-art methods significantly, as reported by the precision, recall, F1 score, precision–recall curves (PRC), and area under the PRC evaluation metrics.

### 3.2.1 Proposed EmoGDB Dataset

To further enhance the research in the area of movie genre detection, especially for Indian cinema, we developed an EmoGDB dataset, which is specifically related to detecting the genre of Hindi Bollywood movies. The major work which has contributed to study the relationship between the field of psychology and cinematography is [177]. Following their work, we adopt six emotion categories, namely: Happy, Surprise, Anger, Sad, Fear, and Neutral. These are the major emotions that are evoked while watching any movie. The prime advantage of this dataset lies in the fact that it is labeled with five-movie genres along with the six different types of emotions (which are elicited while watching a movie trailer) corresponding to each genre. To the best of our knowledge, no dataset has been developed in the literature that provides such rigorous affective-based information for different movie genres. The trailers have a broad range of release dates (1996 to 2020). Figure 3.8 shows some sample images from our dataset (movie name: genre). As seen in the figure, each genre is provoking a wide range of



emotions.



**Figure 3.8** Sample images from EmoGDB dataset (a) 1920: Horror (b) Ae Dil Hai Mushkil : Romance (c) Gameover : Thriller (d) Behen Hogi Teri : Comedy (e) Chhichhore : Drama (f) Baahubali 2\_The Conclusion : Action

We browsed and collected a list of famous Bollywood movies of different genres from four popular film libraries: IMDB, NetFlix, Hotstar, and Amazon Prime. We only focused on those movie trailers which have a common genre on each of these libraries. Our dataset is created and structured to allow the research community to use it with ease. We created one folder per movie. The file structure of the dataset per movie is illustrated in Figure 3.9 (a).

Each movie folder contains two sub-folders of uncropped facial frames and cropped facial frames, along with the corresponding movie trailer. The naming format of each folder and sub-folder is shown in Figure 3.9 (b). The cropped frame information is stored in the CSV file together with the output labels. The CSV contains the following information: Frame\_Name, Movie\_Name, Genre, and Emotion. The final trailers belong to the following genres, namely: Action (17), Comedy (16), Drama (17), Horror (17), Romance (16), and Thriller (17). The reason to limit our work to these genres is that mostly all the movies can roughly be classified into at least one of these genres.

Secondly, we propose that these genres elicit strong induced emotions, which can be crucial for classifying these movie trailers into multiple genres. EmoGDB dataset consists of roughly over 1,00,000 frames corresponding to 100 Bollywood movie trailers. Since the length of a Bollywood film is very long (up to 2-3hrs), hence we concentrated on extracting the features from the movie trailer, which typically has a duration of around 2-4 mins.

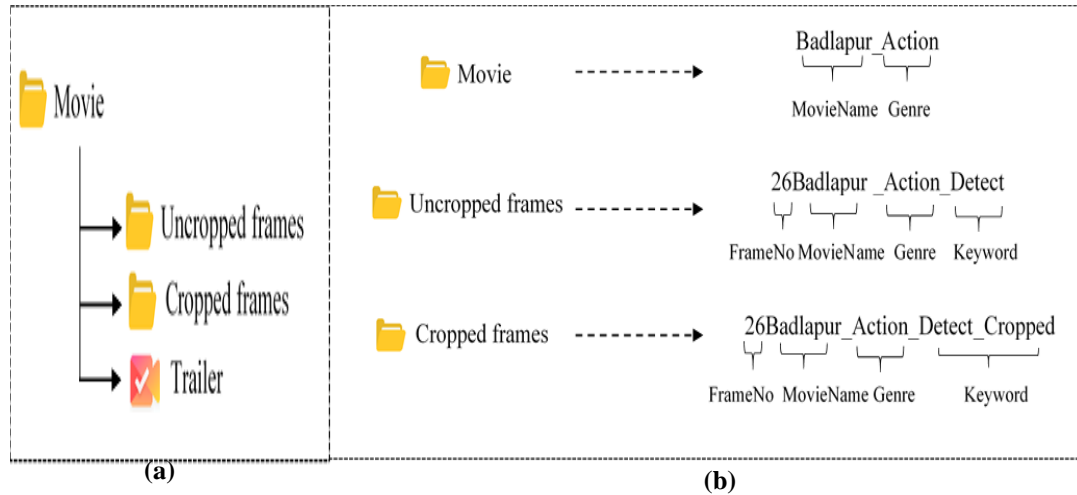


Figure 3.9 (a) Folder structure of the proposed dataset (b) Naming format rules for folders and sub-folders

There may be only one emotion that dominates the entire video, but other emotions could also make interspersed appearances [178], which makes it crucial to analyze the segments of the videos to engender the overall emotions from them. Hence, we start by extracting the frames from the videos. The facial expressions directly depict certain emotions, so we focused on those frames which have faces in them to learn the emotions from the facial expressions. The rest of the frames were ignored. We used OpenCV based Deep learning network, which uses ResNet [179] architecture as a backbone. It is based on the Single Shot Multibox detector [180], which handles objects of different sizes. The input frame is preprocessed before passing it to the deep network. First, we perform mean subtraction and scaling on the input frames to combat the illumination changes. We have used mean values ( $\bar{X}$ ) of Red (R), Green (G), Blue (B) channels as 124.96, 115.97, and 106.13, respectively. The scaling factor  $\sigma$  defaults to 1. The mean is subtracted and scaling factor is divided from each input channel to get the final R, G, B values as follows:

$$R = R - \overline{X_r} \Rightarrow R = (R - 124.96) / 1 \quad (3.7)$$

$$G = G - \overline{X_g} \Rightarrow G = (G - 115.97) / 1 \quad (3.8)$$

$$B = B - \overline{X_b} \Rightarrow B = (B - 106.13) / 1 \quad (3.9)$$

The input is then passed to our OpenCV based deep network which extracts the facial images by calculating the confidence of detection ( $c_r$ ), to filter out the weak detections. If  $c_r > 50\%$ , the corresponding face is extracted from the image, and further detections are checked on the same image. The same process is repeated for all the input frames. The advantage of using this method for facial detection is that it works for tiny faces and can handle substantial occlusion in the images. It can detect different face orientations more accurately as compared to other face detection algorithms, as shown in Figure 3.10.



Figure 3.10 Different face orientations and occluded images are captured by our face detection algorithm.

Table 3.15 and Table 3.16 gives the list of abbreviations and symbols, respectively, along with their meanings, which have been used in this section.

Table 3.15 List of Abbreviations with their meanings

| Abbreviations | Meanings                                    |
|---------------|---|
| EmoGDB        | Emotion-based Genre Detection for Bollywood |
| ILDNet        | Inception-LSTM based Deep Network           |
| LMTD          | Labeled Movie Trailer Dataset               |
| MMTF          | Multifaceted Movie Trailer Feature          |
| ML            | MovieLens                                   |
| ReLU          | Rectified Linear unit                       |
| CNN           | Convolutional Neural Network                |
| LSTM          | Long short term memory                      |
| Bi-LSTM       | Bi-directional LSTM                         |
| PRC           | Precision-Recall curve                      |
| AU(PRC)       | Area under the Precision-Recall curve       |

**Table 3.16 List of symbols with their meanings**

| Symbols                            | Meanings   |
|------------------------------------|--|
| $X_r, X_g, X_b$                    | Mean of red, green, blue channel, respectively   |
| $c_r$                              | Confidence of detection  |
| $\sigma$                           | Scaling factor   |
| $x_1^1$                            | First frame of first genre   |
| $S$                                | Set of discarded frames  |
| $X_f^1$                            | Set of final input frames of first genre   |
| $\sum_j X_f^j$                     | Complete Training set  |
| $\vec{h}_t^{(i)}, \vec{h}_t^{(i)}$ | Hidden layers for the forward and backward pass, respectively                                  |
| $\vec{W}^{(i)}, \vec{W}^{(i)}$     | Weights for the forward and backward pass, respectively  |
| $\vec{b}^{(i)}, \vec{b}^{(i)}$     | Bias for the forward and backward pass, respectively   |
| $\hat{y}_t$                        | Final classification score of LSTM model   |
| $p_i$                              | Probability of an $i^{th}$ sample  |
| $G^1$                              | Matrix showing the relation between video frames and emotions elicited by them for first genre |
| $A_{:,e_1}^{g_1}$                  | Vertical cross-section of all frames in genre $g_1$ for emotion $e_1$                          |
| $E_m$                              | Stacked ensemble model   |

### 3.2.2 Proposed Methodology

In this section, we discuss the proposed deep affect-based movie trailer classification framework in detail. Figure 3.11 shows the pipeline of the proposed framework. The major steps are: (i) fetching and preprocessing the frames (ii) extracting the high-level features from the frames with the help of ILDNet architecture (iii) incorporating several emotions to develop a novel multi-label genre detection theory. The upcoming sections discuss the training and testing phases of the proposed framework with respect to the above steps. To the best of our knowledge, this is the first work that incorporates several emotions for predicting the genre of a movie trailer.

#### 3.2.2.1 Data pre-processing

Initially, we process a sequence of video frames from EmoGDB dataset, defined as  $X_i^1 = \{x_1^1, x_2^1, x_3^1, \dots, x_{i+n-1}^1\}$  where  $x_1^1$  denotes the first frame of genre 1. As discussed in Section 3.2.1, the face detection algorithm extracts the facial expressions from the input frames and discards the remaining structures. Let  $S = \{x_2^1, x_4^1, \dots, x_{i+n-4}^1, x_{i+n-2}^1\}$  denotes the set of discarded frames from genre 1, then the set of final input sequence  $X_f^1 = \{X_i^1 - S\}$ . The same steps are repeated for all the remaining five genres, which gives us the training set of  $\sum_j X_f^j = \{X_f^1, X_f^2, X_f^3, X_f^4, X_f^5\}$  frames.

Moreover, to increase the training set and feed the model with different variants of an image, we perform data augmentation techniques, which include horizontal flipping, zooming, rescaling, and shearing. The same pre-processing steps are repeated for the test set. The input frames from the test set are fetched to perform normalization, followed by the extraction of facial frames by calculating the confidence of detection, as discussed in Section 3.2.1.

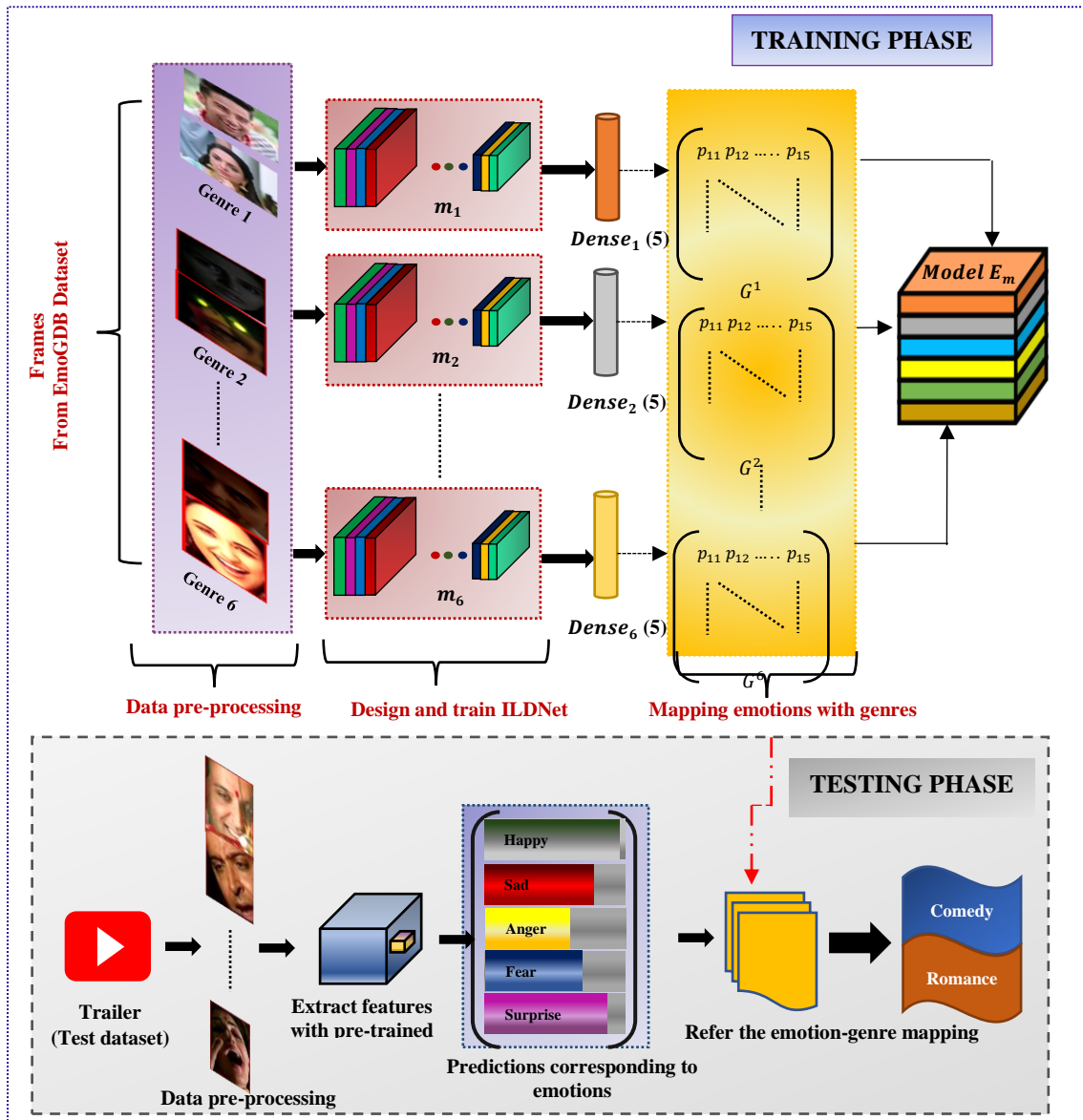


Figure 3.11 Pipeline of the proposed framework

### 3.2.2.2 Inception-LSTM based Deep Network (ILDNet) architecture:

The training phase discusses the proposed ILDNet architecture, which extracts the spatial and temporal features from the video frames, and the testing phase applies this

pre-trained architecture on the test set. The following section discusses both these phases.

**A. Design and Train ILDNet architecture:**

To design an affect-based theory for genre detection, the crucial features from the input frames  $\sum_j X_f^j$ , needs to be learned for each genre. This is achieved by developing a deep learning-based feature extraction ILDNet architecture for each genre, which is a combination of Inception V4 [181], Bi-LSTM, and LSTM layers for extracting the spatial and temporal information from the frames. The classification layers of ILDNet architecture are shown in Table 3.17.

**Table 3.17 Parameters of ILDNet architecture**

| Layer Name : Type                    | Description                       | Input Shape   | Output Shape  | Parameters # |
|--------------------------------------|-----------------------------------|---------------|---------------|--------------|
| Input_1 : Input Layer                | Input image of 299*299            | (299, 299, 3) | (299, 299, 3) | 0            |
| Inception_v4 : CNN                   | Inception_v4 block                | (299, 299, 3) | (1,1001)      | 1538537      |
| bidirectional_1 : Bidirectional LSTM | Bidirectional LSTM with 512 units | (1, 1001)     | (1,1024)      | 6201344      |
| lstm_2 : LSTM                        | LSTM layer with 128 units         | (1, 1024)     | (1,128)       | 590336       |
| dropout_2 : Dropout                  | Dropout with 0.5                  | (1,128)       | (1,128)       | 0            |
| activation_150 : Activation          | ReLu activation function          | (1,128)       | (1,128)       | 0            |
| my_dense_1 : Dense                   | Dense Layer with 5 neurons        | (1,128)       | (1,5)         | 1161         |

The input shape of 299\*299\*3 is passed to the Inception V4 block to obtain a high dimensional representation of the input in the form of [1\*1001] convolution features. The prime motivation for using the Inception V4 model is that it processes the information from different scales, which can capture large size variations of the spatial features from the images. This is achieved by performing parallel filter operations on the input from previous layers by using multiple receptive field sizes for convolutions (1\*1, 3\*3, 5\*5). Moreover, the depth of the network is preserved by using 1\*1 convolutions before the 3\*3 and 5\*5 convolutions, which prevents to increase the computational cost of the network. This can be visualized in Figure 3.12.

As seen in Figure 3.12, an image with  $112*112*64$  ( $h*w*d$ ) dimensions is convolved using  $1*1$  convolutions 32 times for projecting the depth to lower sizes resulting in  $112*112*32$  image. This reduces the depth of an image from 64 to 32 without changing the spatial dimensions. In this way, the inception module can utilize multiple smaller convolution kernels, which could capture the local information in the image without blowing up the computational complexity significantly. The final feature

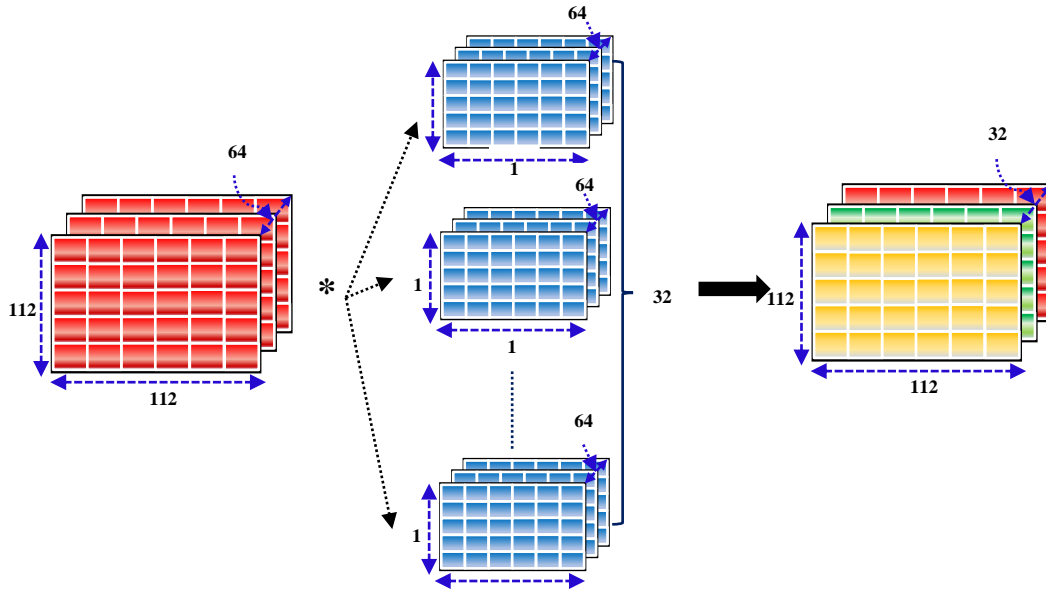


Figure 3.12 Applying  $1*1$  convolutions for reducing the depth of an image without altering the spatial dimensions

vector of  $[1*1001]$  is transformed linearly and passed into the Bi-LSTM layer with a 512-dimensional hidden state to model the correlation of images over time. It exploits the past and future dependencies for a given prediction.

LSTM belongs to the category of RNN, which are popular deep networks for modeling sequential data. However, they suffer from the vanishing gradient problem [182]. Hence, [183] addressed this problem by developing the LSTM architecture. Thereafter, several variants were proposed by refining the classical LSTM architecture. The core idea behind these variants is that they contain a memory cell capable of maintaining the cell's state over time, along with a non-linear gating mechanism to control the flow of information to and from the cell. We discuss some of the popular variants of LSTM by highlighting their fundamental properties. For detailed knowledge regarding these networks, we refer [184] to the interested reader, which provides a comparative analysis of all the networks.

1. **Vanilla LSTM** [183]: This classic version of LSTM includes only the input and output gates. The input gate decides what information should be added to the cell state based on the current input. The output gate decides which part of the cell state should be passed on as the output. Hence, these networks can remember information over a long period of time by passing the information from each gate's input state to the next state.
2. **LSTM with forget gate** [185]: The next popular variant of LSTM includes the forget gate, which controls the extent to which the value of the old cell's state will be discarded based on the current input.
3. **LSTM with peephole connections** [186]: In this variant, peephole connections were added by which the cell can control the gates. This is achieved by adding some direct connections from the cell's state to all the different gates.
4. **GRU** [187]: They combine the input gate and forget gate into a single update gate. They didn't use the peephole connections or an output activation function. Additionally, their output gate was changed into the reset gate, which decides how much past information the network will forget.
5. **Bi-directional RNN (Bi-RNN or BRNN)** [188]: They are made up of two RNN units connected in opposite directions, i.e., going from left to right (forward) and other going from right to left (backward), simultaneously. In this way, these networks can easily preserve the information from both the past and future. The units in Bi-RNN could be standard RNN, LSTM, or GRU cells.

Apart from these architectures, other variants of LSTM includes Depth based gated LSTM [189], clockwise RNN [190], full gradient version LSTM [191], etc. We have used Bi-LSTM in our work. As discussed, in Bi-LSTM, the network maintains two hidden states, one for the left to right (forward) propagation and other for the right to left (backward) propagation. The output from layer  $(i - 1)$  at timestep  $t$  becomes the input to the next neuron at layer  $i$ . Let  $x_t^i$  be the input to the  $i^{th}$  layer at instance  $t$ , the flow of information from the Bi-LSTM layer is depicted in Eq. (3.10) to (3.12):

$$\vec{h}_t^{(i)} = \sigma \left( \vec{W}^{(i)} x_t^{(i-1)} + \vec{U}^{(i)} h_{t-1}^{(i)} + \vec{b}^{(i)} \right) \quad (3.10)$$

$$\overleftarrow{h}_t^{(i)} = \sigma \left( \overleftarrow{W}^{(i)} x_t^{(i-1)} + \overleftarrow{U}^{(i)} h_{t+1}^{(i)} + \overleftarrow{b}^{(i)} \right) \quad (3.11)$$



$$\hat{y}_t = g(W_y[\vec{h}_t^{(i)} + \overleftarrow{h}_t^{(i)}] + b_y) \quad (3.12)$$

where  $\vec{h}_t^{(i)}, \overleftarrow{h}_t^{(i)}$  are the hidden layers for the forward and backward pass, respectively.  $\vec{W}^{(i)}, \vec{U}^{(i)}, \overleftarrow{W}^{(i)}, \overleftarrow{U}^{(i)}$  are the weights and  $\vec{b}^{(i)}, \overleftarrow{b}^{(i)}$  are bias. The final classification score  $\hat{y}_t$  is calculated by combining the scores of both the hidden layers. The obtained feature vector of [1\*1024] from Bi-LSTM is passed into the LSTM unit with a 128-dimensional hidden state. In this way, the temporal information of the images is modeled over time, and the model can learn the temporal relations among the frames. Finally, we add a combination of dropout layer (0.5 rate) and ReLU (Rectified Linear unit) activation to overcome the overfitting problem, followed by a dense layer with the same number of neurons as the number of classes in the datasets. The softmax layer is attached in the end, which gives the probability distribution of size [1 \* 5]. The probability of an  $i^{th}$  sample is given in Eq. (3.13):

$$p_i = \frac{e^{s_i}}{\sum_{j=1}^n e^{s_j}} \quad (3.13)$$

Where,  $s_i$  denotes the score =  $f(x_i, w)$  for  $i^{th}$  sample. Thus, in the training phase, we train six different genre models, namely  $m_1, m_2, m_3, m_4, m_5, m_6$  with their corresponding images from the *EmoGDB* dataset. Finally, we concatenate these models to develop a stacked ensemble model  $E_m$ , as shown in Figure 3.11 (Training Phase). We use categorical cross-entropy loss as our objective function.

### **B. Applying the pre-trained model $E_m$ on test set**

The pre-trained stacked ensemble model  $E_m$  is used for testing the new movie trailers and generate the probability predictions for different emotions evoked from the test set frames.

#### **3.2.2.3 Emotion-genre based theory**

This section discusses how the emotion-genre based theory is developed in the training phase and referred by the test set for multi-label genre classification.

### A. Mapping emotions with movie genres

As discussed in Section 3.2.2.2 (A), the softmax classifier outputs the probability distributions for each emotion class corresponding to every genre. Thus, if the trailer of the first genre (e.g., action) is passed to our ILDNet architecture, which is trained on EmoGDB dataset, the softmax classifier will give a probability distribution  $p_1, p_2, \dots, p_5$  for each of the five emotion classes. The same process will be repeated for every input frame  $f_1, f_2, \dots, f_n$  to yield the total emotions expressed by the movie trailer. This generates a  $2 \times 2$  matrix, which shows the relationship between the frames of the video and emotions elicited by them. Mathematically, this can be represented in the form of a matrix  $G^1$  for the first genre, as given in Eq. (3.14):

$$G^1 = \begin{matrix} & \begin{matrix} e_1 & e_2 & \dots & e_5 \end{matrix} \\ \begin{matrix} f_1 \\ f_2 \\ \vdots \\ f_n \end{matrix} & \begin{pmatrix} p_{11} & p_{12} & \dots & p_{15} \\ p_{21} & p_{22} & \dots & p_{25} \\ \vdots & \vdots & \ddots & \vdots \\ p_{n1} & p_{n2} & \dots & p_{n5} \end{pmatrix} \end{matrix} \quad (3.14)$$

$\downarrow$   $A_{:,e_1}^{g_1}$                        $\downarrow$   $A_{:,e_5}^{g_1}$

where  $e_1, e_2, e_3, e_4, e_5$  represents the five emotions (Anger, Fear, Happy, Joy, and Surprise),  $f_1, f_2, \dots, f_n$  are the processed frames from the *EmoGDB* dataset,  $A_{:,e_1}^{g_1}$  represents the vertical cross-section of  $g_1$  for emotion  $e_1$ . Thus,  $p_{11}$  indicates the amount of emotion  $e_1$  (let's say, surprise) expressed in frame  $f_1$  and so on. Similarly,  $A_{:,e_1}^{g_1}$  indicates the combined prediction of surprise emotion for all the frames of the action movie trailer.

We have removed the neutral emotion for training the ILDNet architecture because experimental results in Section 3.2.3.2 show that it increases the misclassification rate. Hence, the mean probabilities (predictions) of the five emotions for  $k^{th}$  genre is represented as in Eq (3.15):

$$G^k = \{A_{:,e_1}^{g_k}, A_{:,e_2}^{g_k}, A_{:,e_3}^{g_k}, A_{:,e_4}^{g_k}, A_{:,e_5}^{g_k}\} \quad (3.15)$$

In the training phase, the same process is repeated for the remaining five genres

to get the probabilities corresponding to each emotion. Based on the above predictions ( $G^k$ ), we establish a relationship between emotions evoked while watching a trailer and its corresponding genre. Thus, for action genre Eq. (3.15) will be represented as:

$$G^{action} = \{A_{:,surprise}^{g_{action}}, A_{:,sad}^{g_{action}}, A_{:,happy}^{g_{action}}, A_{:,fear}^{g_{action}}, A_{:,anger}^{g_{action}}\} \quad (3.16)$$

$$\text{or, } G^{action} = \{0.18, 0.05, 0.13, 0.12, 0.52\} \quad (3.17)$$

Similar procedure is adopted for the remaining genres. These results are visualized in Figure 3.13, which shows how much each of the above five emotions contributes to different movie genres. Hence, from the figure, we can conclude that emotions play a crucial role in detecting the multiple genres of a movie trailer. The dominant emotions in action, comedy, drama, horror, romance, and thriller genres are anger, happy, sad, fear, happy, and sad, respectively. The results are quite intuitive. However, we observe that the thriller genre mostly involves sad and fear emotions.

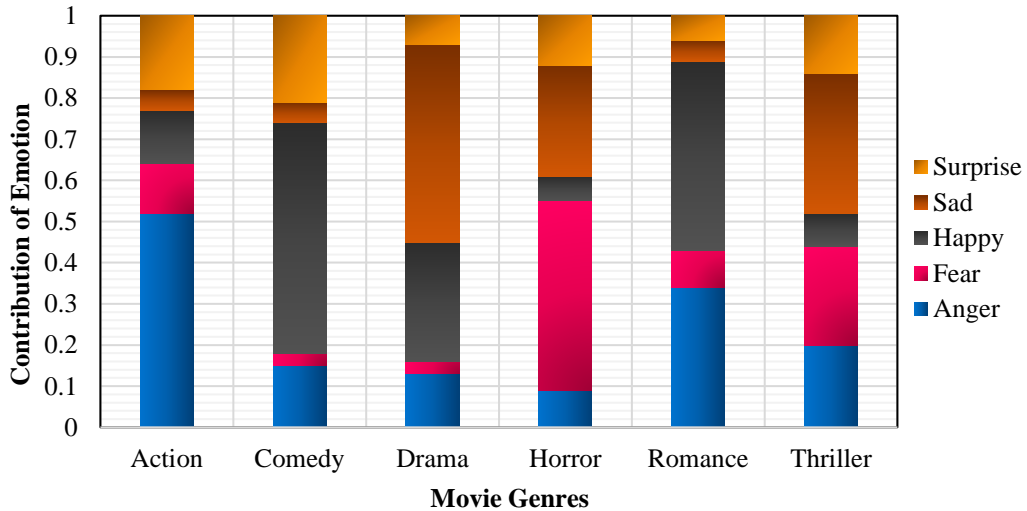


Figure 3.13 Emotion-genre mapping for classifying the movie-trailer into multiple genres

### B. Predicting movie genres

The testing phase involves applying this emotion-genre based mapping theory for calculating the genres corresponding to the emotions predicted in Section 3.2.2.2 (B). This outputs all the genres for the input trailers. The steps discussed above in the training and testing phase of our proposed framework are summarized in Algorithm 1 and Algorithm 2, respectively.

|  |
|--|
| <p><b>Algorithm 1</b> Training phase of deep affect-based movie trailer classification framework</p> <p><b>Input:</b> Sequence of movie trailer frames from <i>EmoGDB</i> dataset for all the six genres.</p> <p><b>Output:</b> Stacked ensemble model (<math>E_m</math>) along with mean probability predictions of each emotion corresponding to the six genres.</p> <p><b>1. Data preprocessing:</b></p> <p>    <b>1.1</b> Fetch the training frames for Genre 1.</p> <p>    <b>1.2</b> Perform various data argumentation techniques to increase the training set.</p> <p><b>2. Design and train ILDNet architecture:</b></p> <p>    <b>2.1</b> Extract the spatial and temporal features from the input frames by ILDNet architecture.</p> <p>    <b>2.2</b> Apply the softmax classifier to get the final probability predictions.</p> <p><b>3.</b> Repeat steps 1 and 2 for all the six genres. This gives us six different models (one for each genre),<br/>namely <math>m_1, m_2, m_3, m_4, m_5, m_6</math>.</p> <p><b>4.</b> Concatenate the above models to generate a stacked ensemble model, <math>E_m</math>.</p> <p><b>5. Mapping emotions with movie genres:</b></p> <p>    <b>5.1</b> Get the probability predictions from Step 2.2.</p> <p>    <b>5.2</b> Compute the vertical cross-section of genre 1 with each of the five emotions (neutral emotion is discarded), to generate the affect-based genre predictions.</p> <p>    <b>5.3</b> Repeat steps 5.1 and 5.2 for rest of the genres to create the affect-based genre mapping.</p> |
|--|

|   |
|---|
| <p><b>Algorithm 2</b> Testing phase of deep affect-based movie trailer classification framework</p> <p><b>Input:</b> Sequence of movie trailers from different test datasets.</p> <p><b>Output:</b> Multilabel genre predictions corresponding to the movie trailers.</p> <p><b>1. Data preprocessing:</b></p> <p>    <b>1.1</b> Extract the input frames from the movie trailers.</p> <p>    <b>1.2.</b> Normalize the input frames by performing mean subtraction and scaling.</p> <p>    <b>1.3.</b> Calculate the confidence of detection (<math>c_r</math>) for each input frame.</p> <p>    <b>1.4.</b> If <math>(c_r) &gt; 50\%</math>, extract the face from the input frame to build the test set for our model corresponding to the first trailer, else discard the frame.</p> <p><b>2. Apply pre-trained <math>E_m</math> on test set:</b></p> <p>    <b>2.1</b> Use model <math>E_m</math> (step 4 of Algo 1) to generate the probability predictions for different emotions evoked from the test set frames.</p> <p><b>3. Predicting movie genres:</b></p> <p>    <b>3.1</b> Apply the affect-based genre theory developed in the training phase for calculating the genres corresponding to the emotions predicted in Step 2.1 above.</p> <p>    <b>3.2</b> Output the multiple genres for the input trailer.</p> <p><b>4.</b> Repeat the above steps for all the trailers in the test set.</p> |
|---|

### 3.2.3 Experiments

In this section, we validate our proposed framework by evaluating it on several datasets and reporting the classification results in terms of several evaluation metrics. Finally, we discuss the computational complexity of our model, followed by implementing the class activation maps to visualize the prominent image regions captured by our architecture.

#### 3.2.3.1 Datasets

We validate the proposed ILDNet architecture by performing cross-datasets testing on

three publicly available large-scale datasets, namely LMTD-9 (*EmoGDB* → *LMTD-9*), MMTF-14K (*EmoGDB* → *MMTF-14K*), and ML-25M (*EmoGDB* → *ML-25M*) datasets. The complete dataset details are as follows:

**A. LMTD-9 (Labelled Movie Trailer) dataset:**

LMTD [192] [193] is one of the large-scale datasets for movie trailer-based genre classification. It consists of 10k movie trailers from 22 different genres. However, for multilabel classification, we consider its subset, LMTD-9 [194], which includes around 4k movie trailers from 9 genres. LMTD-9 removes the trailers which were released before 1980 and contains more than 6500 frames. Since our work focuses on six prominent genres, hence we consider the movie trailers from six genres, namely: Action (853), Comedy (1558), Drama (2023), Horror (435), Romance (649), and Thriller (692). Each of the movie trailers is assigned to at least one and at most three genres. This dataset is challenging because it contains high variability of video features, which includes image quality, aspect ratio, and total length.

**B. MMTF-14K (Multifaceted Movie Trailer Feature) dataset:**

MMTF-14K [195] dataset contains 13,623 Hollywood movie trailers links from 18 different genres. This dataset is primarily used for developing content-based unimodal and multimodal recommender systems. Hence, it addresses three descriptors, namely: Metadata descriptors (Genre features and Tag features), audio descriptors (Block level and I-Vector features), and video descriptors (Aesthetic and AlexNet features). These descriptors help the MMTF-14K dataset to support other multimedia tasks like multilabel genre classification, tag prediction, popularity prediction. We use this dataset for multilabel genre classification tasks and focus on movie trailers from six genres, as discussed above. The Youtube link of the Hollywood movie trailers is parsed to download the trailers. Since some of the trailers are not available on Youtube now, hence we are left with 8,674 movie trailers. Each of the movie trailers is assigned to at least one and at most five genres. Hence, this dataset supports the five-class genre classification.

### C. ML (MovieLens) 25M dataset:

ML-25M [196] is a benchmark dataset for movie recommender systems. It contains star ratings and text tagging activity from MovieLens<sup>1</sup>. Moreover, this dataset contains 62,423 movie trailer IDs, where each trailer is assigned three different IDs each from MovieLens, MovieDB<sup>2</sup>, and IMDB<sup>3</sup>. We crawl and download the trailers from the IMDB website only. Since this dataset contains trailers from the year 1900-2019, some of the trailers were not available on all the three platforms (MovieLens, MovieDB, IMDB). Moreover, we focus on multi-label classification for six genres only. Hence, we discard the trailer from other genres. Thus, the total movie trailers available for testing are 18,150. Each of the movie trailers is assigned to at least one and at most five genres. Hence, this dataset also supports the five-class genre classification. Further, this dataset contains a wide range of trailers from different cinemas like the Cinema of Denmark, Cinema of U.S.A, Cinema of India, Cinema of Japan, Cinema of South Korea, etc.

#### 3.2.3.2 Experiment Setup

This section gives the implementation details and discusses the classification results along with baseline comparison on several state-of-the-art and alternate methods.

#### A. Implementation Details

We build and implement the proposed architecture in Python on popular deep learning framework Keras using Tensorflow backend. All the experiments were performed on Windows 10, 64-bit machine with 128GB RAM using NVIDIA Titan RTX GPUs. Adam optimizer is used with default parameters of  $\beta_1 = 0.9$  and  $\beta_2 = 0.999$ , learning rate = 0.001, and batch size of 64. To feed our model with more amount of data, we perform data augmentation by using different augmentation techniques: horizontal flipping, zooming, rescaling, and shearing. In the training phase, the input samples for

---

<sup>1</sup> <https://movielens.org>

<sup>2</sup> <https://www.themoviedb.org>

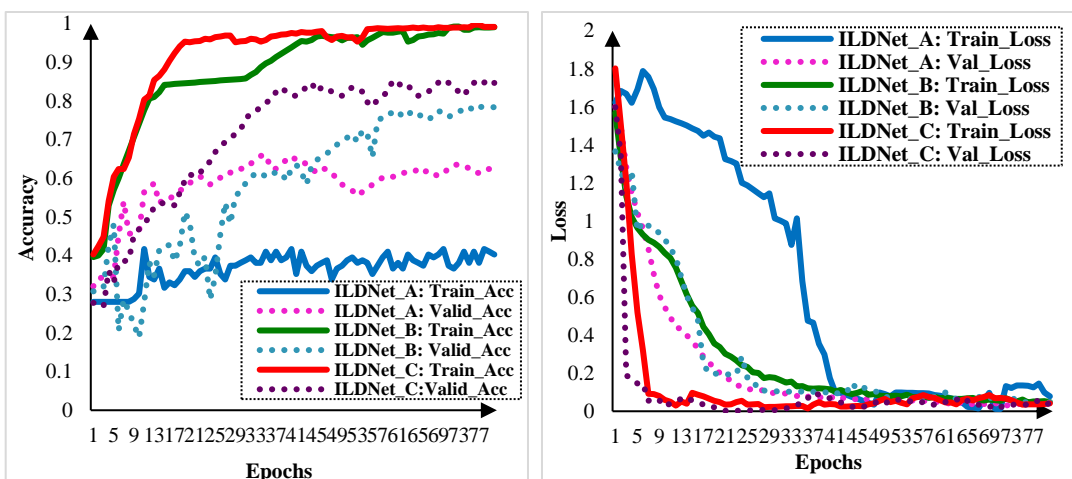
<sup>3</sup> <http://www.imdb.com>

each of the six different genre models are split into 80% training, 5% validation, and 15% testing samples. The model is trained for 200 epochs with early-stopping when the validation accuracy does not improve for 20 consecutive epochs. The model converges within 80 epochs. The model achieving the highest validation accuracy is picked as the best-trained model. This model gives the probability distribution for generating the emotion-genre based mapping for each genre from the 15% testing samples of the *EmoGDB dataset*. In the testing phase, we show the generalizability of the proposed architecture by performing cross-datasets testing on the above-discussed test datasets.

## B. Classification Results

Figure 3.14 (a) shows the training accuracy + validation accuracy and (b) training loss + validation loss curves to evaluate the training and validation process of ILDNet architecture. We perform experiments with three variations (ILDNet\_A, ILDNet\_B, and ILDNet\_C) within the ILDNet architecture to show the contribution of the major layers in our architecture:

- ✓ **ILDNet\_A**: This model is composed of only the Inception\_v4 module.
- ✓ **ILDNet\_B**: This model is composed of Inception\_v4 module + Bi-directional LSTM.
- ✓ **ILDNet\_C**: This is our proposed model which is composed of Inception\_v4 module + Bi-directional LSTM + LSTM + Dropout + Activation.



**Figure 3.14** Evaluating the performance of different variations of ILDNet architecture with (a) Training accuracy + Validation accuracy curves (b) Training Loss + Validation Loss curves

All the above three models are trained and validated on the proposed EmoGDB dataset. The model achieving the best validation results is selected as the final model for cross-dataset testing on other test datasets. From Figure 3.14, we see that the ILDNet\_A model shows the sign of underfitting as the model is not able to learn the features from the input data (stagnant training accuracy) and still achieves validation accuracy of around 60%. The same behavior can be analyzed from the loss curves of the ILDNet\_A model. The second model, ILDNet\_B, is more complex than the first one, as the combination of Inception\_v4 block and Bi-LSTM can learn the input features properly, thus removes the underfitting issue of ILDNet\_A model. Finally, the best results are shown by the ILDNet\_C model, which confirms the adequate learning and validating aspect of the model. This shows the importance of LSTM layers for modeling the temporal information in the input.

**Table 3.18 Classification results of ILDNet on LMTD-9, MMTF-14K, and ML-25M datasets (P: Precision, R: Recall, F1: F1 score)**

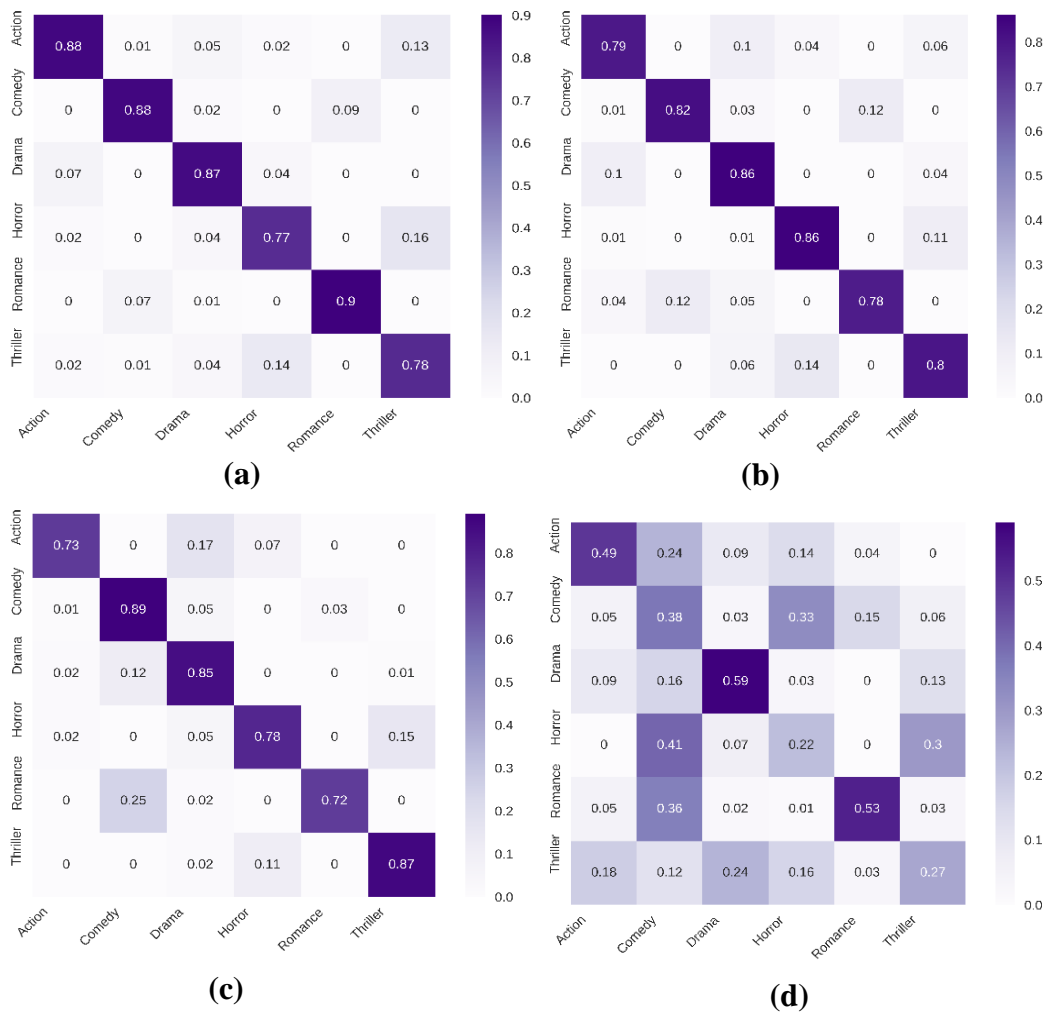
| Dataset | Action |     |     | Comedy |     |     | Drama |     |     | Horror |     |     | Romance |     |     | Thriller |     |     |
|---------|--------|-----|-----|--------|-----|-----|-------|-----|-----|--------|-----|-----|---------|-----|-----|----------|-----|-----|
|         | P      | R   | F1  | P      | R   | F1  | P     | R   | F1  | P      | R   | F1  | P       | R   | F1  | P        | R   | F1  |
| LMT     | 80.    | 88. | 84. | 95.    | 87. | 90. | 92.   | 87. | 89. | 62.    | 77. | 68. | 78.     | 90. | 84. | 85.      | 78. | 81. |
| D-9     | 7      | 2   | 2   | 0      | 1   | 9   | 1     | 6   | 7   | 0      | 0   | 6   | 6       | 7   | 1   | 0        | 3   | 5   |
| MMT     | 69.    | 79. | 74. | 92.    | 82. | 87. | 89.   | 86. | 87. | 76.    | 86. | 81. | 76.     | 78. | 77. | 78.      | 80. | 79. |
| F-14K   | 4      | 6   | 1   | 7      | 5   | 3   | 1     | 3   | 6   | 6      | 2   | 0   | 9       | 3   | 6   | 2        | 8   | 4   |
| ML-     | 57.    | 73. | 63. | 81.    | 89. | 85. | 83.   | 84. | 84. | 89.    | 78. | 83. | 48.     | 71. | 57. | 69.      | 87. | 77. |
| 25M     | 9      | 6   | 2   | 0      | 8   | 1   | 8     | 8   | 3   | 0      | 2   | 1   | 3       | 9   | 7   | 3        | 0   | 0   |

We perform extensive experiments to evaluate the performance of the proposed ILDNet on LMTD-9, MMTF-14K, and ML-25M datasets. Table 3.18 shows the classification results in terms of precision, recall, F1 score for each of the output genre classes in the test datasets. The combined accuracy of ILDNet architecture on LMTD-9, MMTF-14, and ML-25M is 86.15%, 83.06%, and 85.3% respectively. From Table 3.18, we can conclude that the architecture can successfully classify the different genre classes and performs well on all the genres. The best results are seen in Comedy and Drama genres. We also show the confusion matrix in Figure 3.15 (a) – (c) for all the three test datasets.

We notice that misclassification occurs only between comedy and romance genre along with Horror and Thriller genre. We see many comedy frames were being misclassified as romance frames and vice-versa. Similar behavior is observed in Horror and Thriller genres. The intuition behind this is that the comedy and romance genres are majorly dominated by Happy emotions, whereas the Horror and Thriller genres are



dominated by fear and sad emotions. This results in the high inter-class similarity between the respective genre classes as the frame features are not easily distinguishable among them. Despite this, the results indicate that our architecture can correctly recognize and classify different movie genres with great performance. Since our proposed *EmoGDB* dataset has six emotions corresponding to every genre, hence we initially experimented by training ILDNet on all the six emotions, namely: Anger, Fear, Happy, Neutral, Sad, and Surprise.



**Figure 3.15 Confusion matrix for (a) LMTD-9 (b) MMTF-14K (c) ML-25M dataset generated with model trained on five emotions (d) Confusion matrix for LMTD-9 dataset generated with model trained on six emotions (including neutral emotion)**

However, we found that a majority of frames were getting misclassified into the neutral category, thus increasing the misclassification results in the test datasets. We also found that this category is not conveying any vital information about the genre of

the movie. As an example, we show the confusion matrix for LMTD-9 dataset in Figure 3.15 (d), which is computed by training ILDNet on all the six emotions. From the figure, it is evident that neutral emotion is increasing the misclassification results and decreases the performance of the classifier. Hence, we remove the neutral emotion for training the ILDNet architecture.

### C. Baseline Comparison

We compare the performance of proposed architecture with previous movie genre classification methods on LMTD-9, MMTF-14K, and ML-25M datasets, as shown in Table 3.19. Since the datasets have imbalanced classes and vary in size, we validate the performance of our multilabel classifier by comparing the Area under the Precision-Recall curve  $AU(PRC)$  for each genre class. We combine the contribution of all the genre classes to calculate the micro-average scores for each dataset. This measure can adequately capture the noise resulting from the class imbalance problem in multilabel classification. The final performance across the datasets is evaluated by comparing the micro-average  $AU(PRC)$  metric, which is a stricter measure for validating a multilabel classification problem.

**Table 3.19 Comparison of ILDNet architecture with previous works using micro-average  $AU(PRC)$  metric**

| Method \ Dataset       | (EmoGDB →<br>LMTD-9) | (EmoGDB →<br>MMTF-14K) | (EmoGDB →<br>ML-25M) |
|------------------------|----------------------|------------------------|----------------------|
| Low-level + SVM [197]  | 0.31                 | 0.29                   | 0.16                 |
| GIST + KNN [198]       | 0.46                 | 0.48                   | 0.31                 |
| CENTRIST + KNN [198]   | 0.49                 | 0.55                   | 0.39                 |
| w-CENTRIST + KNN [198] | 0.48                 | 0.59                   | 0.53                 |
| CoNNeCT [193]          | 0.78                 | 0.60                   | 0.57                 |
| CNN-Motion [192]       | 0.41                 | 0.46                   | 0.44                 |
| CTT-MMC-C [194]        | 0.62                 | 0.58                   | 0.66                 |
| CTT-MMC-TN [199]       | 0.76                 | 0.81                   | 0.73                 |
| <b>ILDNet</b>          | <b>0.81</b>          | <b>0.94</b>            | <b>0.89</b>          |

As seen in Table 3.19, we validate our work with state-of-the-art methods and several alternate approaches for movie genre classification. For extracting the low-level features, we compute the four video features as described in [197], namely, average shot length, color variance, motion content, and lighting key, followed by SVM

classification. We extract the GIST, CENTRIST, and w-CENTRIST feature descriptors from keyframes of the trailers with the same parameters as discussed in [198] for each of the six movie genres. These are the state-of-the-art methods in low-level feature extraction for movie genre classification. For building CoNNeCT [193] architecture, we combined five different ConvNets models, each one designed to capture various features of the videos. The models are same as in [193], except that we apply GoogleNet architecture, which we pretrain on the LMTD-9 dataset. The CNN-Motion-S [192] extracts the video features using CNN architectures based on [200] and MFCC audio features. The CTT-MMC-C [194] extracts the video features using [179] and applies 2D convolution on them. Similarly, CTT-MMC-TN [21] extends the work of CTT-MMC-C [194] by fusing the video features with audio features extracted by spectrograms. From Table 3.19, we can conclude that our proposed ILDNet architecture surpasses the low-level state-of-the-art methods by 30% and high-level approaches by 5% - 16% on different datasets.

As discussed above, we report the performance of our architecture by visualizing the precision-recall curves for each output genre class in Figure 3.16. We also compute AU(PRC) to compare the performance across the datasets. The best results of 0.81, 0.94, and 0.89 are obtained for LMTD-9, MMTF-14K, and ML-25M datasets, which reaffirms the adequate learning of the ILDNet architecture. The architecture can perform well across the movie trailers of different cinemas, despite being trained on the dataset of Indian cinema.

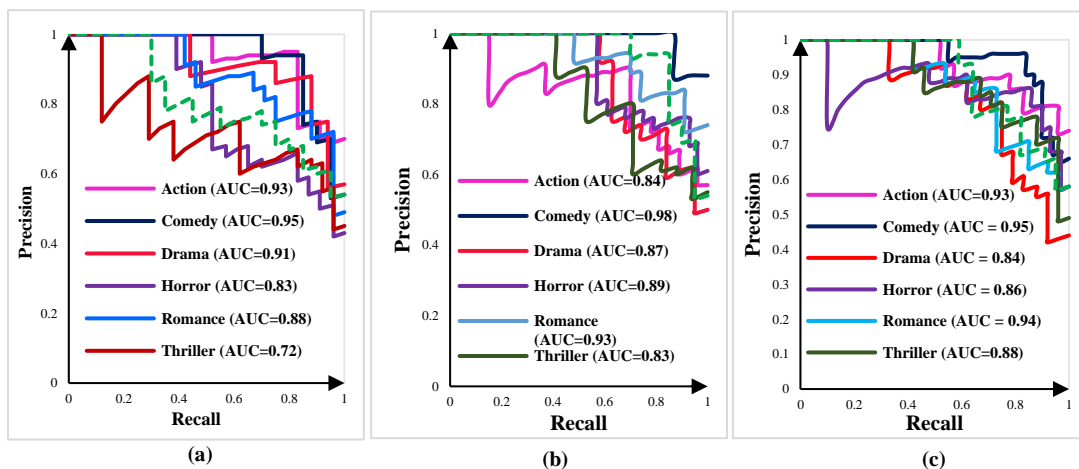


Figure 3.16 Precision-Recall curves for (a) LMTD-9 (b) MMTF-14K, and (c) ML-25M datasets

#### D. Computational complexity

To evaluate the efficiency of our framework, we report the running time to classify a youtube video trailer on Windows 10, 64-bit machine with 128GB RAM using NVIDIA Titan RTX GPUs, after training our network. Our testing phase involves the following steps:

- ✓ Computing the facial frames from the videos, which takes 1.8 seconds.
- ✓ Extracting the spatial and temporal features, which takes 12.3 seconds.
- ✓ Computing the emotion labels and referring to the emotion-genre mapping for predicting the final genre labels of the videos. This process is finished in 2 seconds.

Thus, the proposed framework requires 16.1 seconds for a 3min long video. We compare our results with other popular methods in movie genre classification. The results are summarized in Table 3.20.

**Table 3.20 Time complexity of movie genre classification methods**

| Method           | Time (s)    |
|------------------|-------------|
| CoNNeCT [192]    | 10.6        |
| CNN-Motion [193] | 24.3        |
| CTT-MMC-C [194]  | 18.6        |
| CTT-MMC-TN [199] | 25.2        |
| <b>Ours</b>      | <b>16.1</b> |

As seen in Table 3.20, our framework runs efficiently than other competitors. Although CoNNeCT [192] shows less running time of 10.6 seconds, still trading off the performance of CoNNeCT (Refer Table 3.19) with our proposed framework makes these results acceptable.

#### E. Visualization

We also provide qualitative results in Figure 3.17. We visualize the discriminative image regions captured by the ILDNet architecture for identifying different emotions in the movie trailers by showing the gradient-based class activation maps [201] for comedy, horror, and action genre. We can easily see that ILDNet can detect the relevant regions in the images which evoke certain kinds of emotions. Thus, each of the genres elicits different types of emotions, which validates the fact that there exists a strong

correlation between the emotions and movie genre, justifying the motivation of our proposed work. This proves that induced emotions contribute significantly to identify and classify the genres of movie trailers.

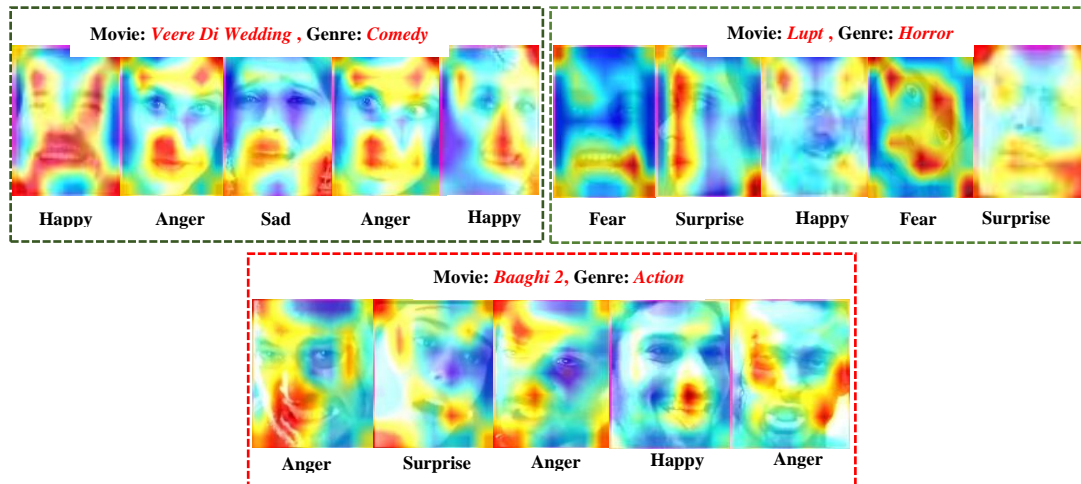


Figure 3.17 Visualizing the discriminative image regions captured by the model for identifying different emotions in the movie trailers

### 3.3 Significant Outcomes

In this chapter, firstly, a framework is developed to extract the visual sentiments and emotions elicited by the images posted on social media. Then, we discussed another framework for affect-based movie trailer classification by mapping the human emotions to the structure of the movie trailers. The observations are summarized as follows:

- The proposed RA-DLNet architecture learns the spatial hierarchies of image features along with attention-aware features that change adaptively as layers go deeper and deeper. Our CNN architecture is a combination of feedforward series of stacked inception-like modules called cells (Normal cell and Reduction cell). The prime advantage of these cells is that they can handle the images of arbitrary spatial dimensions and filter depth.
- The residual attention block enhances the vital features by suppressing the noise without significant information loss. Hence, the model can capture different types of attention extensively, which refines the feature maps gradually, thus guiding the feature learning process, as validated by class activation maps and colormaps.

- A novel EmoGDB dataset is developed, which contains 100 Bollywood movie trailers in six popular and distinct genres: Action, Comedy, Drama, Horror, Romance, Thriller. The entire dataset is labeled with six induced emotions: Anger, Fear, Happy, Neutral, Sad, Surprise corresponding to every movie genre. This dataset focuses on studying the relationship between the field of psychology and cinematography.
- The proposed ILDNet captures the correlation between the movie genre and the evoked emotions, which will provide feedback to the directors about the viewer's perspective.
- A novel idea in the field of affect-based video classification is developed, where we contribute by successfully establishing and validating the relationship between psychology and cinematography. The prime advantage of ILDNet is that without watching the entire movie trailer, the architecture can classify the trailer into multiple movie genres.

*This chapter is based on the following works:*

- ✓ **Ashima Yadav**, Dinesh Kumar Vishwakarma. "A deep learning architecture of RA-DLNet for visual sentiment analysis." *Multimedia Systems*, 26 (2020): 431-451 (**Impact Factor: 1.563**). (Pub: Springer).
- ✓ **Ashima Yadav**, Dinesh Kumar Vishwakarma. "A unified framework of deep networks for genre classification using movie trailer." *Applied Soft Computing*, 96 (2020): 106624 (**Impact Factor: 5.472**). (Pub: Elsevier).

# Chapter-4 Multimodal Sentiment Analysis

---

In this chapter we target the multimodal data to address the issues involved in multimodal sentiment classification by exploring the complicated correlations between the visual and textual data simultaneously.

## 4.1 A Deep Multi-Level Attentive network for Multimodal Sentiment Analysis

Multimodal SA has attracted increasing attention with broad application prospects. The existing methods focus on a single modality, which fails to capture the social media content for multiple modalities. Moreover, in multi-modal learning, most of the works have focused on simply combining the two modalities without exploring the complicated correlations between them. This resulted in dissatisfying performance for multimodal sentiment classification. Motivated by the status quo, we propose a Deep Multi-level Attentive network (DMLANet), which exploits the correlation between image and text modalities to improve multimodal learning. Specifically, we generate the bi-attentive visual map along the spatial and channel dimensions to magnify CNN's representation power. Then we model the correlation between the image regions and semantics of the word by extracting the textual features related to the bi-attentive visual features by applying semantic attention. Finally, self-attention is employed to automatically fetch the sentiment-rich multimodal features for the classification. We conduct extensive evaluations on four real-world datasets, namely, MVSA-Single, MVSA-Multiple, Flickr, and Getty Images, which verifies the superiority of our method.

### 4.1.1 Proposed Methodology

This section presents the details of the proposed DMLANet. In Section 4.1.1.1, we give an outline of the proposed network. Then we present the visual attention module in Section 4.1.1.2, which generates significant bi-attentive visual features by utilizing channel attention and spatial attention. Finally, Section 4.1.1.3 discusses a joint

attended multimodal learning process that learns a combined representation for textual and visual features by applying semantic attention, which measures the semantic closeness of text and visual features, followed by a self-attention mechanism that extracts the crucial multimodal features for sentiment classification.

#### 4.2.1.1 Framework Overview

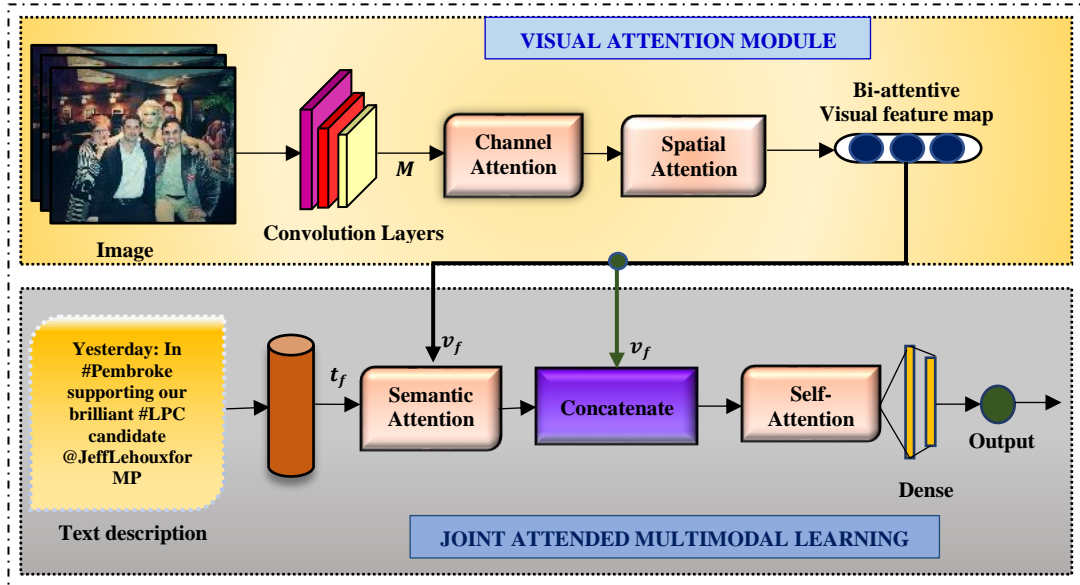


Figure 4.1 Block diagram of the proposed DMLANet

Let  $D$  represent the given set of documents. For each document  $d \in D$ , let  $I = \{I_1, I_2, \dots, I_n\}$  denote the set of images for the visual component of the document and  $T = \{T_1, T_2, \dots, T_n\}$  denote the set of text descriptions or the sequence of sentences for the textual component of the document. Each of the sentence  $T_i$  is composed of a sequence of  $w_i, i \in [1, S]$ . Each document is further associated with one of the following sentiment labels: positive, negative, and neutral (Flickr and Getty datasets are labeled with positive and negative sentiments only). Thus, the objective is to predict the sentiment labels on the unseen documents by training the network on the training corpus.

Figure 4.1 shows the block diagram of the proposed framework. In the *visual attention module*, we employ channel-based attention, which enhances the information-rich channels, and spatial or region-based attention, which further concentrates on the emotional regions based on attended channels to get the bi-attentive visual feature map.



In *joint attended multimodal learning*, semantic attention is applied to measure the emotional words related to the bi-attentive visual features. Next, we combine the attended word features and bi-attentive visual features and pass them to the self-attention block, which automatically highlights the important multimodal features. These features are then passed to the classifier for the sentiment classification.

#### 4.2.1.2 Visual Attention Module

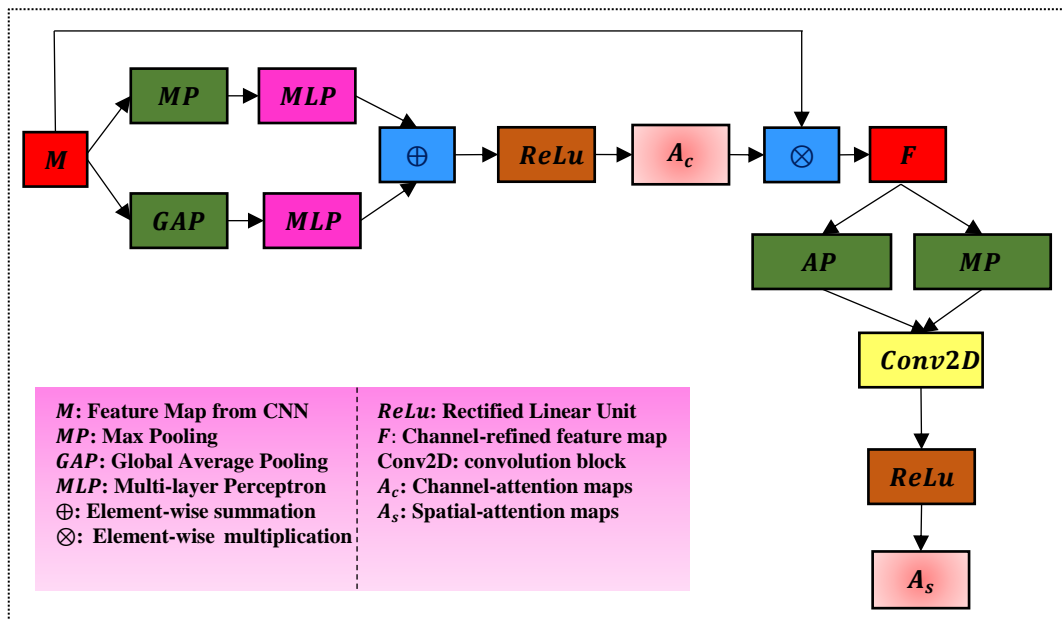


Figure 4.2 Block diagram explaining the Visual Attention Module

Recently, attention networks have shown significant performance in many computer vision tasks [202]. They make CNN learn and focus on the crucial information by suppressing unnecessary information, thus improving the overall classification performance. We achieve this by sequentially generating the bi-attentive map along the spatial and channel dimensions separately to magnify CNN's representation power. This approach was popularly used for the task of object detection [203]. However, in multimodal sentiment classification, most of the previous works have ignored the channel dimension for obtaining the visual features. This is important because channel-based attention concentrates on the information-rich channels, i.e., they highlight 'what' are the crucial elements in the given image. The upcoming sub-sections (A) and (B), and Figure 4.2 describes the entire visual attention module.

### A. Channel Attention

For each image  $I_i$ , we obtain the feature map  $M \in [H * W * C]$  using the Inception V3 [204] network. In channel attention, we apply global average pooling and max average pooling of feature maps to generate the average-pooled and max-pooled features, respectively. Each feature is passed to a multilayer perceptron network with one hidden layer followed by the *ReLU* activation function, and the elements are concatenated to get the final attention map  $A_c = (1 * 1 * 256)$ . The reason to apply *ReLU* over *tanh* is that it converges quickly and results in cheaper computation. The channel attention process can be summarized in Eq. (4.1) as follows:

$$A_c = ReLu [W_1 (W_0 (MP (M))) + W_1 (W_0 (GAP (M)))] \quad (4.1)$$

where,  $W_0, W_1$  are the weights of the multilayer perceptron,  $M$  denotes the feature map,  $MP$  = max-pooling layer,  $GAP$  = global average pooling layer, and  $A_c$  is the channel attention. Hence, channel attention extracts ‘what’ are the meaningful features in a given image by squeezing the spatial dimensions of the feature map using the average and max-pooling layers and merging the output vectors using element-wise summation.

### B. Spatial Attention

The spatial attention map tells ‘where’ is the informative part of the image, i.e., it locates the relevant image regions according to the attended channel-based features. The input feature map  $M$  is element-wise multiplied with the channel attention map  $A_c$  to generate the channel refined feature map  $F$ . The channel refined feature maps are concatenated using average pooled, and max pooled layers and are fed into a convolutional layer with  $7*7$  kernel size to generate the spatial attended features, which we refer to as the bi-attentive visual features as they are the combination of the channel-attended and spatial-attended visual features. The process of converting the channel refined feature map  $F$  into the spatial attended features  $A_s$  can be summarized in Eq. (4.2) as follows:

$$A_s = ReLu [Conv2D (AP (F); MP (F))] \quad (4.2)$$

Finally, we obtain the following sequence of bi-attentive visual features, as shown in Eq. (4.3) below:

$$v_f = \{v_1, v_2, \dots, v_n\}, v_f \in R^{m*d} \quad (4.3)$$

Where,  $m$ = number of regions, and  $d$  = feature dimension of each region.

#### 4.2.1.3 Joint Attended Multimodal learning

Each word  $w_i$  is transformed into a real-valued vector through pre-trained embedding matrix (we use Glove<sup>4</sup> embeddings), and then optimized by applying LSTM network which gives the high-level textual features as  $t_f$ . Existing work fails to detect the sentimental words that are related to the images. However, we address this problem by exploiting the correlation between the words and the different visual features generated in Section 4.1.1.2. We measure the semantic closeness of the textual features with visual content by combining both the features using element-wise multiplication and generating the joint features  $m_f$ . This is shown in Eq. (4.4) as follows:

$$m_f = \tan h (W(v_f \odot t_f)) \quad (4.4)$$

Where  $W$  is the learnable weights. The attention scores are computed, as shown in Eq. (4.5) below:

$$\alpha_f = \frac{\exp(m_f)}{\sum_f \exp(m_f)} \quad (4.5)$$

Finally, we obtain the attended word-level features, which measures the emotional textual features related to the visual features as follows:

$$s_f = \sum_f \alpha_f * t_f \quad (4.6)$$

Next, we concatenate the obtained attended textual features  $s_f$  with the visual

---

<sup>4</sup> <https://nlp.stanford.edu/projects/glove/>

features to obtain the joint-multimodal features  $J_f = (s_f, v_f)$ . Since, in multimodal learning, not all the modalities contribute equally to the classification task [205]. Hence, we apply self-attention networks, which take the multimodal feature vectors as input and automatically identifies the crucial weights corresponding to each modality, as shown in Eq. (4.7) below:

$$v_f = \frac{\exp(\varphi(W * J_f + b))}{\sum_f \exp(\varphi(W * J_f + b))} \quad (4.7)$$

Where  $W$  and  $b$  are the learnable weights, and  $\varphi$  is the activation function.

Thus, in a self-attention network, multiple input modalities are allowed to interact with each other to find the input that gets more attention, which tells the importance of the different multimodal input features in the sequence. The joint attended multimodal features are computed as the weighted average over all the feature sequence, as shown below:

$$M = \sum_f v_f * J_f \quad (4.8)$$

The obtained attended multimodal features  $M$  are passed as an input to the softmax classifier for sentiment classification as follows:

$$P(s) = \text{Softmax}(W_s; M) \quad (4.9)$$

The whole network is trained on a training set by minimizing the cross-entropy loss with backpropagation as follows:

$$\text{Loss} = -\sum \log(P(s), y) \quad (4.10)$$

Where  $y$  is the actual sentiment label of the training data.

### 4.2.2 Experiments

In this section, we conduct several experiments to confirm the efficacy of our DMLANet on popular real-world datasets and report the quantitative and qualitative

results.

#### 4.2.2.1 Datasets

We collected four large-scale, real-world datasets from various social media platforms for conducting the multimodal sentiment classification. Table 4.1 shows the complete statistics of each dataset. Further, the datasets are explained as follows:

- ✓ **MVSA (Multi-View Sentiment Analysis Dataset):** The MVSA dataset [206] consists of two separate datasets. MVSA-Single contains 5129 image-text pairs from Twitter, where each pair is labeled by a single annotator. MVSA-Multiple consists of 19600 image-text pairs, which are labeled by three annotators. The actual sentiment is calculated by taking the majority vote out of the three sentiments (positive, negative, and neutral) for each modality separately. In both cases, the annotator's judgment for the text and image sentiment label is independent. However, many tweets may result in inconsistent textual and image sentiment label. We follow the following rules to deal with inconsistent sentiment labels between different modalities: The tweets with one positive label and one negative label or vice-versa are removed. If the tweet has one positive (or negative) label and other neutral labels, then the final multimodal sentiment is positive (or negative). Finally, we get 4511 image-text pairs for MVSA-Single and 17024 image-text pairs for MVSA-Multiple datasets, respectively.
- ✓ **Flickr:** We collect the image-text pairs from the Flickr website by using the 1200 ANPs, as described in [207]. The images were weakly labeled according to the sentiment of the ANP into the positive and negative sentiment category only. We also collect the English descriptions associated with the images. The images with too short text (<5 words) and too long text (>100 words) were removed. Thus, we obtained a dataset of 276,571 weakly labeled image-text pairs.
- ✓ **Getty Images:** Getty Images is a supplier of videos, photos, music having relatively formal text descriptions, which can be conveniently browsed by the users. Similar to [208], we query Getty images with 101 sentimental keywords

from the Balanced Affective Word List Project<sup>5</sup> to download the image with their corresponding text description. The downloaded image-text pairs were weakly labeled as per the sentiment keywords into the positive and negative sentiment category, giving us 453,289 image-text pairs.

**Table 4.1 Overall Statistics of each Dataset**

| Datasets      | #Positive | #Negative | #Neutral | Total   | Label  |
|---------------|-----------|-----------|----------|---------|--------|
| MVSA-Single   | 2683      | 1358      | 470      | 4511    | Strong |
| MVSA-Multiple | 11318     | 1298      | 4408     | 17024   | Strong |
| Flickr        | 129317    | 147254    | -        | 276,571 | Weak   |
| Getty Images  | 235732    | 217557    | -        | 453,289 | Weak   |

#### 4.2.2.2 Implementation Details

The proposed DMLANet is implemented in Python using Keras deep learning framework. The experiments were performed on a 64-bit Windows 10 machine with 128 GB RAM and NVIDIA Titan-RTX GPUs. We set the learning rate = 0.001, batch size = 256 with Adam optimizer. Dropout is used to avoid overfitting. We performed experiments via a five-fold cross-validation strategy. The datasets are split in the 80:10:10 ratio for training, validation, and testing sets, respectively. The final accuracy is calculated by averaging the results across each of the test fold. The model achieving the highest validation accuracy is selected for the testing phase.

#### 4.2.2.3 Results and Analysis

In this section, we validate the proposed model on all four datasets, as shown in Figure 4.3. We use the following evaluation metrics: Precision, Recall, F1 score, and accuracy to validate our model. All the evaluation metrics are ranged from 0 to 100%, where the higher the value of the metrics, the better is the performance of the model. For multi-class classification, the average F1 score and accuracy are 79.59% and 79.47%, respectively, for MVSA-Single, and 75.26%, and 77.89%, respectively, for MVSA-Multiple. For binary-class classification, the average F1 score and accuracy are 89.19% and 89.30%, respectively, for Flickr, 92.60%, and 92.65% respectively, for Getty

<sup>5</sup> <http://www.sci.sdsu.edu/CAL/wordlist/origwordlist.html>.

images.

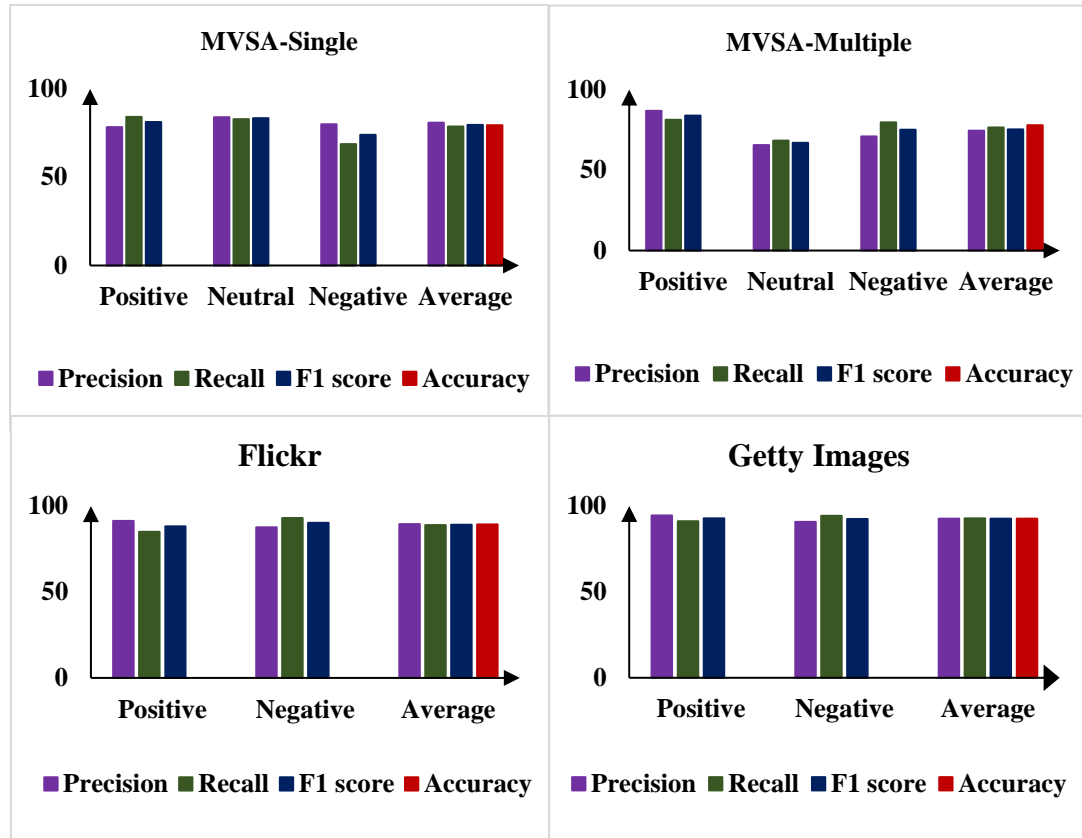


Figure 4.3 Experimental results on the datasets (%)

Since we have imbalanced samples in the dataset, we used ROC (Receiver operating characteristics) and PRC (Precision-Recall curves) to further validate our model's performance. The ROC curves in Figure 4.4 (a) show that our model has shown increased TPR (True Positive rates) on all the datasets. Similarly, AUC (Area under ROC curves) helps to compare the different ROC curves better. The highest value of AUC is 94.46%, which is achieved by Getty images. However, the model can still distinguish between the classes for both the binary and multi-class sentiment classification. Compared with ROC, the PRC is more suitable for imbalanced datasets. Hence, we plotted the PRC in Figure 4.4 (b) between the precision and recall values to compare the performance of our model across the datasets. As evident from the curves, the joint attended learning approach in DMLANet has shown effective results for learning the multimodal features for the sentiment classification.

To show the evolution of model's performance, we plot the training and validation curves for the MVSA-Multiple dataset. Figure 4.5 (a) shows that training

and validation loss decreases with increasing data and (b) shows that training and validation accuracy increases with the data. It can be clearly seen that as more and more data is supplied to the model, it can learn the adequate features from the data and finally converges after approximately 50 epochs.

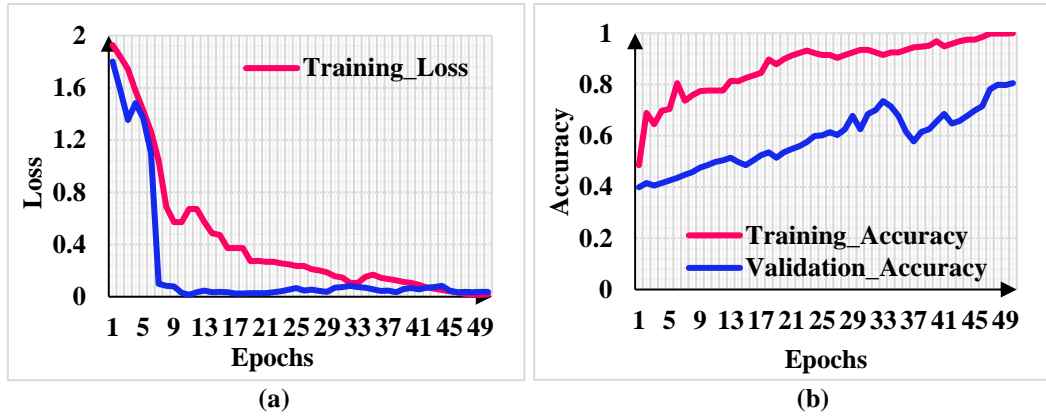


Figure 4.5 (a) Training and Validation Loss curves (b) Training and Validation accuracy curves on MVSA-Multiple Dataset

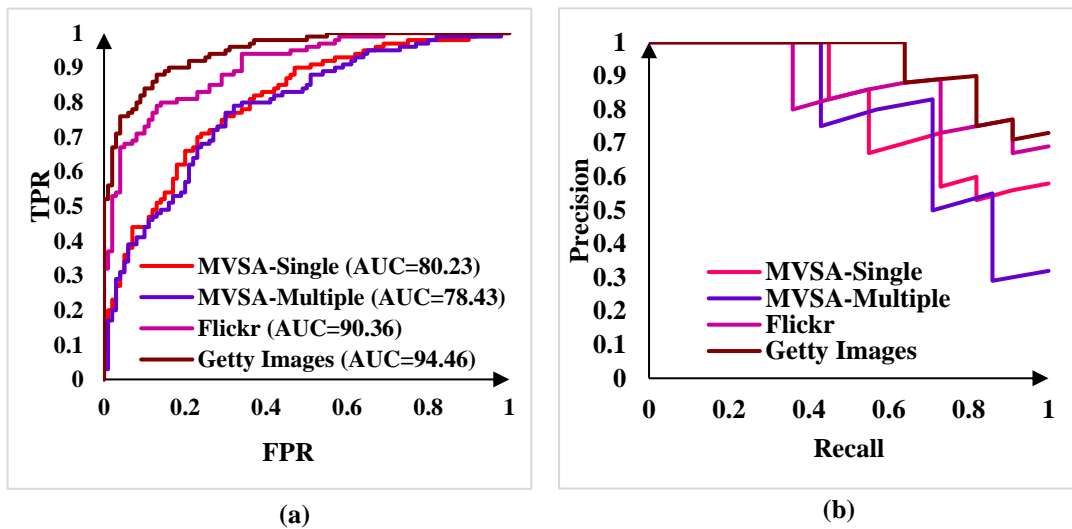


Figure 4.4 (a) ROC curves (b) PRC curves for the datasets

#### 4.2.2.4 Baseline Methods

This section compares our work with state-of-the-art methods for the MVSA-Single, MVSA-Multiple, Flickr, and Getty Images datasets.

##### A. MVSA Datasets:

For MVSA-Single and MVSA-Multiple datasets, the following baselines were used for



comparison:

- ✓ **SentiBank and SentiStrength** [207]: The SentiBank extracts 1200 ANP as mid-level features for image classification, and SentiStrength utilizes grammar and spelling style from the text. Both the techniques are combined to handle the multimodal sentiment classification.
- ✓ **MNN (Merged Neural Network)** [209]: MNN utilizes CNN to extract the multimodal features which are fused by a residual model using early (Early-RMNN) and late fusion (Late-RMNN).
- ✓ **HSAN (Hierarchical Semantic Attentional Network)** [125]: The text-based HAN extracts textual information from tweets, and semantic image features are extracted by CNN-LSTM model. The important words are reflected by using the attention mechanism.
- ✓ **CoMN (Co-Memory Network)** [210]: A stacked co-memory network is developed, which uses text features to capture image feature maps, and image information is utilized for identifying the textual keywords.
- ✓ **MultiSentiNet** [211]: This model extracts the objects and scenes from the image, followed by attention-based LSTM to fetch the textual features. Finally, the features are fused for the final sentiment classification.
- ✓ **FENet (Fusion-Extraction Network)** [212]: It uses Interactive Information Fusion (IIF) mechanism, which applies attention features across both the modalities and Specific Information Extraction layer (SIE), which is based on gated convolution, followed by late fusion for combining both the modalities.

**Table 4.2 Comparison Results of different methods for MVSA Datasets (%)**

| Methods \ Datasets |                           | MVSA-Single  |              | MVSA-Multiple |              |
|--------------------|---------------------------|--------------|--------------|---------------|--------------|
|                    |                           | F1           | Accuracy     | F1            | Accuracy     |
| [207]              | SentiBank & SentiStrength | 50.08        | 52.05        | 55.36         | 65.62        |
| [209]              | Late-RMNN                 | -            | 67.09        | -             | 67.90        |
| [125]              | HSAN                      | -            | 66.90        | -             | 67.76        |
| [210]              | CoMN                      | 70.01        | 70.51        | 68.83         | 68.92        |
| [211]              | MultiSentiNet             | 69.63        | 69.84        | 68.11         | 68.86        |
| [212]              | FENet                     | 74.06        | 74.21        | 71.21         | 71.46        |
| <b>Ours</b>        | <b>DMLANet</b>            | <b>79.59</b> | <b>79.47</b> | <b>75.26</b>  | <b>77.89</b> |

Table 4.2 displays the comparative results for MVSA-Single and MVSA-

Multiple datasets. As compared to the best performing baseline, FENet [212], our model achieves 5% more accuracy for MVSA-Single and 6% more accuracy for MVSA-Multiple dataset. This shows that our multi-level attention contributes to the fine-grained features for sentiment classification. Hence, it is evident that the F1 and accuracy scores of our DMLANet are higher than all the other baselines.

### B. Flickr and Getty images Datasets:

For Flickr and Getty images, we compare our work with the following baselines:

- ✓ **AHRM (Attention-based Heterogeneous Relational Model)** [213]: The visual features are captured by dual attention mechanism, followed by graph convolutional network which combines the social context information.
- ✓ **BDMLA (Bi-directional Multi-level Attention)** [131]: The joint learning is done by two independent networks that learn the visual attention and semantic attention, followed by fusing the modalities with MLP.
- ✓ **AMGN (Attention-based modality Gated networks)** [214]: It utilizes visual and semantic attention models to obtain word-related visual features, followed by gated LSTM to extract more emotional features in the visual and textual modalities.
- ✓ **HDF (Hierarchical Deep Fusion)** [215]: HDF captures the correlations between the image and textual content by using hierarchical LSTM. Late fusion is employed using MLP.
- ✓ **DMAF (Deep Multimodal Attentive Fusion)** [133]: DMAF uses deep CNN to extract the visual features and LSTM based semantic attention for modeling the textual data.
- ✓ **Joint Cross-modal model** [216]: The textual features are computed using attention-based GRU, and visual features are calculated using maximum mean discrepancy. Finally, attention-based LSTM is used to compute the final-sentiment polarity.

**Table 4.3 Comparison Results of different methods for Flickr and Getty images (%)**

| Methods \ Datasets |      | Flickr |          | Getty images |          |
|--------------------|------|--------|----------|--------------|----------|
|                    |      | F1     | Accuracy | F1           | Accuracy |
| [213]              | AHRM | 87.5   | 87.1     | 88.4         | 87.8     |

|             |                         |              |              |              |              |
|-------------|-------------------------|--------------|--------------|--------------|--------------|
| [131]       | BDMLA                   | 84.8         | 84.9         | 86.2         | 86.5         |
| [214]       | AMGN                    | 86.8         | 87.3         | 88.7         | 88.2         |
| [215]       | HDF                     | 86.1         | 85.9         | 88.0         | 88.1         |
| [133]       | DMAF                    | 85.0         | 85.9         | 86.6         | 86.9         |
| [216]       | Joint Cross-modal model | -            | -            | 81.0         | 80.6         |
| <b>Ours</b> | <b>DMLANet</b>          | <b>89.19</b> | <b>89.30</b> | <b>92.60</b> | <b>92.65</b> |

The comparative results on Flickr and Getty images are shown in Table 4.3. The accuracy obtained on the Flickr dataset is 89.30% and on Getty images is 92.65%, which is 4% higher than AMGN [133], which performs best amongst the baselines. We also see that the accuracy and F1 scores on Getty images are higher than Flickr. This may be because, as compared to Flickr, the textual descriptions on Getty are more formal and relevant to the image content. Thus, we can say that our proposed model effectively exploits the correlation between the textual and image modalities for all four datasets.

#### 4.2.2.5 Ablation Study

In this section, we perform an ablation study to evaluate the contribution of each module. We conduct an ablation study on two datasets: MVSA-Multiple (Multiclass and Strongly labeled dataset) and Flickr (Binary class and Weakly labeled dataset). We retrain our model by ablating the following crucial components: Spatial attention (SA) + channel attention (CA), Semantic attention (SMAtt), and Self-Attention (SAtt). The results are shown in Table 4.4.

**Table 4.4 Ablation studies on MVSA-Multiple and Flickr Datasets**

| Datasets      | Model                 | F1 score (%) | Accuracy (%) |
|---------------|-----------------------|--------------|--------------|
| MVSA-Multiple | DMLANet w/o (SA + CA) | 71.29        | 70.85        |
|               | DMLANet w/o (SMAtt)   | 70.17        | 70.00        |
|               | DMLANet w/o SAtt      | 73.98        | 73.54        |
|               | <b>DMLANet</b>        | <b>75.26</b> | <b>77.89</b> |
| Flickr        | DMLANet w/o (SA + CA) | 85.54        | 85.77        |
|               | DMLANet w/o (SMAtt)   | 82.44        | 81.90        |
|               | DMLANet w/o SAtt      | 88.01        | 87.98        |
|               | <b>DMLANet</b>        | <b>89.19</b> | <b>89.30</b> |

✓ **DMLANet w/o (SA + CA)**: This ablated model doesn't use the spatial and channel

attention block. The features obtained from the inception V3 module are directly given to the semantic attention block. This gives a drop in the F1 score and accuracy values for both the datasets, which clearly shows the channel-attended visual features and region-attended features helps in learning the discriminative image features.

- ✓ **DMLANet w/o (SMAtt):** Here, we ablate the semantic attention block and directly concatenate the bi-attentive visual features with the high-level textual features obtained from LSTM. In this case, we observe a significant drop in the performance of the model for both datasets. Around 7% accuracy is dropped for the MVSA-Multiple dataset, and 8% is dropped for the Flickr dataset. These results indicate the importance of semantic attention, which tells how closely the words are linked to the contents of the images. Thus it explores the correlation between both the features of the modalities.
- ✓ **DMLANet w/o SAtt:** In this ablated model, the self-attention module is not used. The joint multimodal features  $J_f$  are directly fed into the dense layer for the final classification. We observe that the F1 score drops by 2% and 1% for MVSA-Multiple and Flickr datasets, respectively. Similarly, the accuracy drops to 73.54% and 87.98% for both datasets. These results also show that it is necessary to focus only on the essential sentiment-rich multimodal features, as not all the features are important for the classifier.

Based on the results in Table 4.4, we conclude that the multi-level attention in the form of channel attention, spatial attention, semantic-attention, and self-attention exploits the correlation between the visual and textual modalities by filtering out the irrelevant and redundant information.



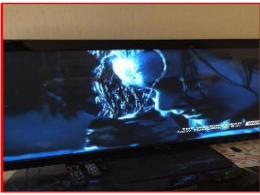
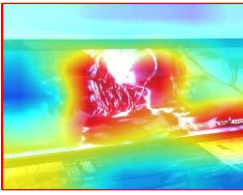


#### 4.2.2.6 Visualization

In this part, we evaluate our proposed model by quantitatively showing the sentiment classification results. We randomly select three positive samples (Figure 4.6 (a)) and three negative samples (Figure 4.6 (b)) from the MVSA datasets. We use gradient-based class activation maps [201] to visualize the visual attention weights, whereas the background color reflects the semantic attention. The brighter the color, the higher is

the attended semantic score. Together, the visual and semantic attention tells “what” our model infers from the image and text sentiment pair.

| Original Image-Text Pair  | Attended Image  | Attended Text   |
|---|---|---|
|  <p>Last night was the first night of The X Factor tour!!! Belfast was an amazing crowd! So excited for tonight!!!</p> |   | <p>Last night was the first night of The X Factor <b>tour</b>!!! Belfast was an <b>amazing crowd</b>! So <b>excited</b> for <b>tonight</b>!!!</p> |
|  <p>Knocked doors with the venerable @kylejpeterson this aft in my hometown, Aurora!</p>                               |   | <p><b>Knocked doors</b> with the <b>venerable</b> candidate kylejpeterson this aft in my <b>hometown</b>, <b>Aurora</b>!</p>                      |
|  <p>Wonderful morning at Mon Sheong meeting seniors who want real change. Thx Martin for the tour!</p>                |  | <p><b>Wonderful</b> morning at Mon Sheong <b>meeting</b> seniors who want <b>real</b> change. Thx Martin for the <b>tour</b>!</p>                 |

(a)

| Original Image-Text Pair  | Attended Image   | Attended Text   |
|---|--|---|
|  <p>Enraged by this? Then spread awareness! Loyalty...betrayed.</p>  |  | <p><b>Enraged</b> by this? Then <b>spread</b> awareness! <b>Loyalty...betrayed</b>.</p>   |
|  <p>Watching Titanic on Valentine's Day while eating a crap ton of chocolate... I'm just... So ecstatic.</p> |  | <p><b>Watching Titanic</b> on Valentine's Day while eating a <b>crap ton</b> of chocolate... I'm just... So <b>ecstatic</b>. Can't you tell??</p> |
|  <p>Dog starved, ignored by neighbors, then thrown from balcony in Murcia, Spain.</p>                        |  | <p><b>Dog starved</b>, <b>ignored</b> by <b>neighbors</b>, then <b>thrown</b> from <b>balcony</b> in Murcia, Spain.</p>                           |

(b)

Figure 4.6 Quantitative analysis of DMLANet for (a) Positive image-text pairs (b) Negative image-text pairs on MVSA Dataset

As seen in Figure 4.6 (a), visual attention is drawn from the image regions by paying attention to more affective regions, which contributes towards the positive sentiment. The semantic attention focuses on words like “amazing”, “excited”, “wonderful”, which conveys the positive sentiment. Similarly, in Figure 4.6 (b), negative sentiment is expressed by focusing on crucial regions and words like “starved”, “crap”, “betrayed”. However, it is difficult to tell the exact text’s sentiment in many cases since a text may contain many sarcastic statements where positive words may sarcastically convey negative sentiments. For e.g., In (b), the second example uses some positive words like “ecstatic”, still it conveys a negative sentiment. However, combining visual attention helps to classify the it as negative.

## 4.2 Significant Outcomes

In this chapter, a multimodal learning process is developed by applying attention mechanisms at various levels from large-scale real-world datasets. The observations are summarized as follows:

- The proposed DMLANet captured the correlation between image and text modalities by extracting only the sentiment-rich multimodal features. The visual attention block applies channel and spatial attention to generate robust bi-attentive visual features. Moreover, joint-attended multimodal learning gives high-quality representations from text and image modalities by focusing on the words that are related to the image contents.
- The multi-level attention in the form of channel attention, spatial attention, semantic-attention, and self-attention enhances the performance of multimodal sentiment classification by filtering out the irrelevant and redundant information in the input data.

*This chapter is based on the following works:*

- ✓ **Ashima Yadav**, Dinesh Kumar Vishwakarma. “A Deep Multi-Level Attentive network for Multimodal Sentiment Analysis.” (*arXiv:2012.08256*).

## Chapter-5 Application of Sentiment Analysis

---

This chapter explores the application of SA in the medical domain by developing a deep language-independent network to analyze the attitude and behavior of the people related to the COVID-19 pandemic.

### 5.1 A Deep Language-independent Network to analyze the impact of COVID-19 on the World via Sentiment Analysis

Towards the end of 2019, Wuhan experienced an outbreak of novel coronavirus, which soon spread all over the world, resulting in a deadly pandemic that infected millions of people around the globe. The government and public health agencies followed many strategies to counter the fatal virus. However, the virus severely affected the social and economic lives of the people. In this paper, we extract and study the opinion of people from the top five worst affected countries by the virus, namely USA, Brazil, India, Russia, and South Africa. We propose a deep language-independent Multilevel Attention-based Conv-BiGRU network (MACBiG-Net), which includes an embedding layer, word-level encoded attention, and sentence-level encoded attention mechanism to extract the positive, negative, and neutral sentiments. The embedding layer encodes the sentence sequence into a real-valued vector. The word-level and sentence-level encoding is performed by a 1D Conv-BiGRU based mechanism, followed by word-level and sentence-level attention, respectively. We further develop a COVID-19 Sentiment Dataset by crawling the tweets from Twitter. Extensive experiments on our proposed dataset demonstrate the effectiveness of the proposed MACBiG-Net. Also, attention-weights visualization and in-depth results analysis shows that the proposed network has effectively captured the sentiments of the people.

#### 5.1.1 Proposed Methodology

In this section, we explain the proposed framework for sentiment classification. It describes the data preprocessing process where we follow a two-step data cleaning

approach, followed by the proposed Multilevel Attention-based Conv-BiGRU network (MACBiG-Net) to learn the semantic information of a sentence by encoding the essential words and sentences in a sequence.

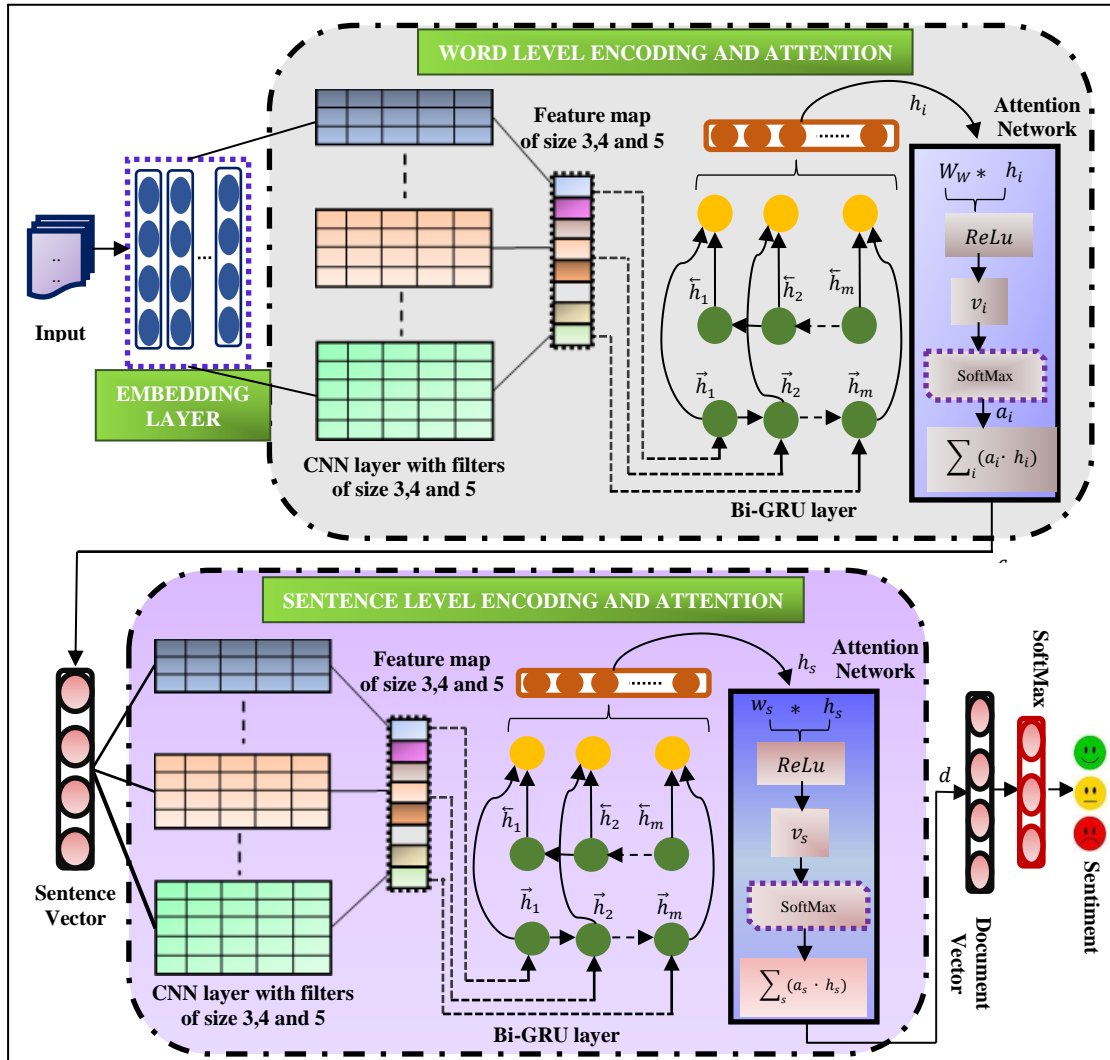


Figure 5.1 Block diagram of the proposed Multilevel Attention-based Conv-BiGRU Network (MACBiG-Net)

### 5.1.1.1 Data preprocessing

We scrapped the data from one of the most popular microblogging sites Twitter, for all the five countries based on their location. The complete data scraping, cleaning, and labeling process are discussed in Section 5.1.2.1. The final dataset comprises of people's views belonging to three output classes: positive, negative, and neutral. Since data processing is one of the most crucial tasks for handling the unstructured textual data, hence we meticulously designed a two-step data cleaning approach. The first



approach is followed before translating the data into the English language so that the translation process is not affected by special characters or non-ASCII codes in the sentence. This includes removing hyperlinks, special characters/symbols, retweet headers, Twitter usernames, non-ASCII codes, NaN (Not a Number) character, and whitespaces. Finally, all the uppercase words are converted into lowercase by using the NumPy library functions.

In the second approach, the labeled dataset goes through stemming and lemmatization to reduce the word to its root form. Porter Stemmer and WordNet Lemmatizer are used for stemming and lemmatization, respectively. We also removed the stop words that frequently occur in the text and convey no meaningful information, followed by hashtags and punctuation removal. The cleaned data is then passed to the Multilevel Attention-based Conv-GRU network, as discussed in the subsequent section for the final classification.

### 5.1.1.2 Multilevel Attention-based Conv-BiGRU Network (MACBiG-Net)

The section describes the proposed MACBiG-Net, as shown in Figure 5.1. The network includes three main steps: Embedding Layer, which encodes the input words into the low-dimensional vectors, Word-level Encoding and Attention, in which the words of a sentence are encoded by 1D Conv-BiGRU based representation to obtain the sentence representation, followed by word-level attention, which computes the essential weights for the sentence vector, and Sentence level encoding and attention, where the sentences of a document are again encoded by 1D Conv-BiGRU based representation to obtain the document representation, followed by sentence-level attention to get the document vector for the classification.

#### A. Embedding Layer

Each word in the sequence of the sentence is converted into a real-valued vector. Formally, let  $\{r_1, r_2, \dots, r_n\}$  be the sequence of  $n$  words in a sentence. We first embed each word  $w_i$  into a real-valued vector through a word embedding matrix  $E \in R^{d_r * |V|}$ ,

where  $d_r$  is the word embedding size, and  $V$  is the vocabulary size. We use the Glove<sup>6</sup> embedding matrix to get the final sequence of vector  $\{e_1, e_2, \dots, e_n\}$ , which serves as an input for the next layer.

## B. Word-level Encoding and attention

Our proposed architecture is inspired by [39], which intends to capture the structure of words from the sentence and the structure of sentences from a document. The word encoding is performed by 1D convolution, followed by the Bi-GRU mechanism. We first describe the convolution process as follows:

The local features are extracted by using a 1D CNN network with different kernel sizes to generate a feature map. The different convolution kernels help in learning the various local characteristics of the text. In general, for  $e_i^{th}$  embedding corresponding to  $r_i^{th}$  word, the concatenation operation  $\oplus$  is expressed as follows:

$$e_{1:n} = e_1 \oplus e_2 \oplus \dots \oplus e_n \quad (5.1)$$

The convolution operation is applied on the window of  $j$  words to generate a new feature value  $m_i$  as shown below:

$$m_i = f(W \cdot e_{i:i+j-1} + b) \quad (5.2)$$

Where,  $b$  = bias and  $f$  = Rectified Linear unit activation function (ReLU). This filter is applied to different possible window of words in a sentence which generates the following feature map:

$$M = [m_1, m_2, \dots, m_{n-j+1}] \quad (5.3)$$

We apply the filter of sizes 3, 4, and 5 to obtain multiple features, followed by the max-pooling layer to capture the maximum feature value. The concatenated output of all three filter sizes is passed into the BiGRU based layer, which contains the hidden

<sup>6</sup> <https://nlp.stanford.edu/projects/glove/>

states to store crucial information. The bidirectional nature of GRU is based on the fact that words in a sentence are not only related to the previous words but also to the following words. We have chosen GRU over LSTM due to its efficiency in training, which makes it computationally cheaper than LSTM. The forward GRU with  $\vec{h}_i = \overrightarrow{GRU}(e_i, \vec{h}_{i-1})$  is concatenated with backward GRU,  $\overleftarrow{h}_i = \overleftarrow{GRU}(e_i, \overleftarrow{h}_{i+1})$  to obtain the output of BiGRU at step  $w$  as  $h_i = [\vec{h}_i, \overleftarrow{h}_i]$ . Thus, the BiGRU layer fetches the contextual information of every sentence centered around the word  $w_i$ . This makes sense as the orientation of sentiment needs to consider the past and future context information in a sentence. Thus, the word is encoded by the Conv-BiGRU block by storing the local and contextual features.

The attention mechanism is based on the idea that when users read a document word by word in each sentence, then they would pay attention to the most informative word or sentence. Hence, we model both the word level and sentence level attention differently. In word-level attention, the word annotation  $h_i$  is fed into the network to get its vector representation  $v_i$ . We apply the *ReLU* activation function over *tanh* as specified in [39], as *ReLU* converges quickly and results in cheaper computation. This is described in Eq. (5.4):

$$v_i = ReLu(W_w \cdot h_i + b_w) \quad (5.4)$$

The resulting annotation  $v_i$  is multiplied (dot product) by the word context vector  $v_{wt}$  which is learned during the training. The softmax function is applied to obtain attention weights  $a_i$  as shown in Eq. (5.5) below:

$$a_i = \frac{\exp(v_{wt} \cdot v_i)}{\sum_i \exp(v_{wt} \cdot v_i)} \quad (5.5)$$

Finally, the attention weights  $a_i$  is concatenated with the word annotations to obtain the sentence vector  $c_i$  as described in Eq. (5.6) below:

$$c_i = \sum_i (a_i \cdot h_i) \quad (5.6)$$

The layer details of the word-level encoding and attention step are shown in Table 5.1.

Table 5.1 Layer description of Step 2: Word-level Encoding and Attention

| Layer Name                     | Input Shape   | Output Shape     | # Parameters | Connected to   |
|--------------------------------|---|------------------|--------------|--|
| Input Layer                    | (None, 200)   | (None, 200)      | 0            | -  |
| Embedding                      | (None, 200)   | (None, 200, 100) | 1835200      | Input Layer  |
| Conv1D_1                       | (None, 200, 100)  | (None, 198, 128) | 38528        | Embedding  |
| Conv1D_2                       | (None, 200, 100)  | (None, 197, 128) | 51328        | Embedding  |
| Conv1D_3                       | (None, 200, 100)  | (None, 196, 128) | 64128        | Embedding  |
| MaxPooling1D_1                 | (None, 198, 128)  | (None, 66, 128)  | 0            | Conv1D_1   |
| MaxPooling1D_2                 | (None, 197, 128)  | (None, 65, 128)  | 0            | Conv1D_2   |
| MaxPooling1D_3                 | (None, 196, 128)  | (None, 65, 128)  | 0            | Conv1D_3   |
| Concatenate                    | (None, 66, 128),<br>(None, 65, 128),<br>(None, 65, 128) | (None, 196, 128) | 0            | MaxPooling1D_1,<br>MaxPooling1D_2,<br>MaxPooling1D_3 |
| MaxPooling1D_4                 | (None, 196, 128)  | (None, 65, 128)  | 0            | Concatenate  |
| Bidirectional_GRU              | (None, 65, 128)   | (None, 65, 200)  | 183200       | MaxPooling1D_4                                       |
| TimeDistributed (Dense)        | (None, 65, 200)   | (None, 65, 100)  | 20100        | Bidirectional_GRU                                    |
| Hierarchical_attention_Network | (None, 65, 100)   | (None, 100)      | 10200        | TimeDistributed (Dense)                              |

### C. Sentence level encoding and attention

As discussed, selecting important sentences in the document becomes another crucial task for the classification. Hence, we follow the same procedure for generating the document representation of the sentences, as discussed above in Section B, by encoding the sentences with relevant context and computing the crucial weights of these contexts for the document classification.

The sentence vector  $c_i$  obtained above is passed into the 1D CNN layer and Bi-GRU layer to encode the sentence and obtain information  $h_s$  of the neighboring sentences centered around sentence  $s$ . The sentence context vector  $v_{st}$  is used to compute the sentence level attention  $a_s$ , which further yields the document vector  $d$  that summarizes all the information of the sentences in a document. Mathematically, this is represented as below in Eq. (5.7) - (5.9).

$$v_s = \text{ReLu}(W_s \cdot h_s + b_s) \quad (5.7)$$

$$a_s = \frac{\exp(v_{st} \cdot v_s)}{\sum_s \exp(v_{st} \cdot v_s)} \quad (5.8)$$

$$d = \sum_s (a_s \cdot h_s) \quad (5.9)$$

**Table 5.2 Layer description of Step 3: Sentence-level Encoding and Attention**

| Layer Name                     | Input Shape                                    | Output Shape    | # Parameters | Connected to                                   |
|--------------------------------|--|-----------------|--------------|--|
| Input Layer                    | (None, 15, 200)                                | (None, 15, 200) | 0            | -  |
| TimeDistributed (Model)        | (None, 15, 200)                                | (None, 15, 100) | 2202684      | Input Layer                                    |
| Conv1D_1                       | (None, 15, 100)                                | (None, 13, 128) | 38528        | TimeDistributed (Model)                        |
| Conv1D_2                       | (None, 15, 100)                                | (None, 12, 128) | 51328        | TimeDistributed (Model)                        |
| Conv1D_3                       | (None, 15, 100)                                | (None, 11, 128) | 64128        | TimeDistributed (Model)                        |
| MaxPooling1D_1                 | (None, 13, 128)                                | (None, 4, 128)  | 0            | Conv1D_1                                       |
| MaxPooling1D_2                 | (None, 12, 128)                                | (None, 4, 128)  | 0            | Conv1D_2                                       |
| MaxPooling1D_3                 | (None, 11, 128)                                | (None, 3, 128)  | 0            | Conv1D_3                                       |
| Concatenate                    | (None, 4, 128), (None, 4, 128), (None, 3, 128) | (None, 11, 128) | 0            | MaxPooling1D_1, MaxPooling1D_2, MaxPooling1D_3 |
| MaxPooling1D_4                 | (None, 11, 128)                                | (None, 3, 128)  | 0            | Concatenate                                    |
| Bidirectional_GRU              | (None, 3, 128)                                 | (None, 3, 200)  | 183200       | MaxPooling1D_4                                 |
| TimeDistributed (Dense)        | (None, 3, 200)                                 | (None, 3, 100)  | 20100        | Bidirectional_GRU                              |
| Hierarchical_attention_Network | (None, 3, 100)                                 | (None, 100)     | 10200        | TimeDistributed (Dense)                        |
| Dropout                        | (None, 100)                                    | (None, 100)     | 0            | Hierarchical_attention_Network                 |
| Dense                          | (None, 100)                                    | (None, 3)       | 303          | Dropout  |

The layer details of the Sentence level encoding and attention step are shown in Table 5.2. The learned vector  $d$  of [1\* 100] dimension is passed to the dropout layer with a 0.5 rate to handle the overfitting problem in deep neural network architectures. This is followed by a dense layer and softmax activation function. The entire network is trained end to end for obtaining the best weights by monitoring the validation accuracy of the model. We use categorical cross-entropy loss, which is defined in Eq. (5.10) as follows:

$$Loss = - \sum_{j=1}^3 (y_j \cdot \log \hat{y}_j) \quad (5.10)$$

Where,  $y_j$  is the target value corresponding to the model output  $\hat{y}_j$  for the  $j^{th}$  sample. We have used the L2-regularization parameter, which penalizes the larger weights in the network for avoiding the high variance problem. This makes the objective function of the network as follows:

$$Cost\ function = Loss + (\lambda/2m * \sum ||w||^2) \quad (5.11)$$

Here,  $\lambda$  is the regularization parameter and one of the hyperparameters of the network, which is tuned according to the validation set. Our model achieves optimized results with  $\lambda = 0.001$ .

## 5.1.2 Experimental Analysis

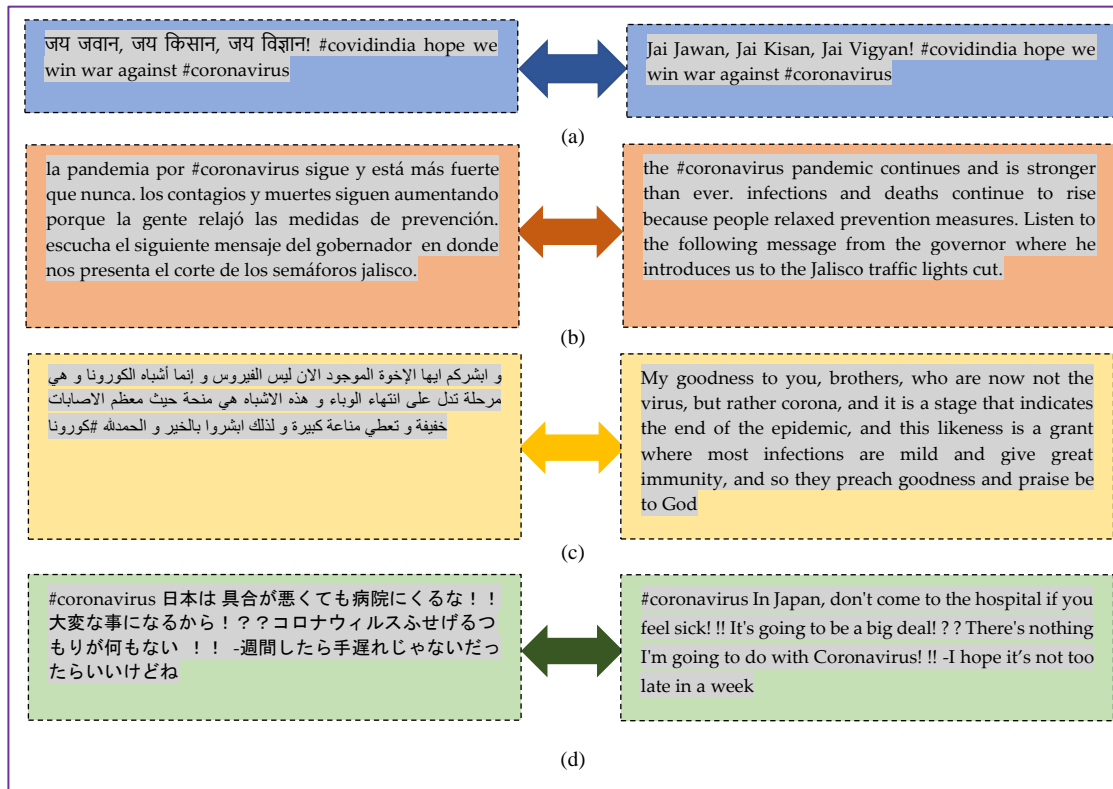
This section discusses the data collection and labeling process in Section 5.1.2.1, implementation details in Section 5.1.2.2. We evaluate the effectiveness of the network in Section 5.1.2.3 and perform baseline comparison in Section 5.1.2.4. Finally, we visualize our classification results and extract the sentiment topics in Section 5.1.2.5.

### 5.1.2.1 Dataset Collection and Labeling

In order to analyze the impact of the novel coronavirus, we extracted the Tweets from January 1, 2020 to June 7, 2020 of the top five most affected countries, namely, USA (52,03,206 cases), Brazil (32,24,876 cases), India (25,26,192 cases), Russia (9,17,884 cases), and South Africa (5,79,140 cases) as per data available from WHO<sup>7</sup>. We used some popular hashtags like #coronavirus, #covid19, and #COVID\_19. The scrapped tweets were preprocessed as discussed in Section 5.1.1.1 and translated into English by the Google translator. Some sample tweets in a different language, and their English

<sup>7</sup>[https://www.who.int/docs/default-source/coronaviruse/situation-reports/20200815-covid-19-sitrep-208.pdf?sfvrsn=9dc4e959\\_2](https://www.who.int/docs/default-source/coronaviruse/situation-reports/20200815-covid-19-sitrep-208.pdf?sfvrsn=9dc4e959_2)  
(As on 15 August, 2020)

translation is shown in Figure 5.2.



**Figure 5.2** Sample tweets in different languages from COVID-19 Sentiment Dataset (a) Hindi to English (b) Spanish to English (c) Arabic to English (d) Japanese to English.

We used the Textblob<sup>8</sup> library to weakly label the processed tweets with the sentiment score of  $[-1, 1]$ . The tweets with  $+1$  sentiment scores were labeled as positive, tweets with  $-1$  score were labeled as negative, and finally, tweets with  $0$  sentiment scores were labeled as neutral. However, we observe that in many cases, the Textblob library fails to classify the data into correct sentiment labels. E.g., Consider the following tweet, “My dad tested positive for the corona” was labeled as positive sentiment, though it belongs to the negative sentiment category. Hence, the shortlisted tweets were again manually annotated (strong labels) by three humans with positive ( $+1$ ), negative ( $-1$ ), and neutral ( $0$ ) sentiment category. Finally, 4118 tweets were annotated with an average value of Cohen Kappa inter-annotator agreement as 0.85. The country-wise dataset details corresponding to each sentiment category are shown in Table 5.3.

<sup>8</sup> <https://textblob.readthedocs.io/en/dev/>

Table 5.3 Country-wise details of COVID -19 Sentiment Dataset

| Country      | Sentiment   |             |             | Total       |
|--------------|-------------|-------------|-------------|-------------|
|              | Positive    | Negative    | Neutral     |             |
| USA          | 383         | 321         | 431         | 1135        |
| Brazil       | 267         | 318         | 444         | 1029        |
| India        | 302         | 164         | 299         | 765         |
| Russia       | 147         | 168         | 249         | 564         |
| South Africa | 200         | 193         | 232         | 625         |
| Total        | <b>1299</b> | <b>1164</b> | <b>1655</b> | <b>4118</b> |

### 5.1.2.2 Implementation Details

The proposed MACBiG-Net is implemented on Python 3 using the popular Keras framework. The experiments were performed on Windows 10 machine with 128GB RAM using NVIDIA Titan RTX GPUs. The embedding dimension is 100, maximum length of all sequences is 200, and Punkt sentence Tokenizer is used for dividing the text into a list of sentences. Adam optimizer with learning rate = 0.0001, default parameters  $\beta_1 = 0.9$  and  $\beta_2 = 0.999$  is set for training. The entire network is trained end to end with 64 batches for 100 epochs. Dropout and Regularization are used to avoid overfitting. We have used a ten-fold cross-validation testing strategy with 80% data for training, 5% data for validation, and 15% data for testing. The validation accuracy is monitored during training the network. The model achieving the highest validation accuracy is selected for the testing phase.

### 5.1.2.3 Classification Results and Observations

We compare the sentiment prediction performance of the MACBiG-Net in terms of precision, recall, F1 score, and accuracy corresponding to different sentiment categories, as shown in Figure 5.3 (a). Since we used a ten-fold cross-validation strategy, the final results are reported by averaging the results across each of the test fold.

The accuracy of the positive, negative, and neutral sentiment category is 79.9%, 80.2%, and 83.6%, respectively. The average accuracy of the network is 81.5%. Further, the confusion matrix in Figure 5.3 (b) gives an in-depth analysis of the classification results. It can be seen that a major amount of misclassification occurs when a positive sample is incorrectly classified as neutral. Yet, we can see that the



network can discriminate well across each of the sentiment categories. These results clearly explain the effectiveness of MACBiG-Net.

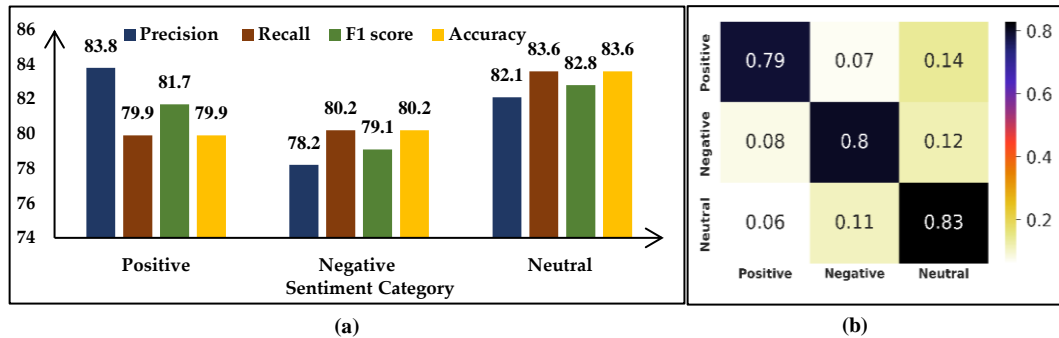


Figure 5.3 (a) Sentiment classification results on COVID-19 Sentiment Dataset (%) (b) Confusion Matrix.

We also provide accuracy and loss curves for training and validation set in Figure 5.4 (a) and (b), respectively. As we can see, the training accuracy is 100%, which means the model is trained completely, and the validation accuracy shows how effectively the model can perform on the unseen samples. Similarly, decreasing loss curves with each epoch confirm the adequate learning of the model and validate its internal classification performance.

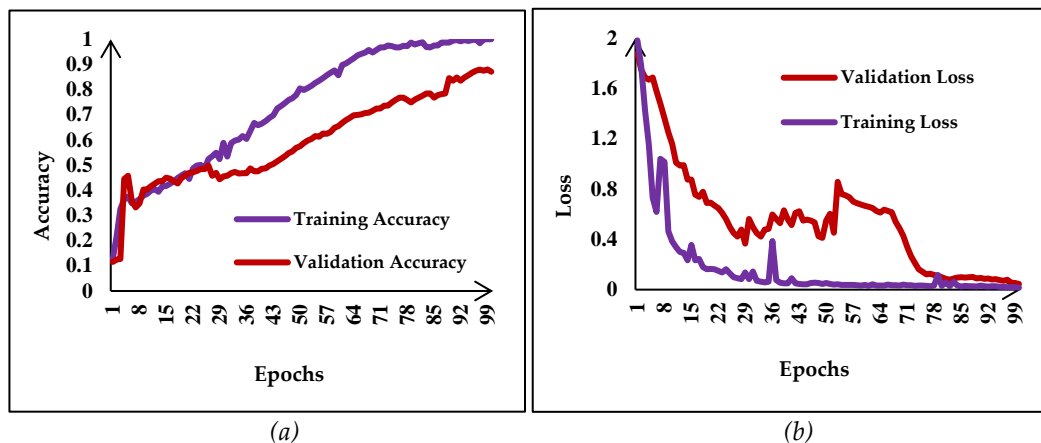
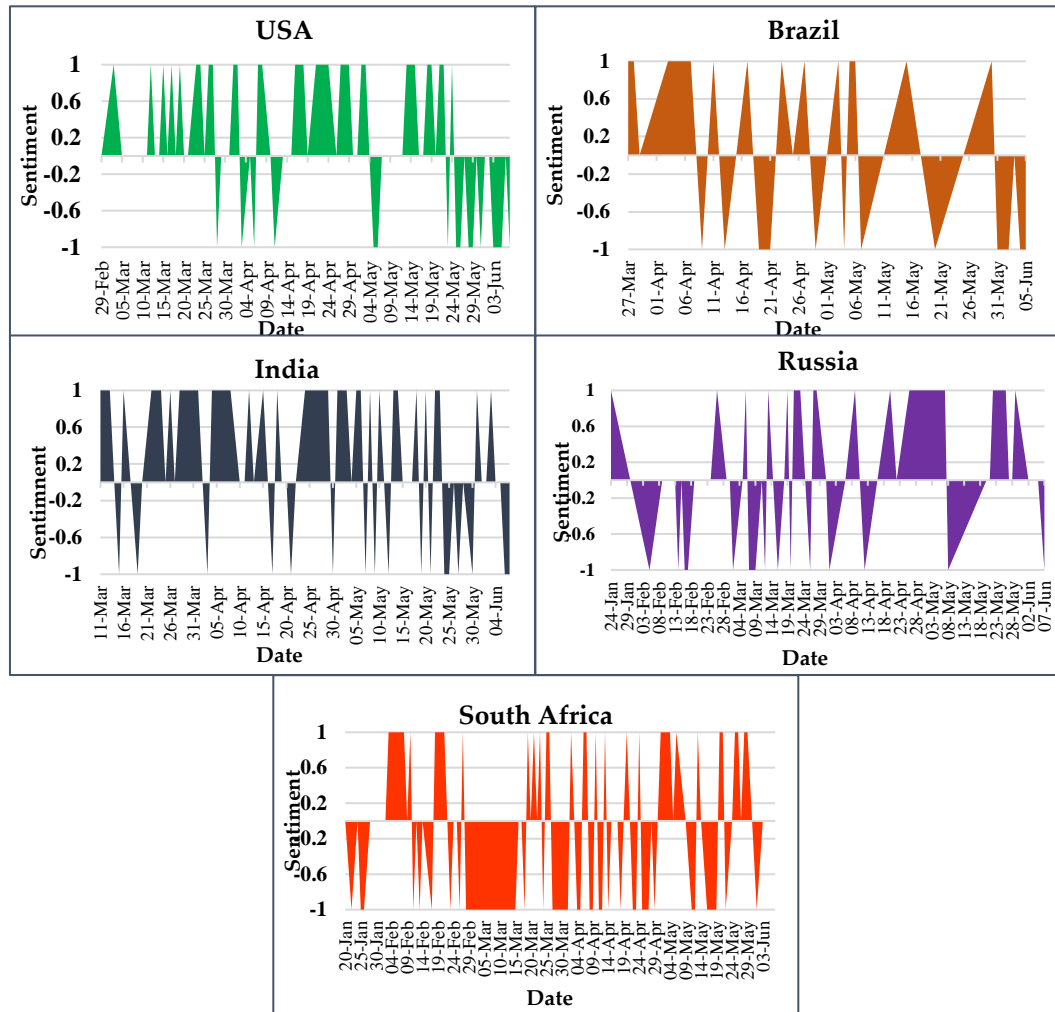


Figure 5.4 Tracing internal performance of MACBiG-Net with (a) Accuracy curve for train and validation set (b) Loss curves for train and validation set.

The foremost aim of our work is to analyze the sentiments of people expressed globally on the COVID-19 pandemic. This can be visualized in Figure 5.5, where we plot the positive (+1), negative (-1), and neutral (0) sentiments of people over time located in the top five worst affected countries by the pandemic. As discussed, we retrieved the tweets from January 1, 2020 to June 7, 2020. However, in many countries,



**Figure 5.5 Visualizing the sentiments of people over time in top five worst affected countries by COVID-19 pandemic.**

sentiments were generated in the month of late February or March. This may be because the first confirmed case in that country started in late February (Eg: *The virus was first confirmed in Brazil on February 25, 2020*<sup>9</sup>), or no specific sentiments were generated during January and February. A five-day window represents the time component in Figure 5.5 along the x-axis. This means that each bar represents the sentiment expressed in these five days. However, each day, several sentiments were being generated. Hence, we took the sentiment that was voiced the maximum number of times by the people on any particular day as the final sentiment for that day. We summarize our observations as follows:

<sup>9</sup> <https://www.gov.br/saude/pt-br>

- ✓ In the USA, significant sentiments emerged from March onwards. This may be because Trump declared a national emergency<sup>10</sup> on March 13, and by the end of March, the cases were reported in all 50 US States<sup>11</sup>. In early March, positive sentiment was recorded. However, by the end of March till mid-April, negative sentiment was prevalent. Similarly, by the end of May, we observed mostly negative sentiments. One of the primary reasons was the breaking news of George Floyd, which created a state of panic and terror in the USA from May 25 onwards. People started tweeting that COVID-19 will continue to spread as citizens were protesting and moving in groups without following any preventive measures. Overall, it was observed that 43.6% of sentiments were neutral, 37.9% were positive, and 18.3% were negative.
- ✓ Brazil has shown positive sentiments in the early days of April. However, mostly negative sentiments were conveyed in May and June. This may be because the number of active cases and deaths was doubled in May. Further, this number only kept on increasing, creating a condition of fear among people. Overall, 45.4% of negative sentiment, 39.3% of positive sentiment, and 15.1% of neutral sentiments were conveyed by the citizens of Brazil.
- ✓ In India, positive sentiments were observed, majorly in March and April. This may be because people were hopeful in the initial stages to overcome the virus. Further, Prime Minister (PM) Narendra Singh Modi came up with some ideas to boost the support of all the frontline workers. On March 22, the PM requested every Indian to clap their hands or ring some bells from their house to boost the morale of frontline workers. Similarly, on April 5, the PM urged everyone to light candles or *diyas* for nine minutes as a mark to fight against the deadly virus. Several people followed these tasks and happily enjoyed them. Towards the end of May, we see the people expressed negative sentiments. Since, by the end of May, a large number of active cases were being reported each day. Overall, 50.6% of positive sentiments, 28.3% of neutral sentiments, and 20.9% of negative sentiments were

---

<sup>10</sup><https://edition.cnn.com/2020/03/13/politics/donald-trump-emergency/index.html>

<sup>11</sup>[https://archive.vn/20200428073138/https://banningca.gov/DocumentCenter/View/7139/CDC\\_COVID19-Weekly-Key\\_Messages\\_03292020\\_FINAL](https://archive.vn/20200428073138/https://banningca.gov/DocumentCenter/View/7139/CDC_COVID19-Weekly-Key_Messages_03292020_FINAL)

observed in India.

- ✓ Russia reported its first positive case on January 31, 2020. People have majorly shown neutral and negative sentiments during the mid of February. However, positive sentiments were expressed in April. Overall, 39.2% of neutral sentiment, 33.9% of positive sentiment, and 26.7% of negative sentiment were observed in Russia.
- ✓ In South Africa, majorly negative sentiments were being observed from late February till mid of March. People finally realized that the deadly virus had hit their country as the number of cases started rising sharply. However, in April, we can see some positive sentiments as citizens started motivating everyone by asking them to wear masks, read books, involve in daily exercise, and posting positive ideas by their former President Nelson Mandela. Overall, South Africa expressed 39.6% of negative sentiments, 32.6% of positive sentiments, and 27.7% of neutral sentiments.
- ✓ Finally, we conclude that overall, 38.3% of positive sentiments, 32.6% of neutral sentiment, and 28.3% of negative sentiments were expressed by these top five affected countries.

#### 5.1.2.4 Baseline Comparison

We use several popular baseline methods for comparison with the proposed MACBiG-Net. The details are discussed as follows:

- ✓ **LSTM** [183]: The classical long-short memory network is applied. Word vectors are initialized by Glove. The hidden layer size is 300 units. Adam optimizer with a learning rate of 0.001 is used.
- ✓ **Bi-LSTM** [191]: This represents standard bi-directional LSTM. We follow the same settings as described above for LSTM.
- ✓ **CNN** [81]: Convolutional neural network with multiple filters of size [3,4,5], each having 100 filters, is applied. Adam optimizer is used with a learning rate of 0.001.
- ✓ **CNN-RNN** [217]: We successively stack the CNN layer with a max-pooling layer followed by RNN units. For RNN, we experimented with LSTM and GRU variants. The GRU based integration shows the highest performance.

- ✓ **HAN** [39]: The hierarchical attention network uses two LSTM layers followed by an attention mechanism at word-level and sentence-level for composing the final text representation. The HAN is the simplest form of our proposed approach as it does not encounter localized features.
- ✓ **Self-attention based LSTM** [218]: The self-attention mechanism is used with Bi-LSTM, which learns the contribution of each hidden state by providing a set of summarized weight vectors for each hidden state. A learning rate of 0.06 is used.
- ✓ **RNN-Capsule** [219]: We apply capsule networks where the input instance representation is taken from the LSTM unit. The dimension of hidden vectors is set to 256. The attention mechanism is used to construct the capsule representation inside another capsule. All word vectors are initialized with Glove. Adam optimizer is used with a learning rate of 0.001.
- ✓ **BERT** [220]: We use a pre-trained Bidirectional Encoder Representation from Transformers (BERT) model with the help of an online released TensorFlow<sup>12</sup> library. The transformer blocks are set to 12, hidden size to 768, and self-attention head to 12.

As evident from Table 5.4, the proposed network has outperformed the baseline methods and recent works in terms of sentiment classification by achieving 81.5% average accuracy. However, the HAN [39] and CNN-RNN [217] provides good enough results for the classification. This serves as the primary motivation of our proposed MACBiG-Net, which inherits the basic idea of both the methods and achieves nearly 2% and 4% higher accuracy compared to [39] and [217], respectively.

| Method                          | Accuracy | Precision | Recall | F1 score |
|---------------------------------|----------|-----------|--------|----------|
| LSTM [183]                      | 72.6     | 73.3      | 73.6   | 73.4     |
| Bi-LSTM [191]                   | 73.2     | 74.9      | 72.0   | 73.4     |
| CNN [81]                        | 75.8     | 73.1      | 75.2   | 74.1     |
| CNN-RNN [217]                   | 77.7     | 78.6      | 74.2   | 76.3     |
| HAN [39]                        | 79.4     | 77.4      | 78.9   | 78.1     |
| Self-attention-based LSTM [218] | 70.9     | 68.2      | 66.7   | 67.4     |
| RNN-Capsule [219]               | 71.5     | 68.5      | 70.8   | 69.6     |
| BERT [220]                      | 76.6     | 75.2      | 74.8   | 74.9     |

<sup>12</sup> <https://github.com/tensorflow/tensor2tensor>

|                   |      |      |      |      |
|-------------------|------|------|------|------|
| MACBiG-Net (Ours) | 81.5 | 81.3 | 81.2 | 81.2 |
|-------------------|------|------|------|------|

Table 5.4 Comparative Results of different methods on COVID-19 Sentiment Dataset (%)

This can also be visualized from the ROC curves and area under the curve (AUC), shown in Figure 5.6, where we plot the ROC curves for the baseline methods. The AUC values help in comparing the ROC curves in a better way. From the AUC values, it is clear that MACBiG-Net has consistently outperformed all the previous methods for sentiment classification.

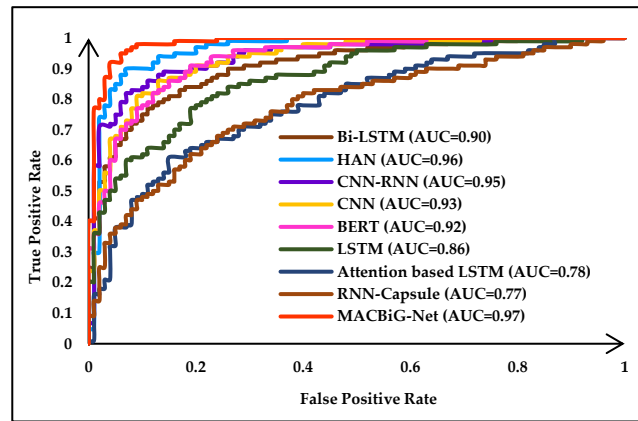


Figure 5.6 Comparison of MACBiG-Net with baseline methods in terms of ROC curve and Area under the curve (AUC).

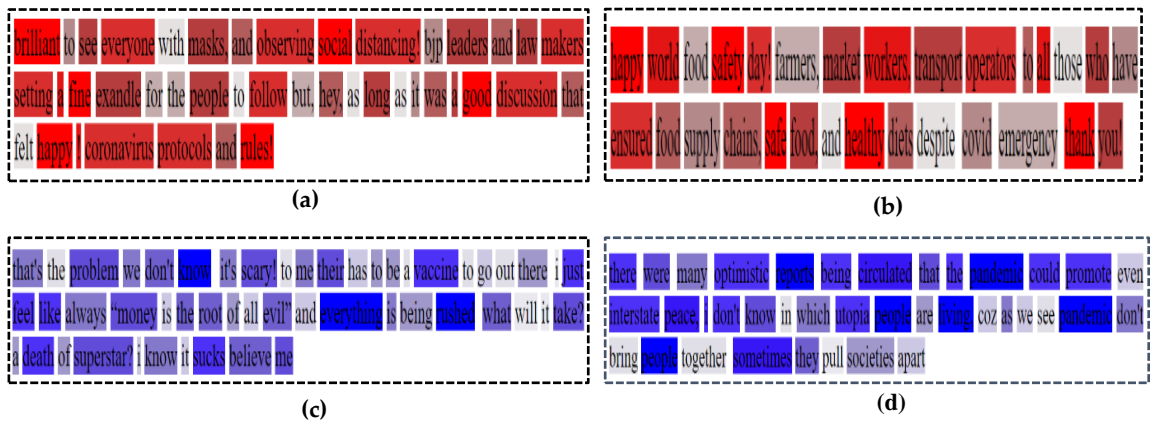
### 5.1.2.5 Qualitative Analysis

In this section, qualitative analysis is performed to visualize the word-level attention and extract the positive and negative sentiment topics that were generated during the pandemic.

#### A. Visualizing Word-level Attention

In this section, qualitative analysis is performed on MACBiG-Net, to investigate the role of word-level attention for sentiment classification. We randomly picked four tweets from our dataset, two of them were positive, and two were negative. Thus, in Figure 5.7, (a) – (b) belongs to the positive sentiment class (shown in Red color), and (c) – (d) belongs to the negative class (shown in Blue color). The color signifies the attention weights of each word. The darker the color in the respective category, the higher amount of attention is given to the word.

We can see that attention focuses on the prominent part of the sentence, by giving darker color to the most important word. E.g., In review sentences (a) - (b), positive sentiment is conveyed by the crucial words like 'brilliant', 'happy', 'healthy', 'safety', 'good', which have received more score by the attention mechanism for better sentiment prediction. Similarly, in reviews (c) - (d), negative sentiment is reflected by words like 'scary', 'pandemic', 'death', which again have received more attention score. The words which convey no informative meaning (like stop words), were given very low weights by highlighting them with light colors. This shows that the attention weights focus on those words which have more relation with the output sentiment category. Thus, the visualization shows how the attention weights are changing along with the words in a sentence.

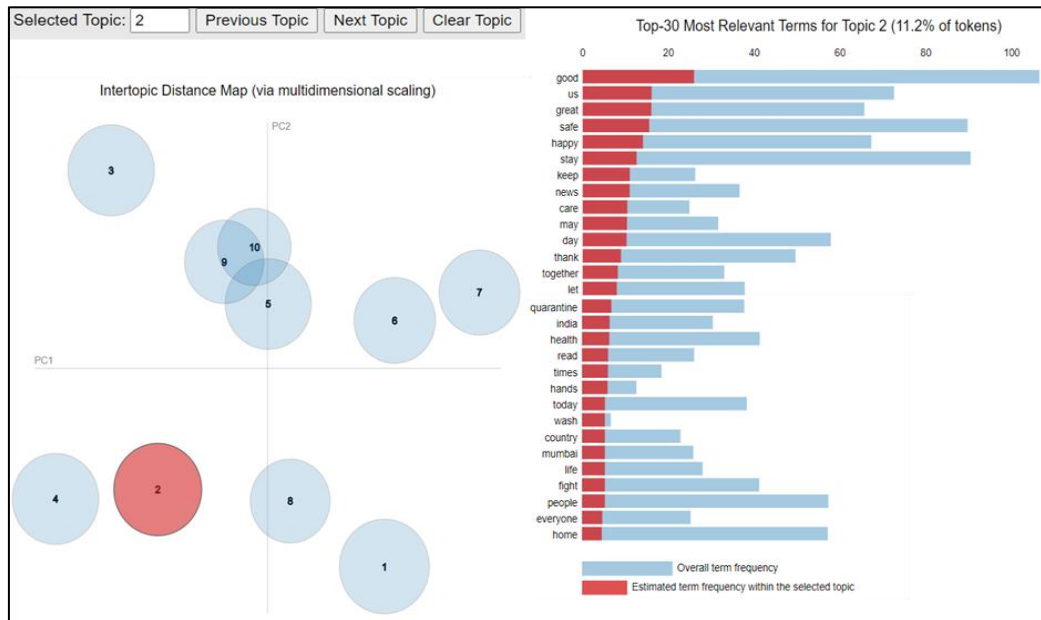


**Figure 5.7 Visualization the word-level attention weights from proposed dataset: (a)-(b) Positive Sentiment (c)-(d) Negative Sentiment.**

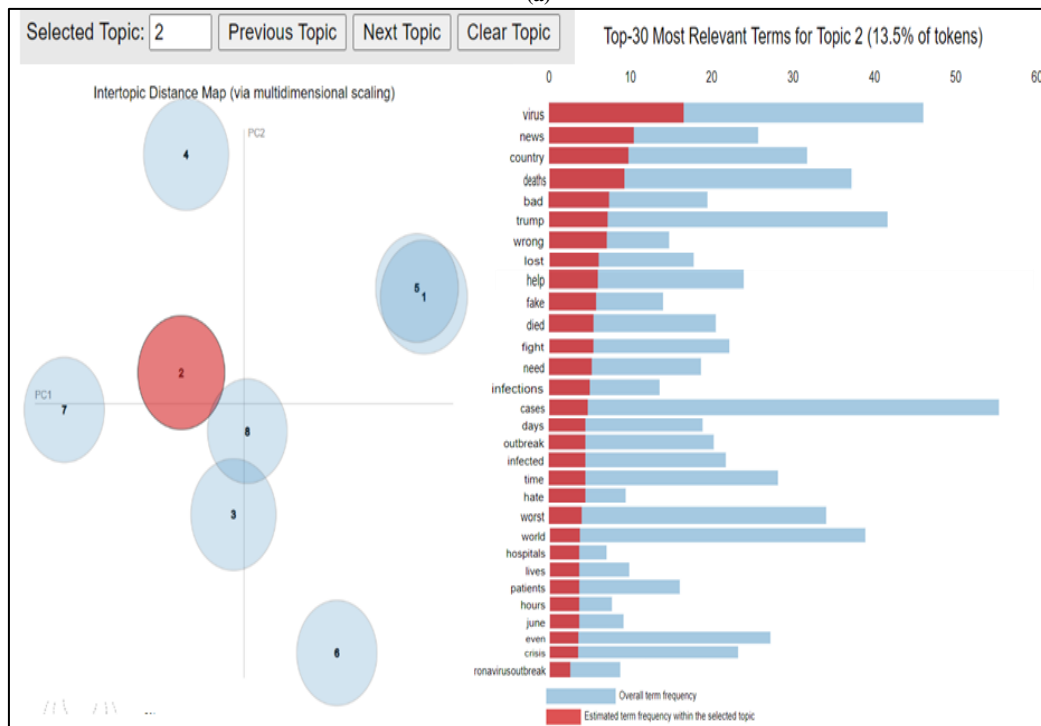
## B. Visualizing positive and negative sentiment topics

To get an in-depth analysis on the opinion of the people, we apply LDA (Latent Dirichlet Allocation) based topic modeling to determine the set of words (topics) that contribute to generate the positive and negative sentiments. This will help in finding the hidden thematic structure of the document and drawing essential conclusions from them. We use LDAvis [221] tool for visualization, which further helps in finding the prevalent topics in a document, relation between the topics, and the conditional distribution of a given term over the set of topics. The optimal number of topics was determined by building many LDA models with a different number of topics ( $k$ ) and selecting the one with the highest coherence score [222]. This gives us  $k = 10$ . The

layout of LDAvis is shown in Figure 5.8 (a)-(b) for positive and negative sentiment labels, respectively.



(a)



(b)

Figure 5.8 Identifying the hidden information in document through LDAvis visualization for (a) Positive and (b) Negative sentiment

If we consider Figure 5.8 (a), the entire layout is divided into two panels: the left panel shows the ten topics in the form of circles where the similar topics are plotted



close to each other, the right panel shows the bar graphs which represent the terms or words that are most useful for understanding the currently selected topic (from the left panel). The length of the gray bars depicts the entire document-wise frequencies of each term, and the red bars depict the topic-wise frequencies of each term. Thus, for a given topic, we can compare the length of the red and gray bars to understand the relevance of the topic. Further, selecting a term on the right panel tells the conditional distribution of the selected term over the topics (on the left panel). In this way, topic modeling helps in explaining the underlying information in a document. Hence, we summarize our observations as follows:

- ✓ The prevalent positive words related to COVID-19 on Twitter are: ‘good’, ‘stay safe’, ‘happy’, ‘best’, ‘thank you’, ‘support’, ‘doctors’, ‘mask’, ‘together’, ‘proud’ and the popular negative words linked to COVID-19 on twitter are: ‘virus’, ‘deaths’, ‘bad’, ‘trump’, ‘help’, ‘lost’, ‘fake’, ‘died’, ‘outbreak’, ‘infected’, ‘hate’, ‘worst’, ‘crisis’, ‘fight’.
- ✓ Many positive tweets were directed towards the frontline workers where users were motivating and boosting the morale of the health care workers by appreciating them, donating them masks, dedicating some songs, providing free accommodation to them, and donating money to restaurants for distributing meals to essential workers.
- ✓ In some countries like Brazil, health care workers celebrate the discharge of the first COVID-19 patient from the hospital. In India, many citizens were delighted to see everyone unified with candles or flashlights in their hands from their balconies.
- ✓ Finally, positive sentiments were directed towards maintaining a healthy diet, reuniting with families after travel restrictions were lifted, boosting the immune system, asking everyone to stay safe, binge-watching television series, encouraging everyone to take precautions, wishing and motivating the users on special occasions by posting positive messages on occasions like Birthdays, Mothers Day, Earth Day, World Bicycle Day, Memorial Day (celebrated in the US), World Labor Day, International Nurses Day, and staying together to beat the deadly virus.
- ✓ During the pandemic, several social-economic factors also contributed to generate

negative sentiments in people all around the world. In the US, the negative tweets were mainly focused on the racial injustice related to the death of George Floyd, which resulted in massive protests and violence even during the time of the pandemic. People fear that these mass demonstrations may occur in an uptick in the COVID-19 cases.

- ✓ In India, Mumbai (one of the worst-hit cities by COVID-19) was severely affected by cyclone winds. In West Bengal, Cyclone Amphan caused extensive damage. The poor, jobless migrants went traveling back to their homes by walking thousands of kilometers due to lockdown restrictions in India.
- ✓ In general, many people also lost their jobs. The unemployment rate was increased during the pandemic. People were bored and anxious during home-quarantine. The upsurge in infected cases also created fear and stress among people. Further, lots of fake news<sup>13</sup> was generated on social media, creating panic about the disease.
- ✓ The popular positive hashtags are: #salutetocoronafighters, #inthistogether, #lovemycountry, #withme, #thankyouindia, #indiafightscorona, #flattenthecurve, #fightthevirus, and #frontlineheroes and negative hashtags are: #DictatorTrump, #BlackLivesMatter, #GeorgeFloyd, #AmericaInCrisis, #Riot2020, #migrantlivesmatter, #pandemia, #chinavirus, #wuhavirus, and #fuck\_coronavirus.

## 5.2 Significant Outcomes

In this chapter, a language-independent framework is developed to analyze the sentiments of people generated during the COVID-19 pandemic by modeling the hierarchical structure of the document. The observations are summarized as follows:

- The MACBiG-Net captures the structure of words from the sentence and the structure of sentences from a document. The word-level and sentence-level encoding extract the spatial and temporal features of the input sequence, and the attention mechanism assigns more weights to relevant contexts for the

---

<sup>13</sup> <https://timesofindia.indiatimes.com/blogs/toi-editorials/social-media-menace-beware-of-fake-news-going-viral-faster-than-covid-19-it-will-cost-lives-literally/>

document classification.

- The experimental results demonstrate that the proposed network can classify the tweets belonging to multiple languages like Hindi, Japanese, Arabic, Spanish, Urdu, etc. effectively.
- We observed that during the COVID-19 pandemic, lots of sentiments were being generated, and people were posting their views or opinions on several topics like frontline workers, upsurge in active cases, travel restrictions, and about the virus itself. Much fake news was also generated during this time, which was misleading the people. However, despite being stressed, people appreciated the efforts of all frontline workers and motivated each other to follow all precautions.

*This chapter is based on the following works:*

- ✓ **Ashima Yadav**, Dinesh Kumar Vishwakarma. “A Language-independent Network to analyze the impact of COVID-19 on the World via Sentiment Analysis.”, *ACM Transactions on Internet Technology (Major Revision-1)* / (*arXiv:2011.10358*)

# Chapter-6 Conclusion and Future Work

---

## 6.1 Conclusion

We developed four approaches that deal with the practical problems of SA faced on different modalities, namely: social media images, movie-trailers (videos), Twitter text reviews, and multimodal data (images and text on social media). These approaches are summarized as follows:

- ✓ Firstly, we develop an effective residual attention-based deep learning network (RA-DLNet) for visual sentiment classification, which applies CNN architecture to capture the spatial features and residual attention mechanism to extract the local sentiment-rich features from the images posted by the users on different social media platforms. The results demonstrate that the RA-DLNet architecture overpasses state-of-the-art results on the six publicly available, challenging real-world datasets for binary sentiment classification and two benchmark datasets for multi-class emotion classification. Furthermore, we have provided a comparative analysis of the NASNet architecture used in our model with the six popular CNN architectures to show the efficacy of the RA-DLNet architecture.
- ✓ In the second approach, we propose an idea in the field of affect-based video classification by designing a novel deep affect-based movie genre classification framework, which utilizes the emotions evoked while watching a movie trailer to classify the movie trailer into different genres. For this, we also developed an EmoGDB dataset, which contains 100 Bollywood movie trailers annotated with popular movie genres: Action, Comedy, Drama, Horror, Romance, Thriller, and six different types of induced emotions: Anger, Fear, Happy, Neutral, Sad, Surprise. The prime advantage of our work is that without watching the entire movie trailer, the architecture can classify the trailer into multiple movie genres.

- ✓ Next, we propose a Deep Multi-level Attentive network (DMLANet) for automatically detecting the sentiments from multimodal data by exploiting the correlation between image and text modalities for improving the multimodal learning process. The visual attention block applies channel and spatial attention to generate robust bi-attentive visual features. Moreover, the joint-attended multimodal learning aims to acquire high-quality representations from text and image modalities by focusing on the words that are related to the image contents. Experimental results on four real-world datasets show promising results, as validated by Precision, Recall, F1 score, accuracy, ROC, and PRC metrics. Hence, the proposed model achieves the best performance as compared to other multimodal based approaches.
- ✓ Finally, we explore the application of SA by analyzing the attitude and behavior of the people related to the COVID-19 pandemic from the top five worst affected countries by the virus, namely the USA, Brazil, India, Russia, and South Africa starting from January 1, 2020, to June 7, 2020. This is achieved by developing and labeling the COVID-19 Sentiment Dataset, which contains 4118 labeled tweets crawled from Twitter. Further, we visualize the sentiments of the people from January 31 to June 7 and summarize essential observations starting from January 1, 2020, to June 7, 2020. This analysis can serve as feedback to the government agencies regarding the mitigation plans taken by them. Further, it may also guide the future planning of the public health agencies in case of any such outbreak.

## **6.2 Future Work**

- ✓ For text-based sentiment classification, most of the existing work on SA has focused on explicit sentiment detection in which the aspect term appears as a noun phrase in the sentence. However, very less work has been done for extracting the implicit aspects. Implicit aspects are not specified explicitly in the text, which makes implicit sentiment detection a challenging task. The combination of explicit and implicit sentiments will increase the overall sentiment classification process as implicit aspects convey opinions for

understanding the customer reviews.

- ✓ For multimodal SA, we have focused on data samples having some fine-grained correlation between image-text pairs. However, this is not true in reality. Hence, a robust fusion method can be developed that could work well on datasets that do not have a close cross-modal correlation in the future.
- ✓ Social media is an open platform for everyone to express themselves. However, it is often misused by a certain group of people, who spread offensive or hate comments targeting other individuals. These comments must be detected on time to maintain harmony as they could negatively affect the target individual, leading to suicide in extreme cases. Hence, SA can be used as a tool to identify hate speech, racist comments, or trolls on social media.
- ✓ Similarly, social media has become one of the prominent sources for spreading news and information to users. However, recent years have seen that it has become a medium for spreading fake news and rumors. Users post videos, photos, news, which is seldom unverified and goes “viral” in a short period. It is a challenging task to detect such fabricated multimedia on the Internet. Hence, sentiment-based features can be taken into account to identify the false news in multimodal data.
- ✓ In multimodal sentiment classification, multimodal fusion considers the individual contribution of each modality. However, this is not sufficient. The literature lacks proper datasets having rich fine-grained annotation for each modality (text, audio, and video), which can better guide the multimodal fusion methods.

---

## References

---

- [1] M. Dragoni and G. Petrucci, “A neural word embeddings approach for multi-domain sentiment analysis,” *IEEE Transactions on Affective Computing*, vol. 8, no. 4, pp. 457-470, 2017.
- [2] D. I. H. Farías, V. Patti and P. Rosso, “Irony Detection in Twitter: The Role of Affective Content,” *ACM Transactions on Internet Technology*, vol. 16, no. 3, pp. 1-24, 2016.
- [3] “Facebook: number of monthly active users worldwide 2008-2020,” Statista, 2020. [Online]. Available: <https://www.statista.com/statistics/264810/number-of-monthly-active-facebook-users-worldwide/>.
- [4] D. She, J. Yang, M.-M. Cheng, Y.-K. Lai, P. L. Rosin and L. Wang, “WSCNet: Weakly Supervised Coupled Networks for Visual Sentiment Classification and Detection,” *IEEE Transactions on Multimedia*, vol. 22, no. 5, pp. 1358-1371, 2019.
- [5] S. Wang, A. Maolinyazi, X. Wu and X. Meng, “Emo2Vec: Learning Emotional Embeddings via Multi-Emotion Category,” *ACM Transactions on Internet Technology (TOIT)*, vol. 20, no. 2, pp. 1-17, 2020.
- [6] L. Wang, J. Niu and S. Yu, “SentiDiff: Combining textual information and sentiment diffusion patterns for Twitter sentiment analysis,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 32, no. 10, pp. 2026 - 2039, 2019.
- [7] R. Ji, F. Chen, L. Cao and Y. Gao, “Cross-Modality Microblog Sentiment Prediction via Bi-Layer Multimodal Hypergraph Learning,” *IEEE Transactions on Multimedia*, vol. 21, no. 4, pp. 1062-1075, 2018.
- [8] K. Ravi and V. Ravi, “A survey on opinion mining and sentiment analysis: tasks, approaches and applications,” *Knowledge-Based Systems*, vol. 89, pp. 14-46, 2015.
- [9] K. Denecke and Y. Deng, “Sentiment analysis in medical settings: New opportunities and challenges,” *Artificial Intelligence in Medicine*, vol. 64, pp. 17-27, 2015.
- [10] Q. Sun, J. Niu, Z. Yao and H. Yan, “Exploring eWOM in online customer reviews: Sentiment analysis at a fine-grained level,” *Engineering Applications of Artificial Intelligence*, vol. 81, pp. 68-78, 2019.
- [11] X. Li, C. Wu and F. Mai, “The effect of online reviews on product sales: A joint sentiment-topic analysis,” *Information & Management*, vol. 56, no. 2, pp. 172-184, 2019.
- [12] C. Clavel and Z. Callejas, “Sentiment analysis: from opinion mining to human-agent interaction,” *IEEE Transactions on affective computing*, vol. 7, no. 1, pp. 74-93, 2016.
- [13] D. Liu and L. Lei, “The appeal to political sentiment: An analysis of Donald Trump’s and Hillary Clinton’s speech themes and discourse strategies in the 2016 US presidential election,” *Discourse, Context & Media*, vol. 25, pp. 143-

- 152, 2018.
- [14] E. Kusen and M. Strembeck, "Politics, sentiments, and misinformation: An analysis of the Twitter discussion on the 2016 Austrian Presidential Elections," *Online Social Networks and Media*, vol. 5, pp. 37-50, 2018.
- [15] X. Li, P. Wu and W. Wang, "Incorporating stock prices and news sentiments for stock market prediction: A case of Hong Kong," *Information Processing & Management*, 2020.
- [16] A. Picasso, S. Merello, Y. Ma, L. Oneto and E. Cambria, "Technical analysis and sentiment embeddings for market trend prediction," *Expert Systems with Applications*, vol. 135, pp. 60-70, 2019.
- [17] X. Chen, Y. Cho and S. Y. Jang, "Crime prediction using twitter sentiment and weather," in *IEEE Systems and Information Engineering Design Symposium*, Virginia, 2015.
- [18] P. Singh, R. S. Sawhney and K. S. Kahlon, "Sentiment analysis of demonetization of 500 & 1000 rupee banknotes by Indian government," *ICT Express*, vol. 4, no. 3, pp. 124-129, 2018.
- [19] J. R. Ragini, P. R. Anand and V. Bhaskar, "Big data analytics for disaster response and recovery through sentiment," *International Journal of Information Management*, vol. 42, pp. 13-24, 2018.
- [20] G. A. Ruz, P. A. Henríquez and A. Mascareño, "Sentiment analysis of Twitter data during critical events through Bayesian networks classifiers," *Future Generation Computer Systems*, vol. 106, pp. 92-104, 2019.
- [21] S. Chen, J. Mao, G. Li, C. Ma and Y. Cao, "Uncovering sentiment and retweet patterns of disaster-related tweets from a spatiotemporal perspective – A case study of Hurricane Harvey," *Telematics and Informatics*, vol. 47, p. 101326, 2020.
- [22] L. Pang, S. Zhu and C.-W. Ngo, "Deep multimodal learning for affective analysis and retrieval," *IEEE Transactions on Multimedia*, vol. 17, no. 11, pp. 2008-2020, 2015.
- [23] J. Jia, S. Wu, X. Wang, P. Hu, L. Cai and J. Tang, "Can we understand van gogh's mood?: learning to infer affects from images in social networks," in *Proceedings of the 20th ACM international conference on Multimedia*, Japan , 2012.
- [24] J. Islam and Y. Zhang, "Visual Sentiment Analysis for Social Images Using Transfer Learning Approach," in *IEEE International Conferences on Big Data and Cloud Computing (BDCloud), Social Computing and Networking (SocialCom), Sustainable Computing and Communications (SustainCom)*, Atlanta, 2016.
- [25] E. Cambria, "Affective Computing and Sentiment Analysis," *IEEE Intelligent Systems*, vol. 31, no. 2, pp. 102-107, 2016.
- [26] E. Cambria, A. Livingstone and A. Hussain, "The hourglass of emotions," *Cognitive behavioural systems*, pp. 144-157, 2012.
- [27] D. Cao, R. Ji, D. Lin and S. Li, "Visual sentiment topic model based microblog image sentiment analysis," *Multimedia Tools and Applications*, vol. 75, no. 15,



- pp. 8955-8968, 2016.
- [28] T. Chen, F. X. Yu, J. Chen, Y. Cui, Y.-Y. Chen and S.-F. Chang, “Object-based visual sentiment concept analysis and application,” in *Proceedings of the 2014 ACM International Conference on Multimedia*, United States.
- [29] J. Yang, D. She, M. Sun, M.-M. Cheng, P. L. Rosin and L. Wang, “Visual Sentiment Prediction based on Automatic Discovery of Affective Regions,” *IEEE Transactions on Multimedia*, vol. 20, no. 9, pp. 2513-2525, 2018.
- [30] Z. Li, Y. Fan, B. Jiang, T. Lei and W. Liu, “A survey on sentiment analysis and opinion mining for social multimedia,” *Multimedia Tools and Applications*, vol. 78, p. 6939–6967, 2019.
- [31] Y.-Y. Chen, T. Chen, T. Liu, H.-Y. M. Liao and S.-F. Chang, “Assistive Image Comment Robot—A Novel Mid-Level Concept-Based Representation,” *IEEE Transactions on Affective Computing*, vol. 6, no. 3, pp. 298-311, 2015.
- [32] K. Dave, S. Lawrence and D. M. Pennock, “Mining the Peanut Gallery: Opinion Extraction and Semantic Classification of Product Reviews,” *Proceedings of the 12th international conference on World Wide Web*, pp. 519-528, 2003.
- [33] T. Nasukawa and J. Yi, “Sentiment Analysis: Capturing Favorability Using Natural Language Processing,” *Proceedings of the 2nd international conference on Knowledge capture*, pp. 70-77, 2003.
- [34] F. Hemmatian and M. K. Sohrabi, “A survey on classification techniques for opinion mining and sentiment analysis,” *Artificial Intelligence Review*, vol. 52, pp. 1-51, 2017.
- [35] H. Kang, S. J. Yoo and D. Han, “Senti-lexicon and improved Naïve Bayes algorithms for sentiment analysis of restaurant reviews,” *Expert Systems with Applications*, vol. 39, no. 5, pp. 6000-6010, 2012.
- [36] P. Melville, W. Gryc and R. D. Lawrence, “Sentiment Analysis of Blogs by Combining Lexical Knowledge with Text Classification,” *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 1275-1284, 2009.
- [37] H. Keshavarz and M. S. Abadeh, “ALGA: Adaptive lexicon learning using genetic algorithm for sentiment analysis of microblogs,” *Knowledge-Based Systems*, vol. 122, pp. 1-16, 2017.
- [38] M. Soleymani, D. Garcia, B. Jou, B. Schuller, S.-F. Chang and M. Pantic, “A survey of multimodal sentiment analysis,” *Image and Vision Computing*, vol. 65, pp. 3-14, 2017.
- [39] Z. Yang, D. Yang, C. Dyer, X. He, A. Smola and E. Hovy, “Hierarchical attention networks for document classification,” *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies, San Diego*, pp. 1480-1489, 2016.
- [40] G. Rao, W. Huang, Z. Feng and Q. Cong, “LSTM with sentence representations for document-level sentiment,” *Neurocomputing*, vol. 308, pp. 49-57, 2018.
- [41] R. Ghosh, K. Ravi and V. Ravi, “A novel deep learning architecture for sentiment classification,” *3rd International Conference on Recent Advances in Information Technology (RAIT)*, pp. 511-516, 2016.

- 
- [42] W. Zhao, Z. Guan, L. Chen, X. He, D. Ca, B. Wang and Q. Wang, "Weakly-supervised deep embedding for product review sentiment analysis," *IEEE Transactions on Knowledge and Data Engineering*, vol. 30, no. 1, pp. 185-197, 2017.
- [43] S. Xiong, H. Lv, W. Zhao and D. Ji, "Towards Twitter sentiment classification by multi-level sentiment-enriched word embeddings," *Neurocomputing*, vol. 275, pp. 2459-2466, 2018.
- [44] H. Peng, Y. Ma, Y. Li and E. Cambria, "Learning multi-grained aspect target sequence for Chinese sentiment analysis," *Knowledge-Based Systems*, vol. 148, pp. 167-176, 2018.
- [45] Y. Wang, M. Huang, L. Zhao and X. Zhu, "Attention-based LSTM for aspect-level sentiment classification," *Proceedings of the 2016 conference on empirical methods in natural language processing*, pp. 606-615, 2016.
- [46] J. Yu, J. Jiang and R. Xia, "Global Inference for Aspect and Opinion Terms Co-Extraction Based on Multi-Task Neural Networks," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 1, pp. 168-177, 2018.
- [47] M. Al-Smadi, . B. Talafha, . M. Al-Ayyoub and . Y. Jararweh, "Using long short-term memory deep neural networks for aspect-based sentiment analysis of Arabic reviews," *International Journal of Machine Learning and Cybernetics*, vol. 10, no. 8, pp. 2163-2175, 2019.
- [48] Y. Ma, H. Peng, E. Cambria, T. Khan and A. Hussain, "Sentic LSTM: A Hybrid Network for Targeted Aspect-Based Sentiment Analysis," *Cognitive Computation*, vol. 10, no. 4, pp. 639-650, 2018.
- [49] X. Ma, J. Zeng, L. Peng, G. Fortino and Y. Zhang, "Modeling multi-aspects within one opinionated sentence simultaneously for aspect-level sentiment analysis," *Future Generation Computer Systems*, vol. 93, pp. 304-311, 2019.
- [50] C. Yang, H. Zhang, B. Jiang and K. Li, "Aspect-based sentiment analysis with alternating coattention networks," *Information Processing & Management*, vol. 56, no. 3, pp. 463-478, 2019.
- [51] Z. Yuan, S. Wu, F. Wu, J. Liu and Y. Huang, "Domain attention model for multi-domain sentiment classification," *Knowledge-Based Systems*, vol. 155, pp. 1-10, 2018.
- [52] F. Chen, R. Ji, J. Su, D. Cao and Y. Gao, "Predicting Microblog Sentiments via Weakly Supervised Multimodal Deep Learning," *IEEE Transactions on Multimedia*, vol. 20, no. 4, pp. 997-1007, 2018.
- [53] S. Poria, E. Cambria, N. Howard, G.-B. Huang and A. Hussain, "Fusing audio, visual and textual clues for sentiment analysis from multimodal content," *Neurocomputing*, vol. 174, pp. 50-59, 2016.
- [54] S. Poria, N. Majumder, D. Hazarika, E. Cambria, A. Gelbukh and A. Hussain, "Multimodal Sentiment Analysis: Addressing Key Issues and Setting Up the Baselines," *IEEE Intelligent Systems*, vol. 33, no. 6, pp. 17-25, 2018.
- [55] T. U. Haque, N. N. Saber and F. M. Shah, "Sentiment analysis on large scale Amazon product reviews," *IEEE International Conference on Innovative Research and Development (ICIRD)*, pp. 1-6, 2018.

- 
- [56] Z. Singla, S. Randhawa and S. Jain, "Statistical and sentiment analysis of consumer product reviews.," *8th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, pp. 1-6, 2017.
- [57] P. Singh , A. Dave and K. Dar, "Demonetization: sentiment and retweet analysis.," *International Conference on Inventive Computing and Informatics (ICICI)*, pp. 891-896, 2017.
- [58] A. Bhardwaj, Y. Narayan, V. P. and M. Dutta, "Sentiment Analysis for Indian Stock Market Prediction Using Sensex and Nifty," *Procedia Computer Science*, vol. 70, pp. 85-91, 2015.
- [59] T. Rao and S. Srivastava, "Analyzing stock market movements using Twitter sentiment analysis," *Proceedings of the 2012 international conference on advances in social networks analysis*, pp. 119-123, 2012.
- [60] F. Xu and V. Keselj, "Collective sentiment mining of microblogs in 24-hour stock price movement prediction," *IEEE 16th Conference on Business Informatics*, pp. 60-67, 2014.
- [61] B. Pang and L. Lee, "A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts," *Proceedings of the 42nd annual meeting on Association for Computational*, p. 271, 2004.
- [62] A. Abdi, S. M. Shamsuddin, S. Hasan and J. Piran, "Machine learning-based multi-documents sentiment-oriented summarization," *Expert Systems with Applications*, vol. 109, pp. 66-85, 2018.
- [63] A. Montejo-Ráez, M. Díaz-Galiano, F. Martínez-Santiago and L. Ureña-López, "Crowd explicit sentiment analysis," *Knowledge-Based Systems*, vol. 69, pp. 134-139, 2014.
- [64] Z. Zhang, Q. Ye, Z. Zhang and Y. Li, "Sentiment classification of internet restaurant reviews written in Cantonese," *Expert Systems with Applications*, vol. 38, pp. 7674-7682, 2011.
- [65] S. Narr, M. Hulfenhaus and S. Albayrak, "Language-independent twitter sentiment analysis," *Knowledge discovery and machine learning (KDML)*, pp. 12-14, 2012.
- [66] A. Abbasi, H. Chen and A. Salem, "Sentiment analysis in multiple languages: feature selection for opinion classification in web forums," *ACM Transactions on Information Systems (TOIS)*, vol. 26, no. 3, pp. 1-34, 2008.
- [67] Z. Zhang, Y. Zou and C. Gan, "Textual sentiment analysis via three different attention convolutional neural networks and cross-modality consistent regression," *Neurocomputing*, vol. 275, pp. 1407-1415, 2017.
- [68] Y. Liu, J.-W. Bi and Z.-P. Fan, "Ranking products through online reviews: A method based on sentiment analysis technique and intuitionistic fuzzy set theory," *Information Fusion*, vol. 36, pp. 149-161, 2017.
- [69] S. Kiritchenko, X. Zhu and S. M. Mohammad, "Sentiment analysis of short informal texts," *Journal of Artificial Intelligence Research*, vol. 50, pp. 723-762, 2014.
- [70] S. Baccianella, A. Esuli and F. Sebastiani, "SENTIWORDNET 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining,"

- Proceedings of the seventh conference on international language resources*, p. 2200–2204, 2010.
- [71] S. M. Mohammad, S. Kiritchenko and X. Zhu, “NRC-Canada: building the state-of-the-art in sentiment analysis,” *Proceedings of the seventh international workshop on semantic evaluation*, pp. 321-327, 2013.
- [72] E. Ohn-Bar and M. M. Trivedi, “Multi-scale volumes for deep object detection and localization,” *Pattern Recognition*, vol. 61, pp. 557-572, 2017.
- [73] K. Noda, Y. Yamaguchi, K. Nakadai, H. G. Okuno and T. Ogata, “Audio-visual speech recognition using deep learning,” *Applied Intelligence*, vol. 42, p. 722–737, 2015.
- [74] O. K. Oyedotun and A. Khashman, “Deep learning in vision-based static hand gesture recognition,” *Neural Computing and Applications*, vol. 28, no. 12, pp. 3941-3951, 2017.
- [75] R. R. Chowdhury, M. S. Hossain, S. Hossain and K. Andersson, “Analyzing Sentiment of Movie Reviews in Bangla by Applying Machine Learning Techniques,” *International Conference on Bangla Speech and Language Processing (ICBSLP)*, pp. 1-6, 2019.
- [76] M. Wang, D. Cao, L. Li, S. Li and R. Ji, “Microblog sentiment analysis based on cross-media bag-of-words model,” *Proceedings of international conference on internet multimedia computing and service*, pp. 76-80, 2014.
- [77] Y. Li and B. Shen, “Research on sentiment analysis of microblogging based on LSA and TF-IDF,” *3rd ieee international conference on computer and communications*, pp. 2584-2588, 2017.
- [78] T. Mikolov, I. Sutskever, K. Chen, G. Corrado and J. Dean, “Distributed Representations of Words and Phrases and their Compositionality,” *Advances in neural information processing systems*, pp. 3111-3119, 2013.
- [79] A. Severyn and A. Moschitti, “Twitter Sentiment Analysis with Deep Convolutional Neural Networks,” *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 959-962, 2015.
- [80] X. Fu, W. Liu, Y. Xu and L. Cui, “Combine HowNet Lexicon to Train Phrase Recursive Autoencoder for Sentence-Level Sentiment Analysis,” *Neurocomputing*, vol. 241, pp. 18-27, 2017.
- [81] Y. Kim, “Convolutional neural networks for sentence classification,” *Empirical Methods in Natural Language Processing*, pp. 1746-1751, 2014.
- [82] J. Pennington, R. Socher and C. D. Manning, “Glove: Global vectors for word representation,” *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pp. 1532-1543, 2014.
- [83] K. Cao and M. Rei, “A joint model for word embedding and word morphology,” *Proceedings of the 1st workshop on representation learning for NLP*, p. 18–26, 2016.
- [84] Y. Zhang and B. C. Wallace, “A Sensitivity Analysis of (and Practitioners’ Guide to) Convolutional Neural Networks for Sentence Classification,” *Proceedings of the The 8th International Joint Conference on Natural Language*

- Processing*, p. 253–263, 2017.
- [85] R. Collobert and J. Weston, “A unified architecture for natural language processing: deep neural networks,” *Proceedings of the 25 th International Conference on Machine Learning*, pp. 160-167, 2008.
- [86] R. Socher, A. Perelygin, J. Y. Wu, J. Chuang, C. D. Manning, A. Y. Ng and C. Potts, “Recursive deep models for semantic compositionality over a sentiment treebank,” *Proceedings of the 2013 conference on empirical methods in natural language processing*, pp. 1631-1642, 2013.
- [87] A. Hassan and A. Mahmood, “Deep learning approach for sentiment analysis of short texts,” *3rd International conference on control, automation and robotics (ICCAR)*, p. 705–710, 2017b.
- [88] Z. Zhang, L. Wang, Y. Zou and C. Gan, “The optimally designed dynamic memory networks for targeted sentiment classification,” *Neurocomputing*, vol. 309, pp. 36-45, 2018.
- [89] H. Yanagimoto, M. Shimada and A. Yoshimura, “Document similarity estimation for sentiment analysis using neural network,” *2013 IEEE/ACIS 12th International Conference on Computer and Information Science (ICIS)*, pp. 105-110, 2013.
- [90] M. Yuan, H. Tang and H. Li, “Real-time keypoint recognition using restricted Boltzmann machine,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 25, no. 11, pp. 2119-2126, 2014.
- [91] K. Song, T. Yao, Q. Ling and T. Mei, “Boosting image sentiment analysis with visual attention,” *Neurocomputing*, vol. 312, pp. 218-228, 2018.
- [92] M. Yang, Q. Qu, X. Chen, C. Guo, Y. Shen and K. Lei, “Feature-enhanced attention network for target-dependent sentiment classification,” *Neurocomputing*, vol. 307, pp. 91-97, 2018.
- [93] F.-C. Yang, A. J. Lee and S.-C. Kuo, “Mining Health Social Media with Sentiment Analysis,” *Journal of medical systems*, vol. 40, no. 11, pp. 1-8, 2016.
- [94] S. Sabra, K. M. Malik and M. Alobaidi, “Prediction of venous thromboembolism using semantic and sentiment analyses of clinical narratives,” *Computers in Biology and Medicine*, vol. 94, pp. 1-10, 2018.
- [95] M. Moh, T. Moh, Y. Peng and L. Wu, “On adverse drug event extractions using twitter sentiment analysis,” *Network Modeling Analysis in Health Informatics and Bioinformatics*, vol. 6, no. 1, pp. 1-12, 2017.
- [96] S. M. Jiménez-Zafra, M. T. Martín-Valdivia, M. D. Molina-González and L. A. Ureña-López, “How do we talk about doctors and drugs? Sentiment analysis in forums expressing opinions for medical domain,” *Artificial intelligence in medicine*, vol. 93, pp. 50-57, 2019.
- [97] I. Korkontzelos, A. Nikfarjam, M. Shardlow, A. Sarker, S. Ananiadou and G. H. Gonzalez, “Analysis of the effect of sentiment analysis on extracting adverse drug reactions from tweets and forum posts,” *Journal of Biomedical Informatics*, vol. 62, pp. 148-158, 2016.
- [98] N. Limsopatham and N. Collier, “Normalising Medical Concepts in Social Media Texts by Learning Semantic Representation,” *Proceedings of the 54th*

- Annual Meeting of the Association for Computational Linguistics*, p. 1014–1023, 2016.
- [99] Y. Chen, B. Zhou, W. Zhang, G. Gong and G. Sun, “Sentiment Analysis Based on Deep Learning and Its Application in Screening for Perinatal Depression,” *IEEE Third International Conference on Data Science in Cyberspace*, pp. 451-456, 2018.
- [100] H. Grisstte and E. Nfaoui, “Daily life patients Sentiment Analysis model based on well-encoded embedding vocabulary for related-medication text,” *IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, pp. 921-928, 2019.
- [101] H. Talpada, M. N. Halgamuge and N. T. Q. Vinh, “An Analysis on Use of Deep Learning and Lexical-Semantic Based Sentiment Analysis Method on Twitter Data to Understand the Demographic Trend of Telemedicine,” *11th International Conference on Knowledge and Systems Engineering (KSE), Vietnam*, pp. 1-9, 2019.
- [102] S. Poria, D. Hazarika, N. Majumder and R. Mihalcea, “Beneath the Tip of the Iceberg: Current Challenges and New Directions in Sentiment Analysis Research,” *arXiv:2005.00357v2*, 2020.
- [103] S. Siersdorfer, E. Minack, F. Deng and J. Hare, “Analyzing and Predicting Sentiment of Images on the Social Web,” in *18th ACM international conference on Multimedia*, USA, 2010.
- [104] V. Vonikakis and S. Winkler, “Emotion-Based Sequence of Family Photos,” in *20th ACM international conference on Multimedia*, Japan, 2012.
- [105] J. Jia, S. Wu, X. Wang, P. Hu, L. Cai and J. Tang, “Can We Understand van Gogh's Mood? Learning to Infer Affects from Images in Social Networks,” in *20th ACM international conference on Multimedia*, Japan, 2012.
- [106] B. Li, S. Feng, W. Xiong and W. Hu, “Scaring or Pleasing: Exploit Emotional Impact of An Image,” in *20th ACM international conference on Multimedia*, Japan, 2012.
- [107] S. Wang, J. Wang, Z. Wang and Q. Ji, “Multiple Emotion Tagging for Multimedia Data by Exploiting High-Order Dependencies Among Emotions,” *IEEE Transactions on Multimedia*, vol. 17, no. 12, pp. 2185-2197, 2015.
- [108] D. Borth, R. Ji, T. Chen, T. Breuel and S.-F. Chang, “Large-scale visual sentiment ontology and detectors using adjective noun pairs,” in *21st ACM international conference on Multimedia*, Spain, 2013.
- [109] J. Yuan, Q. You, S. Mcdonough and J. Luo, “Sentribute : Image Sentiment Analysis from a Mid-level Perspective,” in *Second International Workshop on Issues of Sentiment Discovery and Opinion Mining*, Chicago, 2013.
- [110] S. Zhao, Y. Gao, X. Jiang, H. Yao, T.-s. Chua and X. Sun, “Exploring Principles-of-Art Features For Image Emotion Recognition,” in *22nd ACM International conference on Multimedia*, Florida, 2014.
- [111] Y.-y. Chen, T. Chen, T. Liu, H.-Y. M. Liao and S.-f. Chang, “Assistive Image Comment Robot — A Novel Mid-Level Concept-Based Representation,” *IEEE Transactions on Affective Computing*, vol. 6, no. 3, pp. 298-311, 2015.

- 
- [112] Q. You, J. Luo, H. Jin and J. Yang, "Robust Image Sentiment Analysis Using Progressively Trained and Domain Transferred Deep Networks," in *Twenty-Ninth AAAI Conference on Artificial Intelligence*, USA, 2015.
- [113] J. Yang, D. She, M. Sun, M.-m. Cheng, P. L. Rosin and L. Wang, "Visual Sentiment Prediction based on Automatic Discovery of Affective Regions," *IEEE Transactions on Multimedia*, vol. 20, pp. 2513-2525, 2018.
- [114] H. Xiong, Q. Liu, S. Song and Y. Cai, "Region-based convolutional neural network using group sparse regularization for image sentiment classification," *EURASIP Journal on Image and Video Processing*, vol. 30, pp. 1-9, 2019.
- [115] Z. Li, Y. Jiao, X. Yang, T. Zhang and S. Huang, "3D Attention-Based Deep Ranking Model for Video Highlight Detection," *IEEE Transactions on Multimedia*, vol. 20, no. 10, pp. 2693-2705, 2018.
- [116] B. Zhao, X. Wu, J. Feng, Q. Peng and S. Yan, "Diversified Visual Attention Networks for Fine-Grained Object Classification," *IEEE Transactions on Multimedia*, vol. 19, no. 6, pp. 1245-1256, 2017.
- [117] D. She, J. Yang, M.-M. Cheng, Y.-K. Lai, P. L. Rosin and L. Wang, "WSCNet: Weakly Supervised Coupled Networks for Visual Sentiment Classification and Detection," *IEEE Transactions on Multimedia*, 2019.
- [118] S. Fan, M. Jiang, B. L. Koenig, J. Xu, M. S. Kankanhalli and Q. Zhao, "Emotional Attention: A Study of Image Sentiment and Visual Attention," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Salt Lake, 2018.
- [119] S. Fan, M. Jiang, Z. Shen, B. L. Koenig, M. S. Kankanhalli and Q. Zhao, "The Role of Visual Attention in Sentiment Prediction," in *25th ACM international conference on Multimedia*, California, 2017.
- [120] J. Lee, S. Kim, S. Kim, J. Park and K. Sohn, "Context-Aware Emotion Recognition Networks," in *Proceedings of the IEEE International Conference on Computer Vision*, Seoul, 2019.
- [121] V. S. Bawa and V. Kumar, "Emotional sentiment analysis for a group of people based on transfer learning with a multi-modal system," *Neural Computing and Applications*, vol. 31, no. 12, pp. 9061-9072, 2018.
- [122] C. Baecchi, T. Uricchio, M. Bertini and A. D. Bimbo, "A multimodal feature learning approach for sentiment analysis of social network multimedia," *Multimedia Tools and Applications*, vol. 75, no. 5, pp. 2507-2525, 2016.
- [123] Q. Fang, C. Xu, J. Sang, M. S. Hossain and G. Muhammad, "Word-of-mouth understanding: Entity-centric multimodal aspect-opinion mining in social media," *IEEE Transactions on Multimedia*, vol. 17, no. 12, pp. 2281-2296, 2015.
- [124] S. Dai and H. Man, "Integrating Visual and Textual Affective Descriptors for Sentiment Analysis of Social Media Posts," in *IEEE Conference on Multimedia Information Processing and Retrieval*, Florida, 2018.
- [125] N. Xu, "Analyzing multimodal public sentiment based on hierarchical semantic attentional network," in *IEEE International Conference on Intelligence and Security Informatics (ISI)*, China, 2017.
- [126] F. Chen, R. Ji, J. Su, D. Cao and Y. Gao, "Predicting Microblog Sentiments via

- Weakly Supervised Multi-Modal Deep Learning,” *IEEE Transactions on Multimedia*, vol. 20, no. 4, pp. 997-1007, 2017.
- [127] Z. Zhao, H. Zhu, Z. Xue, Z. Liu, J. Tian, M. Chua and M. Liu, “An image-text consistency driven multimodal sentiment analysis approach for social media,” *Information Processing and Management*, vol. 56, no. 6, p. 102097, 2019.
- [128] J. Yu, J. Jiang and R. Xia, “Entity-Sensitive Attention and Fusion Network for Entity-Level Multimodal Sentiment Classification,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 429-439, 2019.
- [129] L. Meng, A.-H. Tan and D. Xu, “Semi-supervised heterogeneous fusion for multimedia data co-clustering,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 26, no. 9, pp. 2293-2306, 2013.
- [130] S. Poria, I. Chaturvedi, E. Cambria and A. Hussain, “Convolutional MKL Based Multimodal Emotion Recognition and Sentiment Analysis,” in *IEEE 16th International Conference on Data Mining*, Spain, 2016.
- [131] J. Xu, F. Huang, X. Zhang, S. Wang, C. Li, Z. Li and Y. He, “Visual-textual sentiment classification with bi-directional multi-level attention networks,” *Knowledge-Based Systems*, vol. 178, pp. 61-73, 2019.
- [132] J. Xu, F. Huang, X. Zhang, S. Wang, C. Li, Z. Li and Y. He, “Sentiment analysis of social images via hierarchical deep fusion of content and links,” *Applied Soft Computing Journal*, vol. 80, pp. 387-399, 2019.
- [133] F. Huang, X. Zhang, Z. Zhao, J. Xu and Z. Li, “Image-text sentiment analysis via deep multimodal attentive fusion,” *Knowledge-Based Systems*, vol. 167, pp. 26-37, 2019.
- [134] B. Zoph, V. Vasudevan, J. Shlens and Q. V. Le, “Learning Transferable Architectures for Scalable Image Recognition,” in *IEEE conference on computer vision and pattern recognition*, Utah, United States, 2018.
- [135] F. Wang, M. Jiang, C. Qian, S. Yang, C. Li, H. Zhang, X. Wang and X. Tang, “Residual Attention Network for Image Classification,” in *IEEE Conference on Computer Vision and Pattern Recognition*, United States, 2017.
- [136] B. Zoph and Q. V. Le, “Neural architecture search with reinforcement learning,” in *International Conference on Learning representations*, Canada, 2017.
- [137] J. Schulman, F. Wolski, P. Dhariwal, A. Radford and O. Klimov, “Proximal Policy Optimization Algorithms,” in *arXiv:1707.06347v2*, 2017.
- [138] J. Machajdik and A. Hanbury, “Affective image classification using features inspired by psychology and art theory,” in *ACM International Conference on Multimedia*, Italy, 2010.
- [139] A. Zadeh, R. Zellers, E. Pincus and L.-P. Morency, “MOSI: Multimodal Corpus of Sentiment Intensity and Subjectivity Analysis in Online Opinion Videos,” *arXiv preprint arXiv:1606.06259*, 2016.
- [140] A. Zadeh, P. P. Liang, J. Vanbriesen, S. Poria, E. Tong, E. Cambria, M. Chen and L.-P. Morency, “Multimodal language analysis in the wild: Cmu-mosei dataset and interpretable dynamic fusion graph,” *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, pp. 2236-2246, 2018.
- [141] M. Katsurai and S. Satoh, “Image sentiment analysis using latent correlations



- among visual, textual, and sentiment views.,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Spain, 2016.
- [142] Q. You, J. Luo, H. Jin and J. Yang, “Building a Large Scale Dataset for Image Emotion Recognition: The Fine Print and the Benchmark,” in *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence (AAAI-16)*, Arizona, 2016.
- [143] Z. Sun, P. K. Sarma, W. Sethares and E. P. Bucy, “Multi-modal Sentiment Analysis using Deep Canonical Correlation Analysis,” in *The 20th Annual Conference of the International Speech Communication Association (INTERSPEECH)*, Austria, 2019.
- [144] A. Zadeh, P. P. Liang, S. Poria, P. Vij, E. Cambria and L.-P. Morency, “Multi-attention recurrent network for human communication comprehension,” in *Thirty-Second AAAI Conference on Artificial Intelligence*, Louisiana, 2018.
- [145] Z. Li, Y. Fan, W. Liu and F. Wang, “Image sentiment prediction based on textual descriptions with adjective noun pairs,” *Multimedia Tools and Applications*, vol. 77, no. 1, pp. 1115-1132, 2017.
- [146] V. Campos, B. Jou and X. Giro-i-Nieto, “From Pixels to Sentiment : Fine-tuning CNNs for Visual Sentiment Prediction,” *Image and Vision Computing*, vol. 65, pp. 15-22, 2017.
- [147] V. Campos, A. Salvador, B. Jou and X. Giró-i-nieto, “Diving Deep into Sentiment : Understanding Fine-tuned CNNs for Visual Sentiment Prediction,” in *1st International Workshop on Affect & Sentiment in Multimedia*, Australia , 2015.
- [148] J. Wang, J. Fu, Y. Xu and T. Mei, “Beyond Object Recognition : Visual Sentiment Analysis with Deep Coupled Adjective and Noun Neural Networks,” in *Twenty-Fifth International Joint Conference on Artificial Intelligence*, New York, 2016.
- [149] K. Song, T. Yao, Q. Ling and T. Mei, “Boosting image sentiment analysis with visual attention,” *Neurocomputing*, vol. 312, pp. 218-228, 2018.
- [150] J. Islam and Y. Zhang, “Visual Sentiment Analysis for Social Images Using Transfer Learning Approach,” in *IEEE International Conferences on Big Data and Cloud Computing (BDCloud), Social Computing and Networking (SocialCom), Sustainable Computing and Communications (SustainCom)*, Atlanta, 2016.
- [151] R. Sharma, L. N. Tan and F. Sadat, “Multimodal Sentiment Analysis Using Deep Learning,” in *17th IEEE International Conference on Machine Learning and Applications*, Florida, 2018.
- [152] T. Rao, M. Xu, H. Liu, J. Wang and I. Burnett, “Multi-scale blocks based image emotion classification using multiple instance learning,” in *IEEE International Conference on Image Processing (ICIP)*, Arizona, 2016.
- [153] T. Rao, M. Xu and H. Liu, “Generating affective maps for images,” *Multimedia Tools and Applications*, vol. 77, no. 13, p. 17247–17267, 2018.
- [154] X. Wang, J. Jia, J. Yin and L. Cai, “Interpretable aesthetic features for affective image classification,” in *IEEE International Conference on Image Processing*,

- Australia, 2013.
- [155] X. Liu, N. Li and Y. Xia, “Affective image classification by jointly using interpretable art features,” *Journal of Visual Communication and Image Representation*, vol. 58, p. 576–588, 2019.
- [156] A. Zadeh, M. Chen, S. Poria, E. Cambria and L.-P. Morency, “Tensor Fusion Network for Multimodal Sentiment Analysis,” in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, Denmark, 2017.
- [157] M. Chen, S. Wang, P. P. Liang, T. Baltrušaitis, A. Zadeh and L.-P. Morency, “Multimodal Sentiment Analysis with Word-Level Fusion and Reinforcement Learning,” in *Proceedings of the 19th ACM International Conference on Multimodal Interaction (ICMI)*, UK, 2017.
- [158] S. H. Dumpala, I. Sheikh, R. Chakraborty and S. K. Kopparapu, “Sentiment Classification on Erroneous ASR Transcripts: A Multi View Learning Approach,” *IEEE Spoken Language Technology Workshop (SLT)*, pp. 807-814, 2018.
- [159] S. H. Dumpala, I. Sheikh, R. Chakraborty and S. K. Kopparapu, “Audio-Visual Fusion for Sentiment Classification using Cross-Modal Autoencoder,” in *32nd Conference on Neural Information Processing Systems (NIPS 2018)*, Canada, 2018.
- [160] D. S. Chauhan, M. S. Akhtar, A. Ekbal and P. Bhattacharyya, “Context-aware Interactive Attention for Multi-modal Sentiment and Emotion Analysis,” in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, China, 5646–5656.
- [161] Z. Sun, P. K. Sarma, W. A. Sethares and Y. Liang, “Learning Relationships between Text, Audio, and Video via Deep Canonical Correlation for Multimodal Language Analysis,” in *arXiv:1911.05544*, 2019.
- [162] H. Li and H. Xu, “Video-Based Sentiment Analysis with hvnLBP-TOP Feature and bi-LSTM,” in *The Thirty-Third AAAI Conference on Artificial Intelligence (AAAI-19)*, Hawaii, 2019.
- [163] M. S. Akhtar, D. S. Chauhan, D. Ghosal, S. Poria, A. Ekbal and P. Bhattacharyya, “Multi-task Learning for Multi-modal Emotion Recognition and Sentiment Analysis,” in *Proceedings of NAACL-HLT*, Minnesota, 2019.
- [164] T. Chen, D. Borth, T. Darrell and S.-F. Chang, “DeepSentibank: Visual sentiment concept classification with deep convolutional neural networks,” *arXiv preprint arXiv:1410.8586*, 2014.
- [165] X. He, H. Zhang, N. Li, L. Feng and F. Zheng, “A Multi-Attentive Pyramidal Model for Visual Sentiment Analysis,” in *International Joint Conference on Neural Networks*, Hungary., 2019.
- [166] J. Yang, D. She, Y.-K. Lai, P. L. Rosin and M.-H. Yang, “Weakly Supervised Coupled Networks for Visual Sentiment Analysis,” in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Salt Lake City, 2018.
- [167] J. Yang, D. She and M. Sun, “Joint Image Emotion Classification and

- Distribution Learning via Deep Convolutional Neural Network,” in *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence (IJCAI-17)*, Australia, 2017.
- [168] X. Zhu, L. Li, W. Zhang, T. Rao, M. Xu, Q. Huang and D. Xu, “Dependency Exploitation: A Unified CNN-RNN Approach for Visual Emotion Recognition,” in *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence*, Australia, 2017.
- [169] J. Yang, D. She, Y.-K. Lai and M.-H. Yang, “Retrieving and Classifying Affective Images via Deep Metric Learning,” in *The Thirty-Second AAAI Conference on Artificial Intelligence (AAAI-18)*, Louisiana, 2018.
- [170] S. Zhao, C. Lin, P. Xu, S. Zhao, Y. Guo, R. Krishna, G. Ding and K. Keutzer, “CycleEmotionGAN: Emotional Semantic Consistency Preserved CycleGAN for Adapting Image Emotions,” in *The Thirty-Third AAAI Conference on Artificial Intelligence (AAAI-19)*, Hawaii, 2019.
- [171] W. Zhang, X. He and W. Lu, “Exploring Discriminative Representations for Image Emotion Recognition with CNNs,” *IEEE Transactions on Multimedia*, 2019.
- [172] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” in *International Conference on Learning Representations*, California, 2015.
- [173] C. Szegedy, S. Ioffe, V. Vanhoucke and A. Alemi, “Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning,” in *31st AAAI Conference on Artificial Intelligence*, Arizona USA, 2016.
- [174] C. Szegedy, V. Vanhoucke, J. Shlens and Z. Wojna, “Rethinking the Inception Architecture for Computer Vision,” in *IEEE conference on Computer Vision and Pattern Recognition*, United States, 2015.
- [175] K. He, X. Zhang, S. Ren and J. Sun, “Deep Residual Learning for Image Recognition,” in *IEEE Conference on Computer Vision and Pattern Recognition*, Las Vegas, 2016.
- [176] F. Chollet, “Xception: Deep Learning with Depthwise Separable Convolutions,” in *IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu, 2017.
- [177] H. L. Wang and L.-F. Cheong, “Affective understanding in film,” *IEEE Transactions on circuits and systems for video technology*, vol. 16, no. 6, pp. 689-704, 2006.
- [178] G. Tu, Y. Fu, B. Li, J. Gao, Y.-G. Jiang and X. Xue, “A Multi-Task Neural Approach for Emotion Attribution, Classification, and Summarization,” *IEEE Transactions on Multimedia*, vol. 22, no. 1, pp. 148-159, 2019.
- [179] K. He, X. Zhang, S. Ren and J. Sun, “Deep Residual Learning for Image Recognition,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, Boston, 2015.
- [180] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu and A. C. Berg, “SSD: Single Shot MultiBox Detector,” in *arXiv:1512.02325v5*, 2016.
- [181] C. Szegedy, S. Ioffe, V. Vanhoucke and A. Alemi, “Inception-v4, Inception-

- ResNet and the Impact of Residual Connections on Learning,” in *Thirty-first AAAI conference on artificial intelligence*, California, 2017.
- [182] Y. Bengio, P. Simard and P. Frasconi, “Learning long-term dependencies with gradient descent is difficult,” *IEEE Transactions on Neural Networks*, vol. 5, no. 2, pp. 157-166, 1994.
- [183] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural computation*, vol. 9, no. 8, pp. 1735-1780, 1997.
- [184] K. Greff, R. K. Srivastava, J. Koutnik, B. R. Steunebrink and J. Schmidhuber, “LSTM: A search space odyssey,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 10, pp. 2222-2232, 2016.
- [185] F. A. Gers, J. Schmidhuber and F. Cummins, “Learning to forget: Continual prediction with LSTM,” in *Ninth International Conference on Artificial Neural Networks*, 1999.
- [186] F. A. Gers and J. Schmidhuber, “Recurrent nets that time and count,” in *Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks*, Italy, 2000.
- [187] K. Cho, D. Bahdanau, F. Bougares, H. Schwenk, Y. Bengio, B. v. Merriënboer and C. Gulcehre, “Learning phrase representations using RNN encoder-decoder for statistical machine translation,” in *arXiv preprint arXiv:1406.1078*, 2014.
- [188] M. Schuster and K. K. Paliwal, “Bidirectional recurrent neural networks,” *IEEE Transactions on Signal Processing*, vol. 45, no. 11, pp. 2673 - 2681, 1997.
- [189] K. Yao, T. Cohn, K. Vylomova, K. Duh and C. Dyer, “Depth-gated LSTM,” *arXiv preprint arXiv:1508.03790*, 2015.
- [190] J. Koutník, K. Greff, F. Gomez and J. Schmidhuber, “A Clockwork RNN,” *arXiv preprint arXiv:1402.3511*, 2014.
- [191] A. Graves and J. Schmidhuber, “Framewise phoneme classification with bidirectional LSTM and other neural network architectures,” *Neural networks*, vol. 18, no. 5-6, pp. 602-610, 2005.
- [192] S. G. Simoes, R. C. Barros, J. Wehrmann and D. D. Ruiz, “Movie Genre Classification with Convolutional Neural Networks,” in *International Joint Conference on Neural Networks (IJCNN)*, Vancouver, 2016.
- [193] J. Wehrmann, R. C. Barros, G. S. Simoes, T. S. Paula and D. D. Ruiz, “(Deep) Learning from Frames,” in *IEEE 5th Brazilian Conference on Intelligent Systems (BRACIS)*, Brazil, 2016.
- [194] J. Wehrmann and R. C. Barros, “Convolutions through Time for Multi-Label Movie Genre Classification,” in *Proceedings of the Symposium on Applied Computing*, Morocco, 2017.
- [195] Y. Deldjoo, M. G. Constantin, B. Ionescu, M. Schedl and P. Cremonesi, “MMTF-14K: A Multifaceted Movie Trailer Feature Dataset for Recommendation and Retrieval,” in *Proceedings of the 9th ACM Multimedia Systems Conference.*, Netherlands, 2018.
- [196] F. M. Harper and J. A. Konstan, “The MovieLens Datasets: History and Context,” *ACM Transactions on Interactive Intelligent Systems*, vol. 5, no. 4, pp. 1-19, 2015.

- 
- [197] Z. Rasheed, Y. Sheikh and M. Shah, "On the Use of Computable Features for Film Classification," *IEEE Transactions on Circuits and Systems for video technology*, vol. 15, no. 1, pp. 52-64, 2005.
- [198] H. Zhou, T. Hermans, A. V. Karandikar and J. M. Rehg, "Movie Genre Classification via Scene Categorization," in *Proceedings of the 18th ACM international conference on Multimedia*, Italy, 2010.
- [199] J. Wehrmann and R. C. Barros, "Movie genre classification: A multi-label approach based on convolutions through time," *Applied Soft Computing*, vol. 61, pp. 973-982, 2017.
- [200] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *arXiv preprint arXiv:1409.1556*, 2015.
- [201] R. R. Selvaraju, A. Das, R. Vedantam, M. Cogswell, D. Parikh and D. Batra, "Grad-CAM: Why did you say that?," in *arXiv preprint arXiv:1611.07450*, 2016.
- [202] J. Wang, W. Wang, L. Wang, Z. Wang, D. D. Feng and T. Tan, "Learning visual relationship and context-aware attention for image captioning," *Pattern Recognition*, vol. 98, p. 107075, 2020.
- [203] S. Woo, J. Park, J.-Y. Lee and . I. S. Kweon, "CBAM: Convolutional Block Attention Module," *Proceedings of the European conference on computer vision (ECCV)*, pp. 3-19, 2018.
- [204] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens and Z. Wojna, "Rethinking the inception architecture for computer vision," *Proceedings of the IEEE conference on computer vision and pattern recognition.*, pp. 2818-2826, 2016.
- [205] H. Ma, W. Li, X. Zhang, S. Gao and S. Lu, "AttnSense: Multi-level Attention Mechanism For Multimodal Human Activity Recognition.," *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*, pp. 3109-3115, 2019.
- [206] T. Niu, "Sentiment analysis on multi-view social data," *International Conference on Multimedia Modeling, Springer*, pp. 15-27, 2016.
- [207] D. Borth, R. Ji, T. Chen, T. Breuel and S.-F. Chang, "Large-scale visual sentiment ontology and detectors using adjective noun pairs," *Proceedings of the 21st ACM international conference on Multimedia*, pp. 223-232, 2013.
- [208] Q. You, J. Luo, H. Jin and J. Yang, "Cross-modality consistent regression for joint visual-textual sentiment analysis of social multimedia," *Proceedings of the Ninth ACM international conference on Web search and data mining*, pp. 13-22, 2016.
- [209] N. Xu and W. Mao, "A residual merged neutral network for multimodal sentiment analysis," *IEEE 2nd International Conference on Big Data Analysis (ICBDA)*, pp. 6-10, 2017.
- [210] N. Xu, W. Mao and G. Chen, "A Co-Memory Network for Multimodal Sentiment Analysis," *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval.*, pp. 929-932, 2018.
- [211] N. Xu and W. Mao, "MultiSentiNet: A Deep Semantic Network for Multimodal Sentiment Analysis," *Proceedings of the 2017 ACM on Conference on*

- 
- Information and Knowledge Management.*, pp. 2399-2402, 2017.
- [212] T. Jiang, J. Wang, Z. Liu and Y. Ling, "Fusion-Extraction Network for Multimodal Sentiment Analysis," *Pacific-Asia Conference on Knowledge Discovery and Data Mining, Springer*, pp. 785-797, 2020.
- [213] J. Xu, Z. Li, F. Huang, C. Li and P. S. Yu, "Social Image Sentiment Analysis by Exploiting Multimodal Content and Heterogeneous Relations," *IEEE Transactions on Industrial Informatics*, pp. 1-8, 2020.
- [214] F. Huang, K. Wei, J. Weng and Z. Li, "Attention-Based Modality-Gated Networks for Image-Text Sentiment Analysis," *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 16, no. 3, pp. 1-9, 2020.
- [215] J. Xu, F. Huang, X. Zhang, S. Wang, C. Li, Z. Li and Y. He, "Sentiment analysis of social images via hierarchical deep fusion of content and links," *Applied Soft Computing*, vol. 80, pp. 387-399, 2019.
- [216] K. Zhang, Y. Zhu, W. Zhang, W. Zhang and Y. Zhu, "Transfer Correlation Between Textual Content to Images for Sentiment Analysis," *IEEE Access*, vol. 8, pp. 35276-35289, 2020.
- [217] S. Lai, L. Xu, K. Liu and J. Zhao, "Recurrent convolutional neural networks for text classification," *Twenty-ninth AAAI conference on artificial intelligence*, p. 2267-2273, 2015.
- [218] Z. Lin, M. Feng, C. Nogueira, M. Yu, B. Xiang, B. Zhou and Y. Bengio, "A structured self-attentive sentence embedding," *International Conference on Learning Representations*, 2017.
- [219] Y. Wang, A. Sun, J. Han, Y. Liu and X. Zhu, "Sentiment Analysis by Capsules," *Proceedings of the 2018 world wide web conference*, pp. 1165-1174, 2018.
- [220] J. Devlin, M.-W. Chang, K. Lee and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," *Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, p. 4171-4186, 2019.
- [221] C. Sievert and K. S. Shirley, "LDAvis: A method for visualizing and interpreting topics," *Proceedings of the workshop on interactive language learning, visualization, and interfaces*, pp. 63-70, 2014.
- [222] K. Stevens, P. Kegelmeyer, D. Andrzejewski and D. Buttler, "Exploring topic coherence over many models and many topics," *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pp. 952-961, 2012.

## **Author's Biography**

---



**Ashima Yadav** received B.Sc. (Hons) in Computer Science from University of Delhi, New Delhi, India in 2013, and Master's in Computer Applications from Guru Gobind Singh Indraprastha University, New Delhi, India in the year 2016. She is currently working towards the Ph.D. degree from the Department of Information Technology, Delhi

Technological University, New Delhi, India.

Her current research interest includes Sentiment Analysis, Deep learning, Natural language processing, Machine learning, and Emotion Recognition. She is a reviewer of various journals of IEEE, Springer. She has been awarded with "Commendable Research Award" by Delhi Technological University, Delhi, India, in the year 2020 and 2021.