Understanding and Explaining Affective Traits in English and Code-mixed Conversations

A THESIS
SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY
BY
SHIVANI KUMAR

UNDER THE SUPERVISION OF
DR. TANMOY CHAKRABORTY

Computer Science and Engineering
INDRAPRASTHA INSTITUTE OF INFORMATION TECHNOLOGY DELHI

NEW DELHI- 110020

March 2024

Understanding and Explaining Affective Traits in English and Code-mixed Conversations

A THESIS
SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY
BY
SHIVANI KUMAR
(PHD19010)

Computer Science and Engineering

INDRAPRASTHA INSTITUTE OF INFORMATION TECHNOLOGY DELHI

NEW DELHI- 110020

March 2024

THESIS CERTIFICATE

This is to certify that the thesis titled **Understanding and Explaining Affective Traits in English and Code-mixed Conversations**, submitted by **Shivani Kumar**, to the Indraprastha Institute of Information Technology Delhi, for the award of the degree of **Doctor of Philosophy**, is a bona fide record of the research work done by her under my supervision. In my opinion, the thesis has reached the standard fulfilling the requirements of the regulations relating to the degree. The contents of this thesis, in full or in parts, have not been submitted to any other Institute or University for the award of any degree or diploma.

Dr. Tanmoy Chakraborty
Thesis Supervisor
Associate Professor
Dept. of Electrical Engineering
IIT Delhi, 110016

Place: New Delhi
Date: March 2024

ACKNOWLEDGEMENTS

First and foremost, my gratitude extends to my advisor, Dr. Tanmoy Chakraborty, whose meticulous guidance and patience have fostered my professional development. Each interaction with him facilitated the formulation of intriguing and viable hypotheses that kept me aligned with my academic objectives. He has cultivated an inclusive group environment that encourages the expression of ideas and fosters open discussions – a dynamic I hope to sustain in our future collaborations. A heartfelt acknowledgment is also due to Dr. Md Shad Akhtar, whose invaluable assistance and adept resolution of both theoretical and practical queries were instrumental throughout my PhD journey. Without the support of these two, completing my thesis would have been an insurmountable challenge.

I extend my appreciation to Dr. Raghava Mutharaju, who, alongside Dr. Md Shad Akhtar, served on my annual progress committee. The committee's insightful feedback played a pivotal role in enhancing the quality of my research. Looking ahead, I anticipate further opportunities to engage with them in addressing new questions and challenges arising from this thesis. Additionally, I express my gratitude to Dr. Sumit Bhatia and Mr. Milan Aggarwal, exceptional mentors during my internship at Adobe. Their exemplary work ethics and emphasis on critical thinking served as an inspiration for my own self-improvement. I would also like to take this opportunity to thank all my co-authors, who were always ready for a discussion and had interesting insights to the problems at hand

During the course of my thesis, I was funded by the University Grants Commission (UGC), for which I am eternally grateful. Without this financial support, it would have been difficult to manage during my tenure as a phd student. I would also like to thank the industries that supported my travel to various conferences, namely Microsoft, Google, and ACM Women. At the same time, I appreciate all the facilities provided by the institute, which certainly helped me focus on matters only relevant to my thesis.

During my time at IIITD, navigating challenges such as the impact of COVID, I found immense support and camaraderie within the Laboratory for Computational Social Systems (LCS2) research group. From undergrads to RAs, each member contributed to making my journey easier. I am particularly grateful to my fellow PhDs in the lab for embracing me, supporting me, and above all, consistently believing in my abilities. They truly made IIITD feel like home. Additionally, I extend my heartfelt appreciation to friends from before my IIITD years. Their steadfast presence and understanding never left me feeling isolated, and for that, I am profoundly thankful.

Last but definitely not the least, I am most thankful to my family. Without their unwavering support, I would not have been able to complete this journey. They are my pillars of strength and I cannot express the debt of gratitude I owe them. The countless sacrifices and the unconditional love of my parents and siblings ensured my ability to pursue my dream. I love them more than anything.

Shivani Kumar

PUBLICATIONS

Journals

J1. Manjot Bedi, **Shivani Kumar**, Md. Shad Akhtar, and Tanmoy Chakraborty, "Multi-Modal Sarcasm Detection and Humor Classification in Code-Mixed Conversations," *IEEE Transactions on Affective Computing*, vol. 14, 2021.

J2. **Shivani Kumar**, Anubhav Shrimal, Md Shad Akhtar, and Tanmoy Chakraborty, "Discovering emotion and reasoning its flip in multi-party conversations using masked memory network and transformer," *Knowledge-Based Systems*, vol. 240, 2022.

J3. **Shivani Kumar**, Shubham Dudeja, Md Shad Akhtar, and Tanmoy Chakraborty, "Emotion Flip Reasoning in Multiparty Conversations," *IEEE Transactions on Artificial Intelligence*, vol. 14, no. 02, 2023, doi: 10.1109/TAI.2023.3289937.

J4. **Shivani Kumar**, Sumit Bhatia, Milan Aggarwal, and Tanmoy Chakraborty, "Dialogue Agents 101: A Beginner's Guide to Critical Ingredients for Designing Effective Conversational Systems," *Natural Language Engineering*. (Under revision)

Conferences

C1. **Shivani Kumar**, Atharva Kulkarni, Md Shad Akhtar, and Tanmoy Chakraborty, "When did you become so smart, oh wise one?! Sarcasm Explanation in Multi-modal Multi-party Dialogues," *Association for Computational Linguistics (ACL)*, 2022.

C2. **Shivani Kumar**, Ishani Mondal, Md Shad Akhtar, and Tanmoy Chakraborty, "Explaining (Sarcastic) Utterances to Enhance Affect Understanding in Multimodal Dialogues," *AAAI Conference on Artificial Intelligence (AAAI)*, 2023.

C3. **Shivani Kumar**, S Ramaneswaran, Md Shad Akhtar, and Tanmoy Chakraborty, "From Multilingual Complexity to Emotional Clarity: Leveraging Commonsense to Unveil Emotions in Code-Mixed Dialogues," *Empirical Methods in Natural Language Processing (EMNLP)*, 2023.

C4. **Shivani Kumar**, Tanmoy Chakraborty, "Harmonizing Code-mixed Conversations: Personality-assisted Code-mixed Response Generation in Dialogues," *Findings of European Chapter Of The Association For Computational Linguistics (EACL findings), 2024.*

C5. **Shivani Kumar**, Rishabh Gupta, Md Shad Akhtar, Tanmoy Chakraborty, "Adding SPICE to Life: Speaker Profiling in Multiparty Conversations," *Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING), 2024*

ABSTRACT

KEYWORDS: Conversational AI ; Affects in Conversations ; Dialogue Agents; Natural Language Processing

In the past decade, Natural Language Processing has undergone a transformative journey, marked by profound changes. The realm of conversational discourse, in particular, has witnessed remarkable advancements, with contemporary systems exhibiting significant potential. The ubiquitous integration of conversational agents into our daily lives often obscures the intricate computations underpinning their functionality. Yet, instances of non-empathetic responses or a failure to grasp nuances like humour or sarcasm serve as stark reminders that our interactions extend to the realm of machines. Addressing this limitation forms the core of our research, which revolves around refining a specific facet of conversational understanding – the nuanced focus on affects. Affects in conversation encapsulate a myriad of discourse attributes, including emotions, sarcasm, humour, and speaker profiles, all playing a pivotal role in comprehending the comprehensive meaning inherent in a spoken statement. Our dedicated efforts unfold in the unraveling of these intricate characteristics, aimed at enhancing the interpretative capabilities of dialogue agents. Moreover, we posit that the mere identification of these affective cues inadequately captures the profound essence embedded within the uttered statement. Consequently, our inquiry extends beyond identification to elucidate these affective dimensions, fostering a more profound understanding of conversational discourse. Throughout this thesis, we address multiple novel problem statements, curate innovative datasets, and develop cutting-edge methods tailored to solve each task. Specifically, our focus encompasses the tasks of emotion recognition in conversation, emotion flip reasoning, humour identification, sarcasm detection, sarcasm explanation, and speaker profiling. This thesis, therefore, seeks to establish a foundation for dedicated research in the domain of affects in conversation.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

1. Introduction

1.1 Thesis Overview and Statement

Conversation[1] serves as the cornerstone for the exchange of ideas among individuals (5). Consequently, it has become a significant topic of scrutiny within the realms of linguistics (6), psychology (7), and computation (8; 9). While the notion of interacting with a computer through speech or text is not new (10), it has only recently transformed into a tangible reality with the advent of digital personal assistants, smart speakers, and chatbots, exemplified by the likes of Google Assistant[2], Apple Siri[3], Amazon Alexa[4], and Samsung Bixby[5]. In the field of artificial intelligence, recent advancements have ushered in a new generation of dialogue systems and conversational interfaces (11). These transformative developments are closely intertwined with the accessibility of substantial computational power and the abundance of extensive data resources (12). In today's context, with the widespread availability of such dialogue agents across various facets of life, their conversational capabilities extend beyond the realm of mere utilitarian tasks like booking a table at a restaurant. They are increasingly expected to engage in more casual and chit-chat based conversations (13). Moreover, many studies in the field of linguistics (14; 15; 16) have established that a better understanding of affects, such as emotions and sarcasm, improves the quality of responses of the interlocutors. To this end, it is imperative for a dialogue agent to possess a profound understanding of various affective dimensions inherent to a conversation. Without a comprehensive grasp of these nuanced elements, the generation of effective and contextually relevant responses becomes challenging.

In addition to the intricacies of natural chit-chat conversations, the dynamics of linguistic diversity often come into play, particularly in polyglot societies like India (17; 18). It is not a rare occurrence for individuals to engage in dialogues that seamlessly intermingle multiple languages, forming a linguistic phenomenon known as code-mixing. Within the scope of this thesis, we undertake a comprehensive examination of monolingual English and Hindi-English code-mixed conversations, delving into the intricate fabric of affective dimensions they embody. Our primary investigative lens remains fixed on four pivotal facets of affect inherent to dialogues: emotions, humour, sarcasm, and the speaker profile.

Emotions: The intrinsic connection between emotion and humanity underscores the pivotal role of emotion understanding in the pursuit of human-like AI (19). Within the ever-evolving landscape of NLP, the domain of Emotion Recognition in Conversation (ERC) has emerged as a burgeoning research frontier due to its remarkable capacity to extract valuable insights from social media platforms, its utility as a powerful tool for psychological analysis, and its role in crafting empathetic responses (20). Furthermore, delving into the fundamental catalysts of specific emotions or the dynamics behind emotional transitions proves indispensable in unraveling the intricate web of emotional dynamics within a discourse (21).

Humour and Sarcasm: Moving beyond emotions, delving into the subtle nuances of sarcasm and humour within an utterance emerges as a paramount endeavor in the quest for comprehensive understanding (22; 23). humour, in its delightful playfulness, serves as a pivotal tool for infusing conversations with levity (24), while sarcasm adds a layer of intrigue and complexity to the discourse (25). Furthermore, while the task of identifying humour often presents itself as a relatively self-contained challenge, given

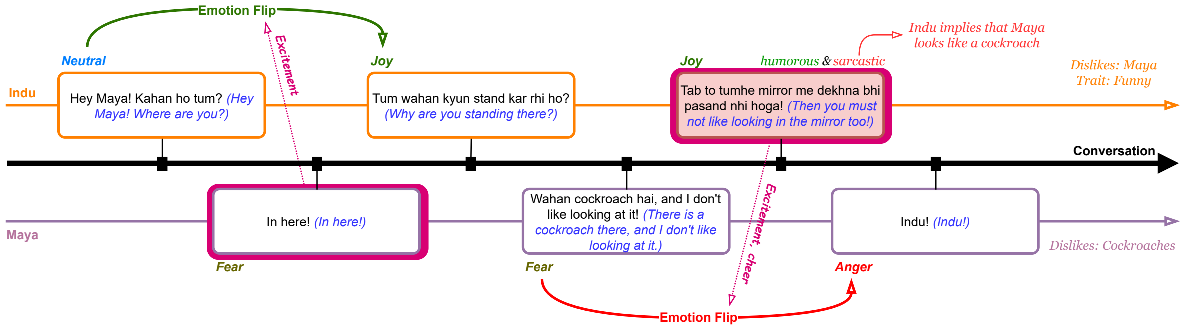
[1] We use the terms 'conversation', 'dialogue', and 'discourse' interchangeably in this thesis.

[2] https://assistant.google.com/

[3] https://www.apple.com/in/siri/

[4] https://alexa.amazon.com/

[5] https://www.samsung.com/us/apps/bixby/



Hey Maya! Kahan ho tum? (Hey Maya! Where are you?)

In here! (In here!)

Tum wahan kyun stand kar rhi ho? (Why are you standing there?)

Wahan cockroach hai, and I don't like looking at it! (There is a cockroach there, and I don't like looking at it.)

Tab to tumhe mirror me dekhna bhi pasand nhi hoga! (Then you must not like looking in the mirror too!)

Indu! (Indu!)

Figure 1.1: **Thesis Overview:** Given a conversation, our aim is to create an affective backdrop to it by focusing on four critical aspects: emotions (their flip and the cause behind the flip), humour, sarcasm (and its explanation), and speaker profile.

the absence of any concealed meaning within amusing utterances, the detection of sarcastic expressions calls for a meticulous and discerning scrutiny to unveil the underlying intent concealed beneath the words.

Speaker Profiling: Each interlocutor in a conversation is shaped by a unique tapestry of life experiences, giving rise to their individualistic personalities (26). This inherent variability naturally culminates in a divergence of potential responses that these distinct speakers might produce, even when confronted with an identical scenario. In light of this perceptible divergence, the notion of discerning and categorizing speaker personas assumes a substantial role. By comprehending and characterizing the distinct speaker identities, it becomes plausible to tailor and condition their responses in a manner that aligns with their respective personas, thereby enhancing the contextual appropriateness and effectiveness of the conversation (27).

Within the scope of this thesis, our focal point centers on the intricate task of discerning and elucidating these multifaceted affective dimensions inherent within conversations. This endeavor is underpinned by a fundamental objective: to foster a deeper and more comprehensive understanding of the dialogues, ultimately paving the way for the generation of responses that are not only more proficient but also more contextually attuned. In pursuit of this overarching goal, we present the thesis statement below and a diagramatical overview in Figure 1.1.

> We aim to understand and explain four primary affective traits, namely emotion, humour, sarcasm, and speaker profile in monolingual English and Hindi-English code-mixed conversations.

1.2 Background

1.2.1 Conversations

The dynamic interplay between two or more individuals engaged in interactive communication is commonly characterized as a conversation or a dialogue. Such exchanges encompass the transmission of signals in various forms, including text, audio, and video modalities. Within the framework of a conversation, we encounter a succession of utterances articulated by the participating speakers. These utterances possess the flexibility to manifest in multiple languages, thereby adding an intricate layer to the conversational landscape. Furthermore, these utterances bear the potential to exhibit a range of affective attributes, such as emotions, humour, and sarcasm, imparting a multifaceted dimension to the discourse.

Broadly, a conversation can take place in two settings – goal-oriented and chit-chat. **Goal-oriented**

conversations focus on achieving a specific objective or task, such as booking a hotel room or providing information. These are typically formal and structured, with the primary aim of accomplishing a predefined goal or obtaining information. **Chit-chat conversations**, on the other hand, are more informal and open-ended. They serve the purpose of social interaction, entertainment, or casual communication, often revolving around topics of general interest or personal anecdotes. Unlike goal-oriented conversations, they lack a specific objective and are more flexible and relaxed in nature. This thesis mostly deals with chit-chat conversations, driven by the significant influence and relevance of affective elements within this context.

1.2.2 Affects

Affects, in general, refers to an entity that cause a change. In other words, something that *affects* someone has an *effect* on them. For instance, the affect of sitting in the sun for too long can cause the effect of tanning. In the context of human communication, affects refer to the wide spectrum of emotional, cognitive, and expressive elements that influence and shape an individual's thoughts, personality, and behavior. These encompass emotions, humour, as well as more complex aspects like sarcasm, and individual characteristics of the speakers involved in a conversation. Affects contribute significantly to the tone, depth, and overall quality of human interactions and are pivotal in understanding the nuances of social and emotional dynamics. In this thesis, we focus on four primary aspects of conversation – emotions, humour, sarcasm, and speaker profile.

1.2.3 Code-mixing

Code-mixing is a linguistic phenomenon that occurs when speakers seamlessly blend two or more languages within a single conversation or discourse. This practice, often observed in multilingual or culturally diverse communities, such as India, allows individuals to draw from their language repertoires to effectively communicate and convey their ideas. Code-mixing can manifest in various forms, such as incorporating loanwords, phrases, or even whole sentences from one language into another. It serves as a dynamic and natural expression of cultural and linguistic hybridity, reflecting the intricate interplay of identities and influences within a given linguistic environment. Understanding code-mixing is pivotal for researchers and language enthusiasts, as it sheds light on the complexities and fluidity of language use in diverse and interconnected societies. In this thesis, along with monolingual English, we focus on Hindi-English code-mixed conversations due to their predominance in the real-world Indian society.

1.2.4 Conversational Agents

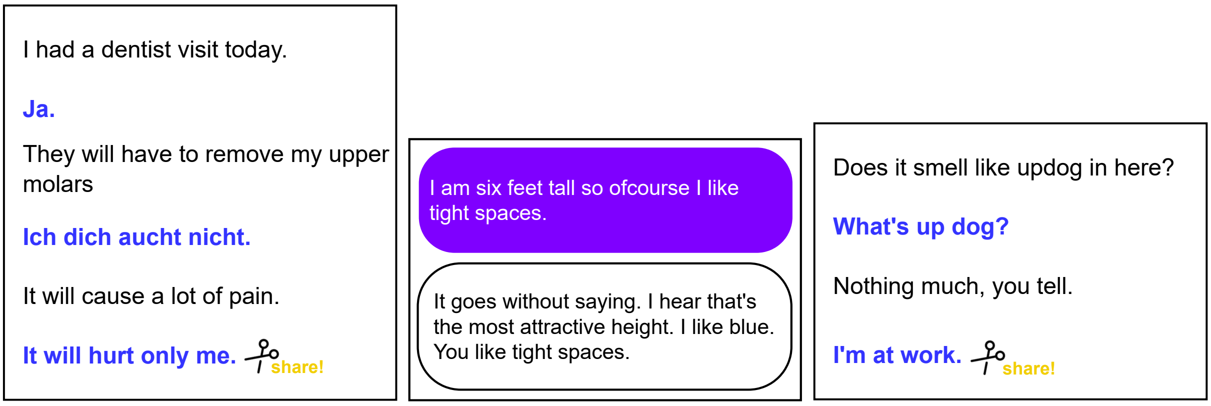
Conversational agents, also known as chatbots, dialogue agents, or virtual assistants, are sophisticated software programs designed to engage in human-like interactions through text, voice, or other communication modalities. These digital entities have evolved significantly, thanks to advancements in AI and NLP. They are capable of understanding and generating human language, enabling them to assist users with a wide array of tasks, from answering questions and providing information to facilitating transactions and even offering emotional support.

In alignment with the distinct categories of dialogues, conversational agents too exhibit a duality, classified as goal-oriented and chit-chat agents. The former, as the name implies, are tailored for conversations driven by specific objectives, such as the reservation of a restaurant table. On the other hand, chit-chat agents engage in more casual and open-ended dialogues, aiming to enhance overall interaction and engagement. Online chatbots like Cleverbot[6], PandoraBots[7], ChatBot[8] and Simisimi[9] fall within the chit-chat category. However, considering real-world demands, a majority of prominent chatbots, like Google Assistant, Apple Siri, and Samsung Bixby, adopt a hybrid approach, proficient in both goal-oriented and chit-chat interactions. In the scope of this thesis, our exclusive focus centers on chit-chat dialogues, driven by the compelling and multifaceted challenges they present.

1.2.5 Affects and Code-mixing in Conversational Agents

Affects and code-mixing in conversational agents represent two fascinating dimensions of human-machine interaction, each laden with its unique complexities. **Affects**, encompassing emotions, humour, sarcasm, and more, introduce the crucial element of intelligence to conversational agents. These chatbots are increasingly expected to comprehend and respond to users' cognitive nuances effectively, making their interactions more contextually relevant. Moreover, the subtleties of humour and sarcasm detection add a layer of sophistication, demanding intricate language analysis and the ability to decipher subtle expressions within dialogues. A comprehensive understanding of these affective cues empowers conversational agents to engage users on a deeper level, fostering more meaningful and enjoyable interactions.

On the other hand, **code-mixing** in conversational agents highlights the intricate interplay of language diversity and cultural influences. In a multilingual and multicultural society, like India, code-mixing is a common practice. For conversational agents to truly serve diverse user groups, they must be equipped to handle code-mixed conversations seamlessly. This entails a thorough understanding of not only individual languages but also the fluid transitions between them. By addressing code-mixing, conversational agents can better adapt to users' linguistic preferences, catering to a broader demographic and enhancing their accessibility. Consequently, the synergy of affects and code-mixing in conversational agents leads to more sophisticated and adaptable AI systems, poised to facilitate effective communication across diverse linguistic and emotional landscapes.



I had a dentist visit today.

Ja.

They will have to remove my upper molars

Ich dich aucht nicht.

It will cause a lot of pain.

It will hurt only me.

I am six feet tall so ofcourse I like tight spaces.

It goes without saying. I hear that's the most attractive height. I like blue. You like tight spaces.

Does it smell like updog in here?

What's up dog?

Nothing much, you tell.

I'm at work.

(a) Dialogue agent is apathetic (b) Dialogue agent is unable to understand sarcasm (c) Dialogue agent is unable to grasp humour

Figure 1.2: Drawbacks of existing dialogue systems. Examples (a) and (c) are taken from `https://www.cleverbot.com/?0`; Example (b) is taken from `https://www.pandorabots.com/mitsuku/`.

[6] https://www.cleverbot.com/?0
[7] https://www.pandorabots.com/mitsuku/
[8] https://www.chatbot.com/
[9] https://simsimi.com/

1.3 Challenges

A typical dialogue agent is composed of three fundamental modules – (i) the input module, (ii) the dialogue understanding module, and (iii) the output generation module. The integral capability to discern and assimilate affective attributes is predominantly embedded within the second module, the dialogue understanding component. Despite notable advancements in dialogue systems, their proficiency in capturing the intricacies of affect within conversations remains limited (28). As illustrated in Figure 1.2, existing online dialogue agents exhibit instances of apathetic responses, as well as a notable inability to comprehend elements of sarcasm and humour.

In Figure 1.2(a), the depicted agent fails to grasp the user's emotional distress, lacking the requisite empathetic response, which is essential in such a sensitive context. Similarly, Figure 1.2(b) showcases an agent's inability to discern the user's sarcastic tone when discussing preferences for confined spaces. Additionally, the challenge of comprehending humour persists, as illustrated in Figure 1.2(c). This persistent deficiency in affective understanding highlights a critical area of improvement in the development of more emotionally intelligent and contextually aware dialogue agents.

1.4 Related Work

Within this thesis, our central focus lies in the comprehensive exploration and explanation of affective dimensions, encompassing emotions, humour, sarcasm, and the distinctive speaker profile, evident in both English and code-mixed conversations. It is worth noting that preceding our research endeavors, numerous studies addressed analogous challenges, both in standalone settings and within the domain of conversational inputs. In this section, we offer a concise overview of some of these prior investigations for context and reference.

Emotion analysis (29; 30) initially revolved around the examination of isolated textual inputs, including tweets (31; 32), online reviews (33; 34), and news articles (35; 36). While ascertaining emotions in such contexts is undeniably significant, these forms of communication lack the dynamic and real-time conversational backdrop. In contrast, the discernment of emotions within the fluid and evolving context of conversations presents a notably intricate and invaluable task (20). This capability holds profound relevance across a spectrum of applications, ranging from the development of chatbots and virtual assistants (37) to mental health support (38) and customer service (39). As a result, a plethora of recent studies have undertaken the challenge of emotion recognition in conversation (20), employing diverse methodologies encompassing traditional machine learning (40; 41; 42), RNN-based approaches (43; 44; 45; 46; 47; 48), and the more contemporary Transformer-based methodologies (49; 50; 51; 52). Identifying emotions serve as the first step in obtaining a comprehensive view of the emotional dynamics in a dialogue, while explaining the cause of emotions serve as the next step, contributing towards explainability (53; 21). To this end, in this thesis, we focus on identifying emotions as well reasoning emotion-flips of speakers in a conversation on the basis of triggers and instigators.

Humour and sarcasm analysis is the next pivotal stage in achieving a profound affective understanding of textual input, following the comprehensive examination of emotions. Much like the progression of emotion recognition, the journey of humour identification initially commenced with the analysis of non-contextual inputs (54; 55; 56). Subsequently, research endeavors extended their purview to encompass contextual inputs, such as conversations, for the nuanced analysis of humour (57). Yet, it is important to note that humour analysis remains comparatively uncomplicated due to the absence of concealed meanings within humorous statements. In contrast, the detection of sarcasm introduces a distinct layer of complexity, demanding an intricate grasp of the latent subtleties within a dialogue (23). As a consequence, sarcasm analysis has garnered substantial attention within the domain of NLP. Various approaches, ranging from the utilization of features like sentiment shifts and contextual incongruity (58; 59) to the application of deep learning models (60; 61; 62), have been explored to unravel the intricacies of sarcasm. Given the implicit nature of sarcasm, mere identification may fall short in comprehending the underlying meaning of the input statement. Therefore, the imperative need arises to delve into the realm of explanation. In alignment with this goal, this thesis embarks on the endeavor of humour and sarcasm identification, complemented by the crucial task of explicating the concealed nuances of sarcasm within the textual context.

Speaker profiling constitutes an indispensable facet in the endeavor to construct a comprehensive and multifaceted affective backdrop for an individual. Given the diverse array of life experiences that shape each individual, distinctive personalities emerge, replete with their own predilections and aversions (26). As a corollary, the integration of personalization within dialogue systems assumes a pivotal role, facilitating the tailored generation of contextually fitting responses for each unique interlocutor (27). Within the domain of conversational agents, the incorporation of personalization has demonstrated its capacity to notably enhance the efficacy of response generation (63; 64; 65; 66; 67; 68). While goal-oriented dialogue systems have previously harnessed user profiles to condition responses and have thereby manifested improved performance (67; 68), recent attention has shifted toward chit-chat settings (69; 70; 63; 64; 65). In this context, user profiles can be deduced from the conversational context, serving as a dynamic foundation for further response customization. With the advent of the PersonaChat dataset (70), an increasing body of research has underscored the advantages of incorporating speaker profiles when generating responses for users (63; 64; 65). While leveraging persona information undeniably results in superior responses, it is essential to refrain from presuming the free availability of such data. Therefore, the initial and imperative step in the development of any personalized dialogue system is the generation or classification of speaker profiles. In alignment with this imperative, this thesis centralizes its focus on the meticulous construction of speaker profiles and their strategic utilization in elevating the performance of response generation.

1.5 Thesis Organization

We now discuss the organization for the rest of the thesis. The entire thesis is divided into three parts as enumerated below. Further, we have provided the full form of the abbreviations used in this thesis in Table 1.1.:

1. **Emotion Analysis:** In this part, we tackle two tasks – Emotion Recognition in Conversation (ERC) and Emotion Flip Reasoning (EFR) for monolingual English and Hindi-English code-mixed dialogues. We propose two datasets, MELD-FR and MELD-I, an extension of the MELD dataset (71) to facilitate the task of EFR in English language. In order to benchmark these datasets, we propose two models – EFR-TX and TGIF. To handle the task of ERC and EFR in code-mixed setting, we develop the E-MaSaC dataset and benchmark it using a novel deep learning architecture, COFFEE.

2. **Humour and Sarcasm Analysis:** This part focuses on identifying the humorous and sarcastic instances in a dialogue and further explaining the sarcastic instances for language comprehension. We utilise the MusTard dataset (2) to gauge the sarcasm in English and propose a new dataset, MaSaC, to analyse humour and sarcasm in code-mixed dialogues. We propose three neural networks – MSH-COMICS, MAF, and MOSES to analyse humour and sarcasm in conversations.

3. **Speaker Profiling:** In this part, we propose SPICE, an English based dataset of conversation with

annotated speaker profile labels. Additionally, we propose SPOT and PA3, two deep learning based architectures to extract speaker profiles from English and code-mixed dialogues, respectively.

Abbreviation	**Full form**
AI	Artificial Intelligence
CNN	Convolutional Neural Networks
COFFEE	COmmonsense aware Fusion For Emotion rEcognition
DPA	Dot Product Attention
EFR	Emotion Flip Reasoning
EFR-TX	EFR-Transformer
ERC	Emotion Recognition in Conversation
ERC-MMN	ERC-Masked Memory Network
FNN	Feedforward Neural Network
GPLMs	Generative Pretrained Models
GRU	Gated Recurrent Unit
kNN	k-Nearest Neighbour
LSTM	Long Short-Term Memory
MAF	Modality Aware Fusion
MOSES	MultimOdal Sarcasm Explanation with Spotlight
MSH-COMICS	Multi-modal Sarcasm Detection and Humor Classification in COde-MIxed ConversationS
NLP	Natural Language Processing
PA3	Personality-Aware Axial Attention
RNN	Recurrent Neural Network
SED	Sarcasm Explanation in Dialogues
SPC	Speaker Profiling in Conversations
SPICE	Speaker Profiling In ConvErsation
SPOT	Speaker PrOfiling using Transformers
SVM	Support Vector Machines
TGIF	Transformer and GRU Inspired Flip reasoner
WITS	Why Is This Sarcastic

Table 1.1: Abbreviations and their definitions used in the thesis.

Part I

Emotion Analysis

2. Emotion Recognition in Conversation

Understanding emotions during conversation constitutes a foundational facet of human communication, propelling the field of NLP to delve into the task of Emotion Recognition in Conversation (ERC). Despite the substantial body of research aimed at discerning emotions in monolingual dialogues, the exploration of emotional dynamics within code-mixed conversations has remained relatively underemphasized. This, in turn, serves as the driving force behind our endeavor to explore ERC within code-mixed conversations, alongside monolingual English, within the scope of this study. In pursuit of this objective, we introduce ERC-MMN, a masked memory network tailored for ERC within the confines of monolingual English, and rigorously evaluate its performance utilizing the widely recognized MELD dataset. Transitioning to the domain of code-mixed dialogues, we acknowledge the imperative role of emotional intelligence intertwined with a profound comprehension of worldly knowledge. As a result, we present an innovative approach that seamlessly integrates common-sense information with the contextual underpinnings of the dialogue, thus paving the way for a deeper comprehension of emotions. To bring this concept to fruition, we devise COFFEE, an astute pipeline architecture specifically engineered to extract pertinent common-sense knowledge from existing knowledge graphs, all based on the code-mixed input. Subsequently, an advanced fusion technique is used to harmoniously blend the acquired commonsense information with the dialogue representation, obtained through a dedicated dialogue understanding module. Furthermore, we meticulously curate a pioneering ERC dataset tailored for Hindi-English code-mixed conversations, E-MaSaC. Through a series of experiments, we demonstrate a notable enhancement in performance stemming from the systematic integration of commonsense knowledge into ERC. This enhancement is substantiated by both quantitative evaluations and in-depth qualitative analyses, reaffirming the central role of common-sense integration in bolstering ERC. In addition to this, we provide anecdotal evidence as well as rigorous qualitative and quantitative error analyses, reinforcing the supremacy of our models in comparison to the baseline approaches.

2.1 Introduction

Early studies in the area of emotion analysis (72; 30) primarily focused on emotion recognition in standalone text (73; 74; 75) which has been shown to be effective in a wide range of applications such as e-commerce (76), social media (77; 78), and health-care (79). Recently, the problem of emotion analysis has been extended to the conversation domain – usually dubbed as **Emotion Recognition in Conversation**, *aka* **ERC** (20), where the inputs are no longer standalone; instead, they appear as a sequence of utterances uttered by more than one speaker. The aim of ERC is to extract the expressed emotion of every utterance in a conversation (or dialogue). Despite the extensive exploration of ERC in numerous studies (43; 44; 46; 48; 50; 45; 47; 49; 80; 51; 52), the primary focus has been into monolingual dialogues, overlooking the prevalent practice of code-mixing. Conversations, in their various forms such as text, audio, visual, or face-to-face interactions (81; 82), can encompass a wide range of languages (83; 84). In reality, it is commonplace for individuals to engage in informal conversations with acquaintances that involve a mixture of languages (17; 18). For instance, two native Hindi speakers fluent in English may predominantly converse in Hindi while occasionally incorporating English words. Figure 2.1 illustrates an example of such a dialogue between two speakers in which each utterance incorporates both English and Hindi words with a proper noun. This linguistic phenomenon, characterized by the blending of multiple languages to convey a single nuanced expression, is commonly referred to as *code-mixing*.

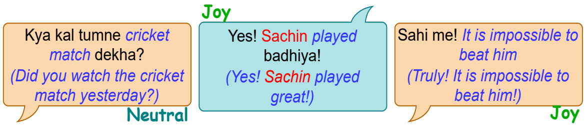

Kya kal tumne *cricket match* dekha?
(Did you watch the cricket match yesterday?)
Neutral
Joy
Yes! Sachin *played* badhiya!
(Yes! Sachin played great!)
Sahi me! *It is impossible to beat him*
(Truly! It is impossible to beat him!)
Joy

Figure 2.1: Example of a code-mixed dialogue, with emotions, between two speakers. Blue denotes English words while red denotes proper noun.

In this work, we aim to perform the task of ERC for monolingual English and Hindi-English code-mixed multi-party dialogues, thereby enabling the modeling of emotion analysis in real-world casual conversations as shown in Figure 2.1 of two speakers with three utterances where the first utterance is identified to be of neutral emotion and the subsequent utterances emits the emotion of joy. We use the popular MELD dataset (85) for the task of ERC in the English language. MELD contains multipartuy conversations from a popular English sitcom. However, to the best of our knowledge, there is no previous work that deals with ERC for code-mixed conversations, leading to a scarcity of available resources in this domain. As a result, we curate a comprehensive dataset, E-MaSaC, comprising code-mixed conversations, where each utterance is meticulously annotated with its corresponding emotion label.

Monolinual English Setting: We propose a masked memory network based framework for ERC . This architecture effectively fuses the dialogue-level global conversational and speaker-level local conversational contexts to learn an enhanced representation for each utterance. Furthermore, we employ a memory network (86) to leverage the historical relationship among all the previous utterances and their associated emotions as additional information. We hypothesize that the memory content at state t can model the dialogue-level emotional dynamics among the speakers so far. Thus, it can be a vital piece of information for the future utterances corresponding to the states $t + y$, where $y = 1, \cdots, n - t$. For evaluation, we use MELD (85), a benchmark ERC dataset. We also perform extensive ablation and comparative studies with five baselines and different variations of our models and obtain state-of-the-art results for the ERC task. A side-by-side diagnostics and anecdotes further explore the errors incurred by the competing models and explain why our models outperform the baselines.

Hindi-English Code-mixed Setting: The elicited emotion in a conversation can be influenced by numerous commonly understood factors that may not be explicitly expressed within the dialogue itself (87). Consider an example in which the phrase "I walked for 20 kilometers" evokes the emotion of *pain*. This association stems from the commonsense understanding that walking such a considerable distance would likely result in fatigue, despite it not being explicitly mentioned. Consequently, capturing commonsense information alongside the dialogue context becomes paramount in order to accurately identify the elicited emotion. To address this, we propose incorporating commonsense for solving the task of ERC. However, the most popular commonsense graphs, such as ConceptNet (88) and COMET (89) are made for English, are known to work for the English language (90; 91), and are not explored for code-mixed input. To overcome this challenge, we develop a pipeline to utilize existing English-based commonsense knowledge graphs to extract relevant knowledge for code-mixed inputs. Additionally, we introduce a clever fusion mechanism to combine the dialogue and commonsense features for solving the task at hand. In summary, our contributions are listed below:

1. We develop a masked memory network based architecture for ERC in monolingual English dialogues, which outperforms several comparable systems.
2. We explore, for the first time, the task of ERC for multi-party code-mixed conversations.

3. We propose a novel code-mixed multi-party conversation dataset, E-MaSaC, in which each discourse is annotated with emotions.
4. We develop COFFEE (COmmonsense aware Fusion For Emotion rEcognition), a method to extract commonsense knowledge from English-based commonsense graphs given code-mixed input and fuse it with dialogue context efficiently.
5. We give a detailed quantitative and qualitative analysis of the results obtained and examine the performance of the popular large language models, including ChatGPT.

2.2 Related Work

Emotion Recognition. Intial emotion analysis (29; 30) primarily centered around the analysis of isolated textual inputs, such as tweets (31; 32), online reviews (33; 34), and news articles (35; 36). While the recognition of emotions within these contexts holds undeniable significance, these forms of communication lack the dynamic and real-time conversational context. In stark contrast, the task of identifying emotions within the fluid and evolving backdrop of conversations is notably complex and invaluable (20). This capability has profound implications across a wide array of applications, ranging from the development of chatbots and virtual assistants (37) to applications in the domains of mental health support (38) and customer service (39). Consequently, numerous recent studies have risen to the challenge of Emotion Recognition in Conversation (ERC) (20). While ERC was solved using heuristics and standard machine learning techniques initially (40; 41; 42), the trend has recently shifted to employing a wide range of deep learning methods (43; 44; 45; 46; 47; 49; 50; 92; 48; 80; 51; 52). The identification of emotions serves as the initial step toward obtaining a comprehensive understanding of the emotional dynamics in a dialogue.

Emotion and Commonsense. Given the implicit and pivotal significance ascribed to commonsense knowledge in the intricate process of emotion identification, researchers have embarked on the formidable journey of incorporating commonsense into the realm of emotion recognition. Particularly in contexts where the textual input stands alone, and the expanse of contextual information is rather limited, scholars advocate for the judicious employment of meticulously curated latent commonsense concepts that possess the seamless capability to harmonize with the textual content (93; 90; 94). However, in more complex scenarios where the textual context extends over extended sequences, such as dialogues, the intelligent and nuanced capture of this contextual information becomes an imperative pursuit. In these intricate scenarios, a multitude of studies delve into the task of ERC characterized by the fusion of commonsense knowledge. This fusion process is facilitated through diverse resources for external knowledge, including ConceptNet (88; 90), Atomic triplets (95; 96), and the comprehensive COMET graph (89; 91; 97).

Emotion and Code-mixing. The current landscape of research pertaining to emotion analysis within the context of code-mixed language predominantly directs its attention towards isolated instances of social media texts (98; 99; 100) and appraisals contained within reviews (101; 102). Within the subdomain of code-mixed conversations, various facets, such as sarcasm (103; 104), humour (105), and instances of offensive language (106), have been scrutinized, leading to insightful investigations. Nevertheless, despite the depth and breadth of exploration into these nuanced dimensions, the domain of emotion analysis, within the specific context of code-mixed conversations, remains conspicuously uncharted. To the best of our knowledge, no pertinent literature exists in this domain, signifying a critical knowledge gap that beckons for comprehensive investigation. It is within this unexplored expanse that our research endeavors to make its significant mark. We embark on a rigorous exploration of the emotionally charged territory of ERC, paying particular attention to code-mixed dialogues that transpire between Hindi and English languages.

2.3 Dataset

Monolingual English – MELD: The MELD dataset (85), an extension of the EmotionLines dataset (107), represents a comprehensive resource for in-depth emotional analysis. Distinguished by its multi-party dialogues, it encompasses a rich array of textual, acoustic, and visual data. These dialogues are sourced from the popular television series, "Friends,"[1] enhancing the dataset's relatability and real-world applicability. Each utterance within these dialogues is meticulously annotated with one of seven distinct emotions, including anger, disgust, sadness, joy, surprise, fear, and neutral. This diverse range of emotions finds its roots in the renowned emotion theory put forth by Paul Ekman (29), with the neutral category thoughtfully added to account for scenarios devoid of emotional expression. In our study, we tailor the MELD dataset to consider only the instances containing atleast one emotion flip, a topic explored extensively in Chapter 3. It is crucial to note that the emotional labels for this specialized subset remain consistent with the original MELD dataset, preserving the dataset's integrity and relevance. Moreover, we adhere to the dataset's established train-validation-test partition, as proposed by its authors. Table 2.1 highlights the salient dataset statistics, offering a succinct yet comprehensive overview of its key characteristics.

Split	**Emotions**							**Total**
	Disgust	**Joy**	**Surprise**	**Anger**	**Fear**	**Neutral**	**Sadness**	
Train	225	1466	1021	911	229	3702	576	8130
Dev	20	156	144	126	39	395	97	977
Test	61	325	238	283	42	943	169	2061

Table 2.1: Statistics of the MELD dataset for ERC. We only consider those dialogues from the original MELD dataset where there is at least one emotion-flip. This step removed 271 dialogues from MELD, resulting $1,161$ dialogues.

Code-mixed Hindi-English – E-MaSaC: A paucity of datasets exists for code-mixed conversations, making tasks on code-mixed dialogues scarce. Nevertheless, our proposed dataset MASAC, introduced in Chapter 4, compiled by extracting dialogues from an Indian TV series, contains sarcastic and humorous Hindi-English code-mixed multi-party instances. We extract dialogues from this dataset and perform annotations for the task of ERC to create E-MaSaC[2]. The resultant data contains a total of $8,607$ dialogues constituting of $11,440$ utterances. Data statistics are summarised in Table 2.2.

Split	**Emotions**								**Total**
	Disgust	**Joy**	**Surprise**	**Anger**	**Fear**	**Neutral**	**Sadness**	**Contempt**	
Train	127	1596	441	819	514	3909	558	542	8506
Dev	21	228	66	118	88	633	126	74	1354
Test	17	349	57	142	122	656	155	82	1580

Table 2.2: Statistics of the E-MaSaC dataset for ERC in Hindi-English code-mixed dialogues.

Emotion Annotation. Given, as input, a sequence of utterances forming a dialogue, D, the aim here is to assign an appropriate emotion, e_i, for each utterance, u_i, uttered by speaker s_j. The emotion e_i should come out of a set of possible emotions, E. Following the standard work in ERC for the English language, we use the latest set of Ekman's emotions as our possible emotions, E = {*anger*, *fear*, *disgust*, *sadness*, *joy*, *surprise*, *contempt*, *neutral*}. We ask three annotators who are linguists fluent in English and Hindi with a good grasp of emotional knowledge with an age between 25-30, (a, b, c) to annotate

[1] `https://www.imdb.com/title/tt0108778/`

[2] We follow the original train-val-test split as is in MASAC

each utterance, u_i, with the emotion they find most suitable for it, e_i^a such that $e_i^a \in E$. A majority voting is done among the three annotations (e_i^a, e_i^b, e_i^c) to select the final gold truth annotation, e_i. Any discrepancies are resolved by a discussion among the annotators; however, such discrepancies are rare. We calculate the inter-annotator agreement, using Kriprendorff's Alpha score (108), between each pair of annotators, $\alpha_{ab} = 0.84$, $\alpha_{bc} = 0.85$, and $\alpha_{ac} = 0.85$. To find out the overall agreement score, we take the average score, $\alpha = 0.85$.

2.4 Proposed Methodology

2.4.1 ERC-MMN: ERC for English Dialogues

For ERC, at each time step t, we learn an emotion label for utterance u_t. We employ an utterance-level memory network as illustrated in Figure 2.2.

A dialogue D can have n utterances spoken by m distinct speakers, and each of these utterances has an associated emotion label E. We model ERC as the sequence-labeling problem, where for each utterance in the dialogue sequence, we predict its corresponding label. In our model, we employ m separate speaker-level forward GRUs (sGRU^{s_j})[3] to capture the utterance pattern of each speaker $s_j \in S$. The hidden representations of each sGRU^{s_j} are arranged in the dialogue order and subsequently fed to cGRU for emotion recognition. For each speaker $s_j \in$ S, we compute a d-dimensional speaker-level hidden representation as follows:

$$[\bar{h}_1^{s_j}, .., \bar{h}_k^{s_j}, .., \bar{h}_p^{s_j}] = \text{sGRU}^{s_j}(v_1^{s_j}, .., v_k^{s_j}, .., v_p^{s_j})$$

where $v_k^{s_j} \in \text{V}^{s_j}$ denotes the utterance spoken by speaker s_j in the dialogue, and p is the number of utterances uttered by speaker s_j. Evidently,

$$U \ = \ \forall_{s_j \in S} \quad union(\text{V}^{s_j})$$

Using stacked GRUs, our model becomes agnostic to the number of speakers present in the dialogue. Note that the modeling of a speaker-level GRU, e.g., sGRU^{s_j}, is in isolation with other speaker-level GRUs (i.e., sGRU^{s_k}, where $j \neq k$). However, a natural conversation does not happen in isolation; therefore, to provide a mechanism for the interaction among the speakers and to model the natural conversation, we leverage the dialogue-level context in the learning process of the speaker dynamics. The dialogue-level context is maintained through a global GRU (gGRU) which is shared across all the speakers within a dialogue. For each utterance $u_i^{s_j}$, we compute an association with the previous global state gGRU$_{[u_1,...,u_{i-1}]}$ through an attention mechanism. It ensures that the current utterance is aware of the dialogue-level context. The context-aware attended representation is then forwarded to sGRU^{s_j} to learn the speaker-specific conversation dynamics. Mathematically,

$$v_k^{s_j} = \text{Attention}(\text{gGRU}_{[u_1,...,u_{i-1}]}, u_i^{s_j}) \oplus u_i^{s_j}$$

where $u_i^{s_j}$ denotes the i^{th} utterance in the dialogue sequence, and $v_k^{s_j}$ denotes the corresponding k^{th} utterance in the speaker sequence s_j. In parallel, the attended vector is consumed by gGRU to update the global state of the dialogue, i.e.,

$$\text{gGRU}_{[u_1,...,u_i]} = \text{gGRU}([u_1, ..., u_{i-1}], v_k^{s_j}, d_i^{s_j})$$

[3] To model the natural conversation, we did not account for any future context anywhere in the architecture.

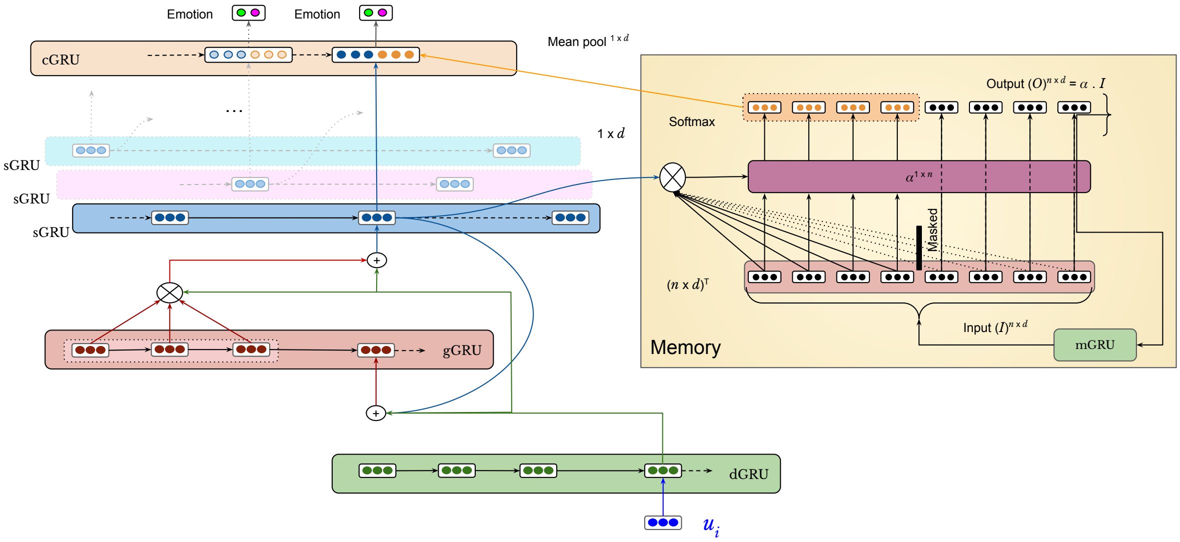


Figure 2.2: The proposed masked memory network based ERC-MMN. ERC-MMN takes a dialogue (a sequence of utterances) and aims to predict the emotion of each utterance in order.

where $d_i^{s_j}$ represents the hidden representation of utterance u_i in the dialogue sequence.

Masked Memory Network. To learn the dialogue conversation efficiently, the role of each utterance u_i in the dialogue needs to be carefully examined. Some utterances have lesser importance in the dialogue context, whereas others have a long-lasting effect. In general, there is a higher chance that important utterances may participate in predicting emotions for multiple utterances. In our manual analysis of the MELD dataset, we also observe a reasonable correspondence between a few utterances and the emotional labels for multiple utterances in the dialogue. Hence, we hypothesize that the information regarding these few utterances may be exploited by the future utterances u_l, $i < l$, in the dialogue for emotion prediction. To simulate this, we employ a memory network (86) that maintains the information captured during the previous states. At each state t,[4] the memory network learns over the state $t-1$ memory content through a forward-GRU (mGRU) and updates it according to the current query q_t. The updated memory at state t is then utilized by the network in the emotion recognition for the utterance u_i (represented by the query q_t). Furthermore, it also acts as input for the next state $t+1$. We argue that the memory content accumulated over the emotion labels reveals the relationship among previous utterances, and the future utterances leverage it to establish their relationships with the previous utterances.

Here, we employ a masked interactive attention mechanism (109) to incorporate the information regarding the current query. For each state t, we compute the masked attention weights β_t considering input I_t ($I_t = \text{mGRU}(O_{t-1})$) and the query $q_t = \bar{h}_i$. Subsequently, the attended vector is computed for each hidden representation of input I_t. Since the masked attention weight β_t signifies the probability over the first t input hidden representations (i.e., $I_1, I_2, ..., I_t$) and $\sum \beta_t = 1$, we compute the attended vector for the first t hidden representations only and bypass the rest of the input representations (i.e., $I_{t+1}, ..., I_n$). The two sets of representations, i.e., t attended and $n-t$ bypassed, form the memory output O_t. The attended vectors, i.e., $O_t[1..t]$, are then utilized as the memory context at state t for the subsequent processing corresponding to the utterance u_i. We apply mean-pooling over $O_t[1..t]$ to compute $\bar{o}_i$, and concatenate it with the enriched speaker-dialogue hidden representation $\bar{h}_i$ for the final predictions, i.e., $E_i = Softmax(\bar{o}_i \oplus \bar{h}_i)$.

[4] The memory state t corresponds to the utterance u_i, $t+1$ to u_{i+1}, and so on.

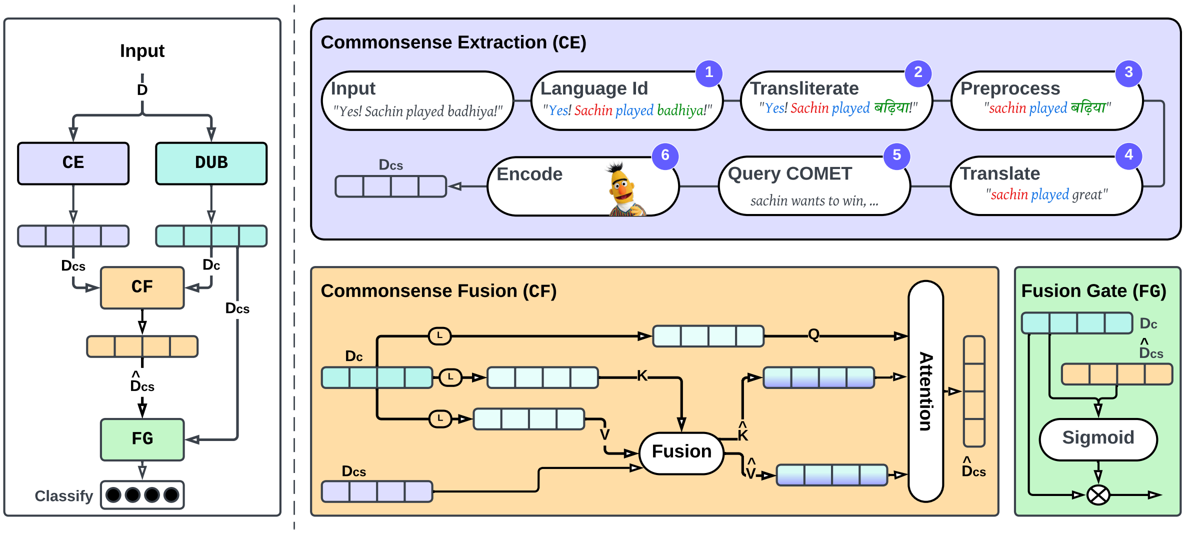


Figure 2.3: A schematic diagram of COFFEE. The Commonsense Extraction (CE) module takes a code-mixed input and provides a representation of the extracted commonsense information relevant to it. The commonsense information is fused with the representation obtained from a Dialogue Understanding Backbone (DUB) via the Commonsense Fusion (CF) and the Fusion Gate (FG) modules.

2.4.2 COFFEE: ERC for Code-mixed Dialogues

The manifestation of emotional concepts within an individual during a conversation is not solely influenced by the dialogue context, but also by the implicit knowledge accumulated through life experiences. This form of knowledge can be loosely referred to as commonsense. In light of this, we present an efficient yet straightforward methodology for extracting pertinent concepts from a given commonsense knowledge graph in the context of code-mixed inputs. Additionally, we introduce a clever strategy to seamlessly incorporate the commonsense features with the dialogue representation obtained from a backbone architecture dedicated to dialogue understanding. Figure 2.3 outlines our proposed approach, COFFEE while each of the intermediate modules is elucidated in detail below.

Dialogue Understanding Backbone (DUB). For input containing long contextual history, such as a dialogue, it becomes crucial to capture and comprehend the entire progression leading up to the present statement. Consequently, an effective dialogue understanding architecture which gives us a concrete dialogue representation is required. We use existing Transformer based architectures as our Dialogue Understanding Backbone, DUB. The given code-mixed dialogue D goes through DUB to give us the contextual dialogue representation, D_c. Specifically, $D_c = \texttt{DUB}(D)$, such that $D_c \in \mathbb{R}^{n \times d}$ where n is the maximum dialogue length, and d is the dimensionality of the resulting vectors.

Commonsense Extraction (CE). While the conversational context provides insights into the participants and the topic of the dialogue, the comprehension of implicit meanings within statements can be greatly facilitated by incorporating commonsense information. Therefore, in order to capture this valuable commonsense knowledge, we employ the COMET graph (89), which has been trained on ATOMIC triplets (95), to extract relevant commonsense information for each dialogue instance. However, it is worth noting that the COMET graph is pretrained using triplets in the English language, making it particularly effective for English inputs (87). Given that our input consists of a mixture of English and Hindi, we have devised a specialized knowledge extraction pipeline to tackle this challenge. The entire process of obtaining commonsense knowledge for a given code-mixed textual input is shown in Figure 2.3 and is comprehensively explained below.

1. *Language Identification:* To handle the input dialogue D, which includes a mix of English and Hindi words, the initial task is to determine the language of each word to appropriately handle different languages in the most suitable way.
2. *Transliteration:* The identified Hindi language words are transliterated to Devanagari script from roman script so that language-specific preprocessing can be applied to them.
3. *Text Processing:* The next step is to preprocess the text. This step involves converting text to lowercase and removal of non-ASCII characters and stopwords. The resultant text is considered important or *'topic specifying'* for the text.
4. *Translation:* Since COMET is trained for monolingual English, the query can only have English terms. Therefore, we translate the Devanagari Hindi *'topics'* back to romanised English.
5. *Querying COMET:* Finally, all the *'topics'* together are sent as a query to the COMET graph, and all possible relations are obtained.

COMET provides us with a vast array of effect-types corresponding to the input text. Specifically, it provides us with information such as *oEffect*, *oReact*, *oWant*, *xAttr*, *xEffect*, *xIntent*, *xNeed*, *xReact*, *xWant*. Refer Table 2.3 for the description of each of these values. We carefully select the relevant attributes from the extracted pairs and encode them using the BERT model (110). The representation obtained from BERT acts as our commonsense representation. Formally, $D_{cs} = CE(D)$, such that $D_{cs} \in \mathbb{R}^{m \times d}$ where m is the length of the commonsense information, and d is the vector dimension obtained from the BERT model. After we obtain the commonsense representation D_{cs}, we need to integrate it with the dialogue representation D_c. Consequently, we devise a sophisticated fusion mechanism as described below.

oEffect	The impact of input on the listeners.
oReact	The listeners' reaction to the input statement.
oWant	The listeners' desire after hearing the input.
xAttr	What the input reveals about the speaker.
xEffect	The speaker's desire after uttering the input.
xIntent	The speaker's objective in uttering the input.
xNeed	The speaker's needs according to the input.
xReact	The speaker's reaction based on the input.
xWant	The speaker's desire according to the input.

Table 2.3: Commonsense effect-types returned by the COMET and their description.

Commonsense Fusion (CF). Several studies discuss knowledge fusion, particularly in the context of multimodal fusion (111), where the most successful approaches often employ traditional dot-product-based cross-modal attention (112; 113). However, the traditional attention scheme results in the direct interaction of the fused information. As each fused information can be originated from a distinct embedding space, a direct fusion may be prone to noise and may not preserve maximum contextual information in the final representations. To address this, taking inspiration from context-aware attention (114), we propose to fuse commonsense knowledge using a *context-aware attention mechanism*. Specifically, we first generate commonsense conditioned key and value vectors and then perform a scaled dot-product attention using them. We elaborate on the process below.

Given the dialogue representation D_c obtained by a dialogue understanding backbone architecture, we calculate the query, key, and value vectors Q, K, and $V \in \mathbb{R}^{n \times d}$, respectively, as outlined in Equation 6.1 where W_Q, W_K, and $W_V \in \mathbb{R}^{d \times n}$ are learnable parameters, and n and d denote the maximum sequence length of the dialogue and dimensionality of the backbone architecture, respectively.

$$\begin{bmatrix} QKV \end{bmatrix} = D_c \begin{bmatrix} W_Q W_K W_V \end{bmatrix} \tag{2.1}$$

On the other hand, with the commonsense vector, D_{cs}, we generate commonsense infused key and value vectors $\hat{K}$ and $\hat{V}$, respectively as outlined in Equation 2.2, where U_k and $U_v \in \mathbb{R}^{d \times d}$ are learnable matrices. A scalar $\lambda \in \mathbb{R}^{n\times 1}$ is employed to regulate the extent of information to integrate from the commonsense knowledge and the amount of information to retain from the dialogue context. λ is a learnable parameter learnt using Equation 6.3, where $W_{k_1}, W_{k_2}, W_{v_1}$, and $W_{v_2} \in \mathbb{R}^{d\times 1}$ are trained along with the model.

$$\begin{bmatrix}\hat{K}\\ \hat{V}\end{bmatrix} = (1 - \begin{bmatrix}\lambda_k\\ \lambda_v\end{bmatrix}) \begin{bmatrix}K\\ V\end{bmatrix} + \begin{bmatrix}\lambda_k\\ \lambda_v\end{bmatrix} (D_{cs} \begin{bmatrix}U_k\\ U_v\end{bmatrix}) \tag{2.2}$$

$$\begin{bmatrix}\lambda_k\\ \lambda_v\end{bmatrix} = \sigma(\begin{bmatrix}K\\ V\end{bmatrix} \begin{bmatrix}W_{k_1}\\ W_{v_1}\end{bmatrix} + D_{cs} \begin{bmatrix}U_k\\ U_v\end{bmatrix} \begin{bmatrix}W_{k_2}\\ W_{v_2}\end{bmatrix}) \tag{2.3}$$

Finally, the commonsense knowledge infused vectors $\hat{K}$ and $\hat{V}$ are used to compute the traditional scaled dot-product attention.

$$\hat{D_c} = Softmax(\frac{Q\hat{K}^T}{\sqrt{d_k}})\hat{V} \tag{2.4}$$

Fusion Gating (FG). In order to control the extent of information transmitted from the commonsense knowledge and from the dialogue context, we use a Sigmoid gate. Specifically, $g = [D_c \oplus \hat{D_c}]W + b$. Here, $W \in \mathbb{R}^{2d\times d}$ and $b \in \mathbb{R}^{d\times 1}$ are trainable parameters, and $\oplus$ denotes concatenation. The final information fused representation $\hat{D_c}$ is given by $\hat{D_c} = D_c + g \odot \hat{D_c}$. $\hat{D_c}$ is used to identify the emotion class for the input dialogue.

2.5 Experiments and Results

2.5.1 Evaluating ERC-MMN

Baseline Methods

In this work, we employ the following set of baseline methods for a comparative study –

- **CMN** (20)**:** It utilizes memory networks to store the speaker-level contextual history within a dialogue. The authors showed that maintaining the conversational history in a memory helped CMN in predicting emotions more precisely. They also used these memories in capturing inter-speaker dependencies.
- **ICON** (115)**:** It maintains a memory network to preserve the interaction between the *self* and *inter-speaker* influences in dyadic conversations. It models this interaction into the global memory in a hierarchical way. Finally, the memory is used as a contextual summary which aid in predicting the emotional labels.
- **DGCN** (116)**:** It models the inter-speaker dynamics in a dialogue via a graph convolutional network. This work also leverages the self and inter-speaker dependencies of the participants for modeling conversations. By using graphs, the authors claim to have modeled context propagation in an efficient way.
- **AGHMN** (117)**:** It incorporates an attention GRU mechanism that controls the flow of information through a modified GRU cell based on the attention-weights, computed over the historical utterances in a dialogue.
- **Pointer Network** (118) **:** They are often used to generate output sequence when the length of

System	F1-score							
	Dg	Jy	Sr	An	Fr	Ne	Sa	W-Avg
CMN	0.0	48.6	**54.0**	33.7	8.6	**75.9**	19.9	51.7
ICON	0.0	36.8	45.5	37.0	0.0	69.6	11.0	50.1
DGCN	0.0	48.1	52.9	31.6	4.5	75.8	15.5	51.8
AGHMN	0.0	40.1	43.1	11.7	0.0	63.0	25.0	44.2
Pointer Network	3.0	15.1	17.0	13.1	0.0	63.2	7.0	35.1
ERC-MMN	**20.2**	**48.7**	50.4	42.9	9.8	71.9	29.6	**55.7**

Table 2.4: Comparative analysis for ERC. (Dg: disgust, Jy: joy, Sr: surprise, An: anger, Fr: fear, Ne: neutral, Sa: sadness). Performance is on the modified MELD dataset, which contains dialogues only if it has atleast one emotion flip..

output sequence depends on the length of the input sequence. Pointer networks have been applied to solve various combinatorial optimization and search problems such as Convex hull, and travelling salesman problem. Here, we use it to map our input sequence of utterances of a dialogue into a sequence of emotions.

These baselines are readily suitable for ERC, and a few of them reported their performance on the MELD dataset. However, we reproduced the results of these baselines on the modified MELD data where we discard any dialogues that contain no emotion flip for any speaker.

Results

We fine-tune BERT (110) for ERC and extract its last layer hidden representation as utterance representation. We keep the standard vector dimension of BERT to represent an utterance. All reported results are averaged over 5 runs. The MMN based system, i.e., ERC-MMN, obtain F1-scores of 55.78%. We utilize the publicly available implementations of the baselines for the comparative study and report the performance in Table 2.4. DGCN turns out to be the best baseline (51.80%), while CMN (51.70%) and ICON (50.15%) yield comparable performances. Pointer network seems to be the worst performing baseline (35.1%). In comparison, our proposed system, ERC-MMN reports the best performance with an improvement of $\sim 4\%$ against DGCN with 55.78% F1-score. nother critical observation is that due to class-imbalance, all baselines find it difficult to identify the *disgust* emotion. Similarly, three out of five baselines fail to classify any *fear* emotion as well. In contrast, our proposed models correctly identify both *disgust* and *fear* emotions for at least a few utterances. Furthermore, except for *surprise* and *neutral*, our proposed model outperforms the baselines in remaining five emotion labels. We believe this performance boost is the result of the use of memory network in an effective manner. As can be seen from the baseline results, AGHMN and Pointer Network do not perform at par with the others since they do not contain any memory component. While CMN, ICON and our method perform better.

Generalizability. To analyze the performance of our model on an out-of-distribution generalization test set, we consider another dataset, IEMOCAP (119). It contains crowdsourced conversations revolving around 16 topics. For the construction of our test set, we randomly pick two conversations from each topic. Table 2.5 gives us a brief statistics of the subset

ERC							
Disgust	Joy	Surprise	Anger	Fear	Neutral	Sadness	Total
0	671	44	1413	25	552	407	3112

Table 2.5: Statistics of the subset of IEMOCAP considered in this study.

of IEMOCAP dataset considered in this study. We test our model trained on MELD on IEMOCAP and report the results in Table 2.6. Our model produces the best results. However, the results are significantly less than the results obtained on MELD. This reduction can be attributed to the inherent differences in the dialogues present in the two datasets. IEMOCAP contains more than 50 utterances in a dialogue on average whereas MELD contains an average of 9 utterances per dialogue. Secondly, the emotion distribution between the two sets also differ in a major way. IEMOCAP does not contain any *disgust* emotion, and the *neutral* emotion is not as commonly present in it as it is in MELD. Consequently, the task of emotion recognition becomes challenging for this dataset.

System	**F1-score**							
	Dg	**Jy**	**Sr**	**An**	**Fr**	**Ne**	**Sa**	**W-Avg**
CMN	0.0	7.1	0.0	**56.4**	0.0	2.1	2.3	28.2
ICON	0.0	14.2	0.0	49.7	0.0	9.1	8.0	28.4
DGCN	0.0	15.5	2.3	54.2	0.0	6.0	11.5	30.8
AGHMN	0.0	7.3	0.0	45.8	0.0	0.0	11.2	24.4
Pointer Network	0.0	12.4	0.0	32.2	0.0	2.6	6.0	18.1
ERC-MMN	0.0	**19.3**	**3.2**	52.7	0.0	**10.2**	**17.1**	**33.7**

Table 2.6: Comparative analysis for ERC on the subset of IEMOCAP dataset. (Dg: disgust, Jy: joy, Sr: surprise, An: anger, Fr: fear, Ne: neutral, Sa: sadness). *Trained on MELD; Tested on IEMOCAP*.

Error Analysis

This section presents both quantitative and qualitative analysis *w.r.t.* the confusion matrix and misclassification examples. We also supplement our analysis of the proposed systems with DGCN (the best baseline).

Confusion Matrix. Table 2.7 shows the confusion matrix for ERC. For most of the emotions, our proposed EFR-MMN model reports better true-positives against the baseline, as highlighted in blue-colored text in the Table. The confusion matrix reveals the most confusing pair of emotions as *neutral* and *joy*, with 104 *joy* samples misclassified as *neutral* and 72 *neutral* samples misclassified as *joy*. Another important observation is that DGCN ignores the under-represented emotions, such as *disgust* and *fear*, completely. In contrast, our proposed model assigns these two emotions to a few utterances with a bit of success. It suggests EFR-MMN to be unbiased towards the under-represented emotion labels.

Qualitative Analysis. We also perform error analysis on the predictions of proposed systems. For illustration, we present one representative dialogue with its gold and predicted labels (ours and DGCN) for the ERC task. We observe from Table 2.8 that in a dialogue of ten utterances with three speakers, EFR-MMN misclassifies only one utterance, whereas DGCN misclassifies five utterances in the same dialogue. Moreover, DGCN predicts six utterances as *neutral*, out of which only four are correct. In comparison, our proposed model does not misclassify any emotion as *neutral* in the dialogue. It can be related to the biasness of DGCN towards the majority emotions.

		Predicted						
		Disgust	**Joy**	**Surprise**	**Anger**	**Fear**	**Neutral**	**Sadness**
Actual	**Disgust**	15/0	5/5	4/10	14/9	1/0	19/33	3/4
	Joy	13/0	157/110	12/32	26/19	4/0	104/154	9/10
	Surprise	7/0	31/40	115/87	32/21	4/0	44/80	5/10
	Anger	16/0	32/45	31/56	118/74	6/0	64/101	16/7
	Fear	1/0	4/5	5/9	7/2	4/0	14/24	7/3
	Neutral	28/0	72/54	42/44	50/19	14/0	705/808	32/18
	Sadness	7/0	18/25	9/12	19/17	6/0	68/100	42/15

Table 2.7: Confusion matrices for ERC. Cell (a/b) represents 'a' number of samples predicted by EFR-MMN (our best model) and 'b' number of samples predicted by DGCN. Blue-colored and red-colored texts represent superiority and inferiority of EFR-MMN, respectively, compared to the baseline, DGCN.

#	Speaker	Utterance	Actual	Prediction	
				ERC-MMN	DGCN
u_1	Phoebe	Well alright! We already tried feeding her, changing her, burping her. Oh! Try this one.	*sadness*	*anger*	*joy*
u_2	Phoebe	Go back in time and listen to Phoebe!	*anger*	*anger*	*joy*
u_3	Monica	Alright here's something. It says to try holding the baby close to your body and then swing her rapidly from side to side.	*neutral*	*neutral*	*neutral*
u_4	Rachel	Ok.	*neutral*	*neutral*	*neutral*
u_5	Monica	It worked!	*surprise*	*surprise*	*anger*
u_6	Rachel	Oh! Oh! No, just stopped to throw up a little bit.	*sadness*	*sadness*	*neutral*
u_7	Rachel	Oh come on! What am I gonna do? Its been hours and it won't stop crying!	*sadness*	*sadness*	*neutral*
u_8	Monica	Umm 'she' Rach not 'it', 'she'.	*neutral*	*neutral*	*neutral*
u_9	Rachel	Yeah I'm not so sure.	*neutral*	*neutral*	*neutral*
u_{10}	Monica	Oh my god! I am losing my mind!	*anger*	*anger*	*anger*

Table 2.8: Actual and predicted emotions for a dialogue having 10 utterances (u_1, ..., u_{10}) from the test set. Red-colored text represents misclassification. For the given example, ERC-MMN misclassifies only one utterance, whereas DGCN (best baseline) commits mistakes for 5 out of 10 utterances.

2.5.2 Evaluating COFFEE

Dialogue Understanding Backbone

Existing approaches for ERC predominantly concentrate on the English language. Nonetheless, we incorporate two state-of-the-art techniques for ERC using English datasets and leverage four established Transformer-based methodologies as our foundation systems to address the ERC task.

- **BERT** (110) is a pre-trained language model that utilizes a Transformer architecture and bidirectional context to understand the meaning and relationships of words in a sentence.
- **RoBERTa** (120) is an extension of BERT that improves its performance utilizing additional training techniques such as dynamic masking, longer sequences, and more iterations.
- **mBERT** [5] (multilingual BERT) is a variant of BERT that is trained on a multilingual corpus, enabling it to understand and process text in multiple languages.
- **MURIL** (121) (Multilingual Representations for Indian Languages) is a variant of BERT specifically designed to handle Indian languages.

[5] `https://huggingface.co/M-CLIP/M-BERT-Base-ViT-B`

- **CoMPM** (122) is a Transformer-based architecture especially curated for ERC. It extracts pre-trained memory as an extractor of external information from the pre-trained language model and combines it with the context model.
- **DialogXL** (50) addresses multi-party structures by utilizing increased memory to preserve longer historical context and dialogue-aware self-attention. It alters XLNet's recurrence method from segment to utterance level to better represent conversational data.
- **KET** (123) or the Knowledge-Enriched Transformer deciphers contextual statements by employing hierarchical self-attention, while simultaneously harnessing external common knowledge through an adaptable context-sensitive affective graph attention mechanism.
- **COSMIC** (91) is a framework that integrates various aspects of common knowledge, including mental states, events, and causal connections, and uses them as a foundation for understanding how participants in a conversation interact with one another.

Although BERT and RoBERTa are trained using monolingual English corpus, we use them for romanised code-mixed input, anticipating that finetuning will help the models grasp Hindi-specific nuances. To ensure a fair comparison, we also include multilingual models such as mBERT and MURIL in our analysis. Additionally, since we are dealing with the task of ERC, we consider two state-of-the-art baseline architectures in this domain for monolingual English dialogues, namely CoMPM and DialogXL and two state-of-the-art baseline that incorporates commonsense for ERC – KET, and COSMIC.

Experiment Setup and Evaluation Metric

The COMET graph gives us multiple attributes for one input text (c.f. Table 2.3). However, not all of them contributes towards the emotion elicited in the speaker. Consequently, we examine the correlation between the extracted commonsense attributes with emotion labels in our train instances. We use BERT to obtain representation for each commonsense attribute and find out their correlation with the emotion labels. We show this correlation in Figure 2.4. As can be seen, *'xWant'* is most positively correlated with the emotion labels, and *'oReact'* is most negatively correlated. Consequently, we select the attributes *'xWant'*, and *'oReact'* as commonsense. Further, for evaluating the performance, we select weighted F1 score as our metric of choice to handle the imbalanced class distribution of emotions present in our dataset.

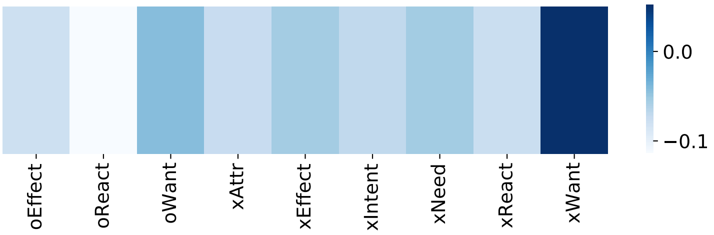


Figure 2.4: Correlation between different commonsense attributes with the emotion attribute.

Results

Table 6.14 illustrates the results (F1-scores) we obtain for the task of ERC with and without using COFFEE to incorporate commonsense knowledge. Notably, in the absence of commonsense, RoBERTa and DialogXL outperform the other systems. However, it is intriguing to observe that mBERT and MURIL, despite being trained on multilingual data, do not surpass the performance of BERT, RoBERTa, or DialogXL. Further, when commonsense is included as part of the input using the COFFEE approach, all systems exhibit improved performance. The F1 scores corresponding to individual emotions show a proportional relationship with the quantity of data samples available for each specific emotion, as anticipated within a deep learning architecture. The *neutral* emotion achieves the highest performance,

	Model	Anger	Contempt	Disgust	Fear	Joy	neutral	Sadness	Surprise	Weighted F1
Standard	**BERT**	0.23	0.18	0.11	**0.20**	0.45	0.54	0.16	0.32	0.40
	RoBERTa	0.26	0.21	0.16	0.06	0.47	0.57	0.12	0.34	0.41
	mBERT	0.10	0.11	0.00	0.11	0.23	0.50	0.13	0.08	0.30
	MURIL	0.24	0.22	0.07	0.00	0.42	0.51	0.06	0.23	0.35
	CoMPM	0.10	0.12	0.00	0.00	0.44	0.57	0.02	0.00	0.35
	DialogXL	0.25	0.09	0.07	0.17	0.43	0.59	0.17	0.28	0.41
COFFEE	**BERT**	0.24 (↑0.01)	0.2 (↑0.02)	0.12 (↑0.01)	0.19 (↓0.01)	0.46 (↑0.01)	0.56 (↑0.02)	0.18 (↑0.02)	**0.35** (↑0.03)	0.41 (↑0.01)
	RoBERTa	**0.29** (↑0.03)	**0.24** (↑0.03)	**0.18** (↑0.02)	0.10 (↑0.04)	**0.49** (↑0.02)	**0.61** (↑0.04)	**0.18** (↑0.06)	0.34 (↑ 0.00)	**0.44** (↑0.03)
	mBERT	0.11 (↑0.01)	0.13 (↑0.02)	0.04 (↑0.04)	0.12 (↑0.01)	0.24 (↑0.01)	0.51 (↑0.01)	0.12 (↓0.01)	0.10 (↑0.02)	0.31 (↑0.01)
	MURIL	0.26 (↑0.02)	0.21 (↓0.01)	0.10 (↑0.03)	0.01 (↑0.01)	0.46 (↑0.04)	0.52 (↑0.01)	0.08 (↑0.02)	0.22 (↓0.01)	0.37 (↑0.02)
	CoMPM	0.11 (↑0.01)	0.14 (↑0.02)	0.02 (↑0.02)	0.02 (↑0.02)	0.45 (↑0.01)	0.56 (↓0.01)	0.03 (↑0.01)	0.10 (↑0.01)	0.36 (↑0.01)
	DialogXL	0.26 (↑0.01)	0.11 (↑0.02)	0.10 (↑0.03)	0.19 (↑0.02)	0.44 (↑0.01)	0.59 (↑ 0.00)	0.20 (↑0.03)	0.31 (↑0.03)	0.42 (↑0.01)
CS	**KET**	0.14 (↓0.15)	0.11 (↓0.13)	0.09 (↓0.09)	0 (↓0.10)	0.34 (↓0.15)	0.41 (↓0.20)	0.08 (↓0.10)	0.19 (↓0.15)	0.28 (↓0.16)
	COSMIC	0.21 (↓0.08)	0.18 (↓0.06)	0.15 (↓0.03)	0.03 (↓0.07)	0.39 (↓0.10)	0.49 (↓0.12)	0.13 (↓0.05)	0.27 (↓0.07)	0.34 (↓0.10)

Table 2.9: Performance of comparative systems with and without incorporating commonsense via COFFEE. Numbers in parenthesis indicate the corresponding performance gain over the non-commonsense (standard) version. The last two rows compare the performance of the best performing COFFEE model (RoBERTa) with other commonsense (CS) based ERC methods.

followed by *joy* and *surprise*, as these classes possess a greater number of data samples (see Table 2.2). Conversely, the minority classes such as *contempt* and *disgust* consistently obtain the lowest scores across almost all systems. Furthermore, we can observe from the table that the existing strategies of commonsense fusion perform poorly when compared with the COFFEE method. The loss in performance can be attributed to two aspects of the comparative system – KET uses NRC_VAD (124), which is an English-based lexicon containing VAD scores, i.e., valence, arousal, and dominance scores, to gather words for which knowledge is to be retrieved. Since our input is code-mixed with the matrix language as Hindi, using only the English terms makes the KET approach ineffective. In contrast, although COSMIC uses the COMET graph, it uses the raw representations obtained from the commonsense graph and concatenates them with the utterance representations obtained from the GRU architecture. Since we use the generated natural language commonsense with the smart fusion method, we hypothesize that our model is able to capture and utilize this knowledge effectively. Additionally, we performe a T-test on our results to check the statistical significance of our performance gain and obtained a p-value of 0.0321 for our RoBERTa model which, being less than 0.05, makes our results statistically significant.

Ablation Study

Fusion Methods. We investigate the effectiveness of COFFEE in capturing and incorporating commonsense information. To evaluate different mechanisms for integrating this knowledge into the dialogue context, we present the results in Table 6.7. Initially, we explore a straightforward method of concatenating the obtained commonsense knowledge with the dialogue context and passing it through the RoBERTa model. Interestingly, this simple concatenation leads to a decline in the performance of emotion recognition, suggesting that the introduced commonsense information may act as noise in certain cases. This outcome can be attributed to the inherent nature of some utterances, where external knowledge may not be necessary to accurately determine the expressed emotion. For instance, consider the sentence “Aaj me sad hun” (*“I am sad today”*), which can be comprehended without relying on commonsense information to identify the emotion as sadness. In such scenarios, enforcing additional information may disrupt the model’s behavior, resulting in suboptimal performance. Conversely, by allowing the model the flexibility to decide when and to what extent to incorporate commonsense knowledge, as demonstrated by the attention and COFFEE approaches, we observe an improvement in system performance, with COFFEE yielding the most favorable outcomes.

	RoBERTa	Anger	Contempt	Disgust	Fear	Joy	neutral	Sadness	Surprise	Weighted F1
	Standard	0.26	0.21	0.16	0.06	0.47	0.57	0.12	**0.34**	0.41
CS	**Concat**	0.22 (↓0.04)	0.19 (↓0.02)	0.15 (↓0.01)	0.04 (↓0.02)	0.44 (↓0.03)	0.52 (↓0.05)	0.09 (↓0.03)	0.31 (↓0.03)	0.37 (↓0.04)
	DPA	0.27 (↑0.01)	0.21 (↑ 0.00)	0.16 (↑ 0.00)	0.08 (↑0.02)	0.48 (↑0.01)	0.59 (↑0.02)	0.11 (↓0.01)	0.33 (↓0.01)	0.42 (↑0.01)
	COFFEE$_{Eng}$	0.11 (↓0.15)	0.09 (↓0.12)	0.01 (↓0.15)	0 (↓0.06)	0.16 (↓0.31)	0.24 (↓0.33)	0.02 (↓0.10)	0.11 (↓0.23)	0.16 (↓0.25)
	COFFEE$_{Hin}$	0.20 (↓0.06)	0.15 (↓0.06)	0.12 (↓0.04)	0.02 (↓0.04)	0.36 (↓0.11)	0.53 (↓0.04)	0.12 (↑0.00)	0.29 (↓0.05)	0.35 (↓0.06)
	COFFEE$_{xW}$	0.26 (↑ 0.00)	0.22 (↑0.01)	0.15 (↓0.01)	0.04 (↑0.02)	0.47 (↑ 0.00)	0.59 (↑ 0.02)	0.16 (↑0.04)	0.33 (↓0.01)	0.42 (↑0.01)
	COFFEE$_{oR}$	0.27 (↑0.01)	**0.24** (↑0.03)	0.17 (↑0.01)	0.07 (↑0.01)	0.43 (↓0.04)	0.59 (↑0.02)	**0.18** (↑0.06)	0.33 (↓0.01)	0.41 (↑ 0.00)
	COFFEE	**0.29** (↑0.03)	**0.24** (↑0.03)	**0.18** (↑0.02)	**0.10** (↑0.04)	**0.49** (↑0.02)	**0.61** (↑0.04)	**0.18** (↑0.06)	**0.34** (↑ 0.00)	**0.44** (↑0.03)

Table 2.10: Ablation results comparing different fusion techniques for the best performing system (RoBERTa). Numbers in parenthesis indicate the performance gain over the non-commonsense (standard) version. Performance when only one of the matrix or embedding language is used for experimentation is also shown. (CS: Commonsense; DPA: Dot Product Attention; COFFEE$_{xW}$: COFFEE with only *xWant* attribute as commonsense knowledge; COFFEE$_{oR}$: COFFEE with only *oReact* attribute as commonsense knowledge).

Effect of Language. In code-mixing, the input amalgamates two or more languages, often with one language being the dominant one, called the matrix language, while others act as embedding languages. The foundation of grammatical structure comes from the matrix language (Hindi in our case), and solely relying on the embedding language (English in our case) can lead to a decline in the model's performance. On the flip side, the embedding language plays a vital role in capturing accurate contextual details within the input. Therefore, confining ourselves to only the matrix language should also result in a drop in performance. To verify this hypothesis, the third and fourth row of Table 6.7 shows the performance of the COFFEE methodology, using the RoBERTa model, when we use only English (the embedding language) and only Hindi (the matrix language) in our input. The results reinforce our hypothesis, where the usage of only embedding language (English only) deteriorates the model performance extensively, while the sole use of matrix language (Hindi only) also hampers the performance when compared to the system that uses both the languages.

oEffect	***oReact***	***oWant***	***xAttr***	***xEffect***	***xIntent***	***xNeed***	***xReact***	***xWant***
0.32	0.41	0.39	0.34	0.37	0.37	0.32	0.36	**0.42**

Table 2.11: Ablation results comparing different attibutes of commonsense when fused with RoBERTa using the COFFEE. The scores are weighted F1.

COMET Attributes. We explore the utilization of various COMET attributes as our commonsense information. The last three rows in Table 6.7 demonstrate the outcomes when we integrate the two most correlated attributes, *xWant* and *oReact* with the RoBERTa backbone model using COFFEE. It is evident that the individual consideration of these attributes does not significantly enhance the performance of ERC compared to when they are combined. Additionally, Table 2.11 presents the weighted F1 scores achieved by the RoBERTa model when each commonsense attribute is incorporated individually using COFFEE. These results align well with the observed correlation between the attributes and the corresponding emotion labels in Figure 2.4.

Error Analysis

A thorough quantitative analysis, detailed in the previous section, revealed that the integration of commonsense knowledge enhances the performance of all systems under examination. However, to gain a deeper understanding of the underlying reasons for this improvement, we conduct a comprehensive error analysis, comprising of confusion matrices and subjective evaluations.

#	Speaker	Utterance	Emotion		
			Gold	**w/o CS**	**w CS**
u1	Maya	Khatam ho gaya Sahil it's over! *(It's over, Sahil!)*	sadness	sadness	sadness
u2	Monisha	Mummyji, tissue paper ke aur 2 boxes hai, laati hun. *(Mummyji, I have two more boxes of tissue paper, I'll bring them.)*	neutral	joy	neutral
u3	Sahil	Monisha, mom tissue paper ki baat nhi kar rhi hai... *(Monisha, mom is not talking about tissue paper...)*	neutral	sadness	neutral
u4	Maya	My life! Meri zindagi! Khatam ho gayi hai. Can you imagine Sahil? Uss Rita se toh Monisha zyada achi hai. Can you imagine? *(My life! My life! It's over. Can you imagine, Sahil? Monisha is better than that Rita. Can you imagine?)*	sadness	sadness	sadness
u5	Monisha	Mein kya itni buri hun mummy ji? *(Am I that bad, mummyji?)*	sadness	sadness	contempt
u6	Maya	Haan beta. Lekin wo Rita! Oh my god! Saans leti hai toh bhi cheekh sunai deti hai. Jab logo ko pata chalega ke rosesh ne loudspeaker se shaadi ki hai?! *(Yes, dear. But that Rita! Oh my god! Even when she breathes, she makes a sound. When people find out that Rosesh got married through a loudspeaker?!)*	sadness	disgust	sadness
u7	Monisha	Logo ko pata chal gaya mummyji... *(People found out, mummyji...)*	neutral	surprise	surprise
u8	Maya	What do you mean?! *(What do you mean?!)*	fear	fear	contempt
u9	Monisha	Wo sarita aunty ka phone aaya tha na... *(Sarita aunty called, right...)*	neutral	surprise	neutral

Table 2.13: Actual and predicted emotions (using RoBERTa) for a dialogue having nine utterances from the test set of E-MaSaC. Red-colored text represents misclassification.

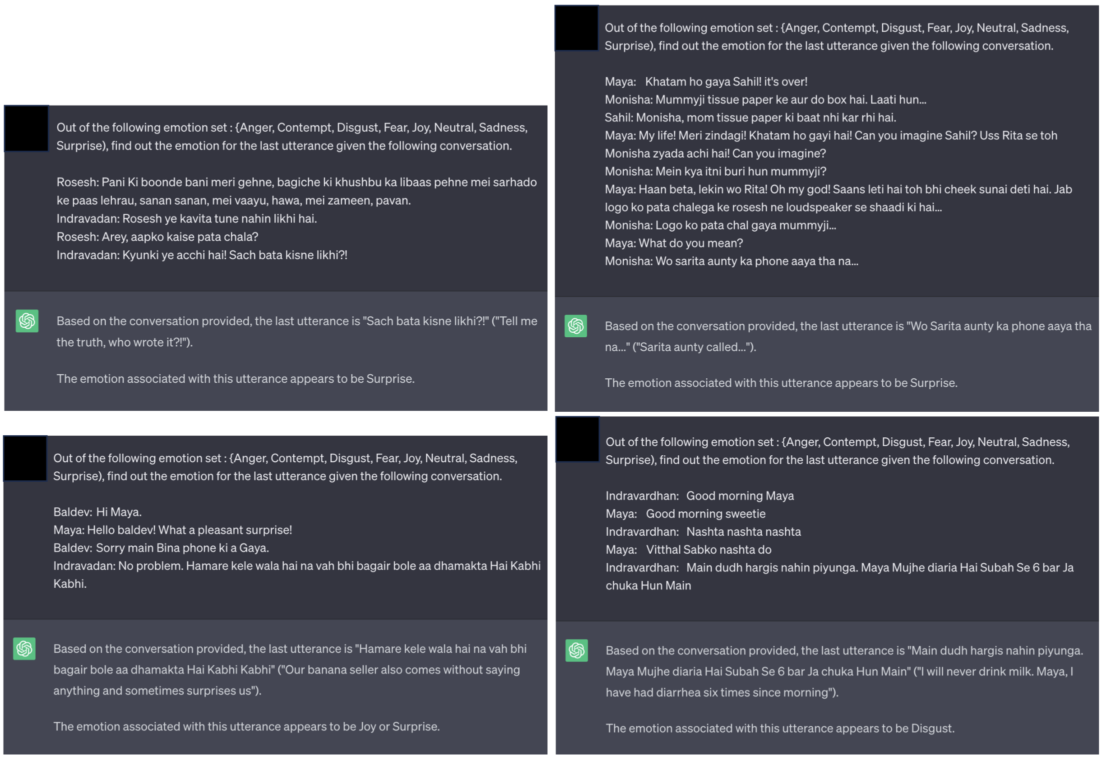
Confusion Matrix. Given the superior performance of the RoBERTa model we conduct an examination of its confusion matrices with and without commonsense fusion, as shown in Table 2.12. We observe that the RoBERTa model with COFFEE integration achieves higher true positives for most emotions. However, it also exhibits a relatively higher number of false negatives when compared with its standard variant, particularly for the *neutral* class. This observation suggests that the commonsense-infused model excels in recall but introduces some challenges in terms of precision, thereby presenting an intriguing avenue for future research. Additionally, we notice a heightened level of confusion between *neutral* and *joy* emotions, primarily due to their prevalence in the dataset. Both models, however, demonstrate the least confusion between the *disgust* and *surprise* emotions, indicating their distinguishable characteristics.

		Predicted							
		An	**Co**	**Di**	**Fe**	**Jo**	**Ne**	**Sa**	**Su**
Gold	**An**	**26/33**	5/5	8/4	2/9	13/20	86/62	1/4	1/5
	Co	6/4	**9/16**	6/6	1/2	7/8	50/42	2/2	1/2
	Di	2/6	1/3	**5/2**	0/0	1/1	7/4	0/0	1/1
	Fe	9/13	2/5	3/4	**1/3**	17/19	76/55	12/17	2/6
	Jo	3/8	3/4	2/2	1/9	**159/171**	162/139	15/8	4/8
	Ne	21/32	14/18	2/7	2/9	76/81	**496/450**	21/27	24/32
	Sa	5/11	7/6	5/3	0/1	19/26	90/71	**28/35**	1/2
	Su	1/1	0/0	0/0	0/0	8/6	22/18	0/1	**26/31**

Table 2.12: Confusion matrices for ERC for the best performing RoBERTa model (without/with commonsense). (An: Anger; Co: Contempt; Di: Disgust; Fe: Fear; Jo: Joy; Ne: Neutral; Sa: Sadness; Su: Surprise).

Subjective Evaluation

For the purpose of illustration, we select a single instance from the test set of E-MaSaC and present, with it, the ground-truth and the predicted labels for the task of ERC for the best performing RoBERTa model with and without using the COFFEE approach in Table 2.13. It can be observed that the inclusion of commonsense knowledge in the model significantly reduces errors. Comparatively, the variant of RoBERTa that does not incorporate commonsense knowledge makes errors in 5 out of 9 instances, whereas the variant utilizing commonsense knowledge, using COFFEE, misclassifies only 3 utterances. Within the test set, numerous similar instances exist where the commonsense-infused variant outperforms its counterpart due to the implicit information embedded in the utterances.



[redacted]

Out of the following emotion set : {Anger, Contempt, Disgust, Fear, Joy, Neutral, Sadness, Surprise), find out the emotion for the last utterance given the following conversation.

Rosesh: Pani Ki boonde bani meri gehne, bagiche ki khushbu ka libaas pehne mei sarhado ke paas lehrau, sanan sanan, mei vaayu, hawa, mei zameen, pavan.
Indravadan: Rosesh ye kavita tune nahin likhi hai.
Rosesh: Arey, aapko kaise pata chala?
Indravadan: Kyunki ye acchi hai! Sach bata kisne likhi?!

Based on the conversation provided, the last utterance is "Sach bata kisne likhi?!" ("Tell me the truth, who wrote it?!").

The emotion associated with this utterance appears to be Surprise.

[redacted]

Out of the following emotion set : {Anger, Contempt, Disgust, Fear, Joy, Neutral, Sadness, Surprise), find out the emotion for the last utterance given the following conversation.

Maya: Khatam ho gaya Sahil! it's over!
Monisha: Mummyji tissue paper ke aur do box hai. Laati hun...
Sahil: Monisha, mom tissue paper ki baat nhi kar rhi hai.
Maya: My life! Meri zindagi! Khatam ho gayi hai! Can you imagine Sahil? Uss Rita se toh Monisha zyada achi hai! Can you imagine?
Monisha: Mein kya itni buri hun mummyji?
Maya: Haan beta, lekin wo Rita! Oh my god! Saans leti hai toh bhi cheek sunai deti hai. Jab logo ko pata chalega ke rosesh ne loudspeaker se shaadi ki hai...
Monisha: Logo ko pata chal gaya mummyji...
Maya: What do you mean?
Monisha: Wo sarita aunty ka phone aaya tha na...

Based on the conversation provided, the last utterance is "Wo Sarita aunty ka phone aaya tha na..." ("Sarita aunty called...").

The emotion associated with this utterance appears to be Surprise.

[redacted]

Out of the following emotion set : {Anger, Contempt, Disgust, Fear, Joy, Neutral, Sadness, Surprise), find out the emotion for the last utterance given the following conversation.

Baldev: Hi Maya.
Maya: Hello baldev! What a pleasant surprise!
Baldev: Sorry main Bina phone ki a Gaya.
Indravadan: No problem. Hamare kele wala hai na vah bhi bagair bole aa dhamakta Hai Kabhi Kabhi.

Based on the conversation provided, the last utterance is "Hamare kele wala hai na vah bhi bagair bole aa dhamakta Hai Kabhi Kabhi" ("Our banana seller also comes without saying anything and sometimes surprises us").

The emotion associated with this utterance appears to be Joy or Surprise.

[redacted]

Out of the following emotion set : {Anger, Contempt, Disgust, Fear, Joy, Neutral, Sadness, Surprise), find out the emotion for the last utterance given the following conversation.

Indravardhan: Good morning Maya
Maya: Good morning sweetie
Indravardhan: Nashta nashta nashta
Maya: Vitthal Sabko nashta do
Indravardhan: Main dudh hargis nahin piyunga. Maya Mujhe diaria Hai Subah Se 6 bar Ja chuka Hun Main

Based on the conversation provided, the last utterance is "Main dudh hargis nahin piyunga. Maya Mujhe diaria Hai Subah Se 6 bar Ja chuka Hun Main" ("I will never drink milk. Maya, I have had diarrhea six times since morning").

The emotion associated with this utterance appears to be Disgust.

Figure 2.5: Screenshots of ChatGPT responses when prompted to provide emotions for the last utterance in the dialogue.

ChatGPT and Code-mixing

Considering the emergence and popularity of ChatGPT, it becomes imperative to conduct an analysis of it for the task of ERC in code-mixed dialogues. Although ChatGPT exhibits remarkable performance in a zero-shot setting across various tasks and scenarios, it is important to note its shortcomings, particularly when dealing with code-mixed input. To evaluate its performance, we extract instances from E-MaSaC and engage ChatGPT in identifying the emotions evoked within the dialogues. To accomplish this, we construct a prompt that includes a potential set of emotions along with the code-mixed dialogue as input. Specifically, the prompt used is as follows:

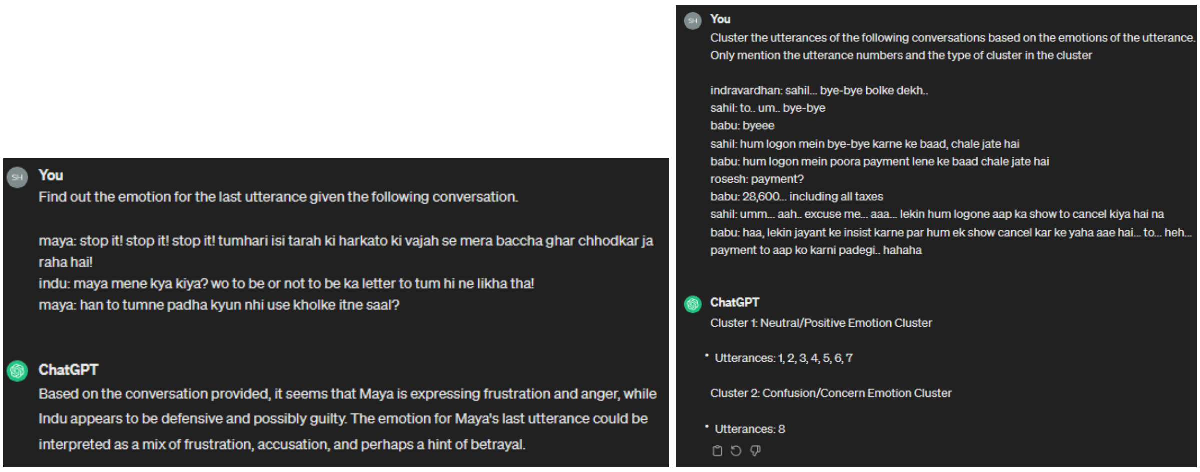
“Out of the following emotion set : {Anger, Contempt, Disgust, Fear, Joy, Neutral, Sadness, Surprise}, find out the emotion for the last utterance given the following conversation. <Conv>"

Although ChatGPT demonstrated proficiency in discerning emotions within concise and uncomplicated conversations, its performance waned when faced with the challenge of identifying the accurate emotion as the dialogue context extended, occasionally encompassing more than three utterances. Figure 2.5 shows four such instances. Moreover, as ChatGPT primarily operates based on prompts, we conducted experiments by employing varying prompts and querying the model to determine emotions. For instance, we evaluated ChatGPT without restricting it to a predefined set of emotions. The prompt used for this evaluation was:

“Find out the emotion for the last utterance given the following conversation. <Conv>"

In another assessment, we tasked the model with categorizing the utterances of a conversation based on the similarity of emotions displayed within it. The prompt employed in this evaluation was: *“Cluster the utterances of the following conversations based on the emotions of the utterance. Only mention the utterance numbers and the type of cluster in the cluster. <Conv>"*

Figure 2.6 depicts the responses generated by ChatGPT in response to the aforementioned prompts. It is observed that while we analyze discrete emotions where a single utterance may convey a single emotion, ChatGPT tends to attribute a mixture of emotions to a single utterance. Additionally, ChatGPT appears to struggle in recognizing sarcasm in the utterance "hum logon mein bye-bye karne ke baad, chale jate hai. *(After saying bye-bye to each other, we leave.)*", erroneously identifying it as a positive emotion utterance.



You
Find out the emotion for the last utterance given the following conversation.

maya: stop it! stop it! stop it! tumhari isi tarah ki harkato ki vajah se mera baccha ghar chhodkar ja raha hai!
indu: maya mene kya kiya? wo to be or not to be ka letter to tum hi ne likha tha!
maya: han to tumne padha kyun nhi use kholke itne saal?

ChatGPT
Based on the conversation provided, it seems that Maya is expressing frustration and anger, while Indu appears to be defensive and possibly guilty. The emotion for Maya's last utterance could be interpreted as a mix of frustration, accusation, and perhaps a hint of betrayal.

You
Cluster the utterances of the following conversations based on the emotions of the utterance. Only mention the utterance numbers and the type of cluster in the cluster

indravardhan: sahil... bye-bye bolke dekh..
sahil: to.. um.. bye-bye
babu: byeee
sahil: hum logon mein bye-bye karne ke baad, chale jate hai
babu: hum logon mein poora payment lene ke baad chale jate hai
rosesh: payment?
babu: 28,600... including all taxes
sahil: umm... aah.. excuse me... aaa... lekin hum logone aap ka show to cancel kiya hai na
babu: haa, lekin jayant ke insist karne par hum ek show cancel kar ke yaha aae hai... to... heh... payment to aap ko karni padegi.. hahaha

ChatGPT
Cluster 1: Neutral/Positive Emotion Cluster

- Utterances: 1, 2, 3, 4, 5, 6, 7

Cluster 2: Confusion/Concern Emotion Cluster

- Utterances: 8

Figure 2.6: Screenshots of ChatGPT responses when provided with varied prompts based on emotion analysis.

2.5.3 Evaluating LLMs

In recent years, there has been a significant surge in the emergence of both open-source and proprietary Large Language Models (LLMs) such as Llama (125). To assess these models' efficacy on the tasks and datasets outlined in this chapter, we evaluate the Llama model and compare its performance against our top-performing systems. Table 2.14 highlights the F1-scores obtained by these models. Our findings reveal that while the larger language model excels in English dialogues, its performance in Hindi-English code-mixed dialogues is comparable to our system when considering weighted F1 scores. This suggests that while Llama demonstrates proficiency in processing English text, there remains room for enhancement in its comprehension of multilingual text.

Dataset	Model	Anger	Contempt	Disgust	Fear	Joy	Neutral	Sadness	Surprise	Wtd. F1
MELD	ERC-MMN	0.42	-	0.20	0.09	0.48	0.71	0.29	**0.50**	0.55
	Llama	**0.45**	-	**0.27**	**0.12**	**0.51**	**0.75**	**0.33**	0.49	**0.58**
E-MaSaC	COFFEE	0.29	**0.24**	0.18	0.10	0.49	**0.61**	0.18	0.34	**0.44**
	Llama	**0.30**	0.21	**0.20**	**0.11**	**0.51**	0.59	**0.21**	**0.35**	**0.44**

Table 2.14: Performance of Llama when compared with our proposed methodologies for the ERC task.

2.6 Conclusion

This chapter delves into the multifaceted domain of emotion recognition in conversation, encompassing both monolingual English and the intricacies of Hindi-English code-mixed dialogue. In the context of monolingual English, our research introduces a sophisticated masked memory network. This network adeptly captures the subtleties inherent in individual speakers' emotions and the overarching emotional dynamics within a conversation. To put this system to the test, we turn to the well-regarded MELD dataset as our benchmark. In parallel, when we shift our focus to the code-mixed setting, we take the initiative to curate an emotion-laden dialogue dataset tailored for Hindi-English conversations. Our methodology introduces an innovative approach that leverages existing knowledge graphs, specifically designed to extract salient commonsense concepts relevant to code-mixed inputs. This extracted commonsense knowledge is seamlessly integrated into the core architecture through a groundbreaking fusion technique, employing a context-aware attention mechanism. Our rigorous findings unequivocally demonstrate that the incorporation of these commonsense features yields a substantial enhancement in ERC performance, substantiated through both quantitative metrics and in-depth qualitative analyses. Nevertheless, our exploration does not conclude with the mere identification of emotions; it extends further into the intricate terrain of understanding how emotions evolve within a dialogue. This nuanced endeavor seeks to recognize the transitions in emotions, paving the way for more adaptive and contextually relevant dialogue modifications. This pioneering task, known as Emotion Flip Reasoning (EFR), takes center stage in the forthcoming chapter, amplifying our emotion analysis capabilities and offering a deeper insight into the evolving dynamics of conversational emotions.

3. Emotion Flip Reasoning

In a conversational dialogue, speakers may have different emotional states and their dynamics play an important role in understanding dialogue's emotional discourse. However, simply detecting emotions is not sufficient to entirely comprehend the speaker-specific changes in emotion that occur during a conversation. To understand the emotional dynamics of speakers in an efficient manner, it is imperative to identify the triggers and instigators behind any changes or flips in emotion expressed by the speaker. In this chapter, we propose a new task called Emotion Flip Reasoning (EFR), which aims to identify the triggers and instigator behind a speaker's emotion flip within a conversation. For example, an emotion flip from *joy* to *anger* could be caused by an instigator like *threat*. To facilitate this task, we present MELD-FR and MELD-I, two datasets that includes ground-truth EFR trigger and instigator labels, which are in line with emotional psychology. To evaluate the dataset, we propose novel neural architectures called EFR-TX and TGIF, which leverage Transformer encoders and stacked GRUs to capture the dialogue context, speaker dynamics, and emotion sequence in a conversation. Our evaluation demonstrates state-of-the-art performance against five baselines used for the task. Further, we establish the generalizability of EFR-TX and TGIF on unseen datasets in a zero-shot setting. Additionally, we provide a detailed analysis of the competing models, highlighting the advantages and limitations of our neural architectures.

3.1 Introduction

Understanding emotions is essential for assessing the current state of a speaker in a conversation. Consequently, there has been a significant amount of research in this field (126). Emotional awareness has proven beneficial in areas that involve aspect analysis of users such as social media (127; 128; 129), and e-commerce (130). Initial studies focused on standalone texts like tweets (127; 128) to extract the appropriate emotions. However, with the advent of online dialogue agents, the focus of emotion analysis has shifted towards conversation data, usually termed as Emotion Recognition in Conversation (ERC) (131). Here, the input is a sequence of utterances or a dialogue, instead of isolated texts, and the aim is to identify the emotion of each dialogue utterance. Though emotion is an imperative aspect of a conversation, we posit that it is insufficient to simply identify the speakers' emotion in a dialogue. To reason out the change/flip in emotion of a speaker, a more detailed analysis is required. To this end, we propose a new task – **Emotion Flip Reasoning** *aka* **EFR**.

Exploring reasons behind emotion-flips of a speaker has wide variety of applications. For example, a dialogue agent can utilize this as feedback for the response generation, as and when, it senses an emotion-flip due to one of its generated responses. A positive emotional-flip (e.g., *sadness* → *joy*) can be treated as a reward, whereas the system can penalize the agent for a negative emotional-flip (e.g., *neutral* → *angry*). Other than empathetic response generation, another possible application of identifying the triggers for an emotion-flip is in the domain of affect monitoring. An organization or an individual can reason upon the emotion-flip in a conversation and make an informed decision for a downstream task.

EFR deals with identifying the cause/reason behind an emotional flip of a speaker in a dialogue. The entire EFR pipeline works in three stages-

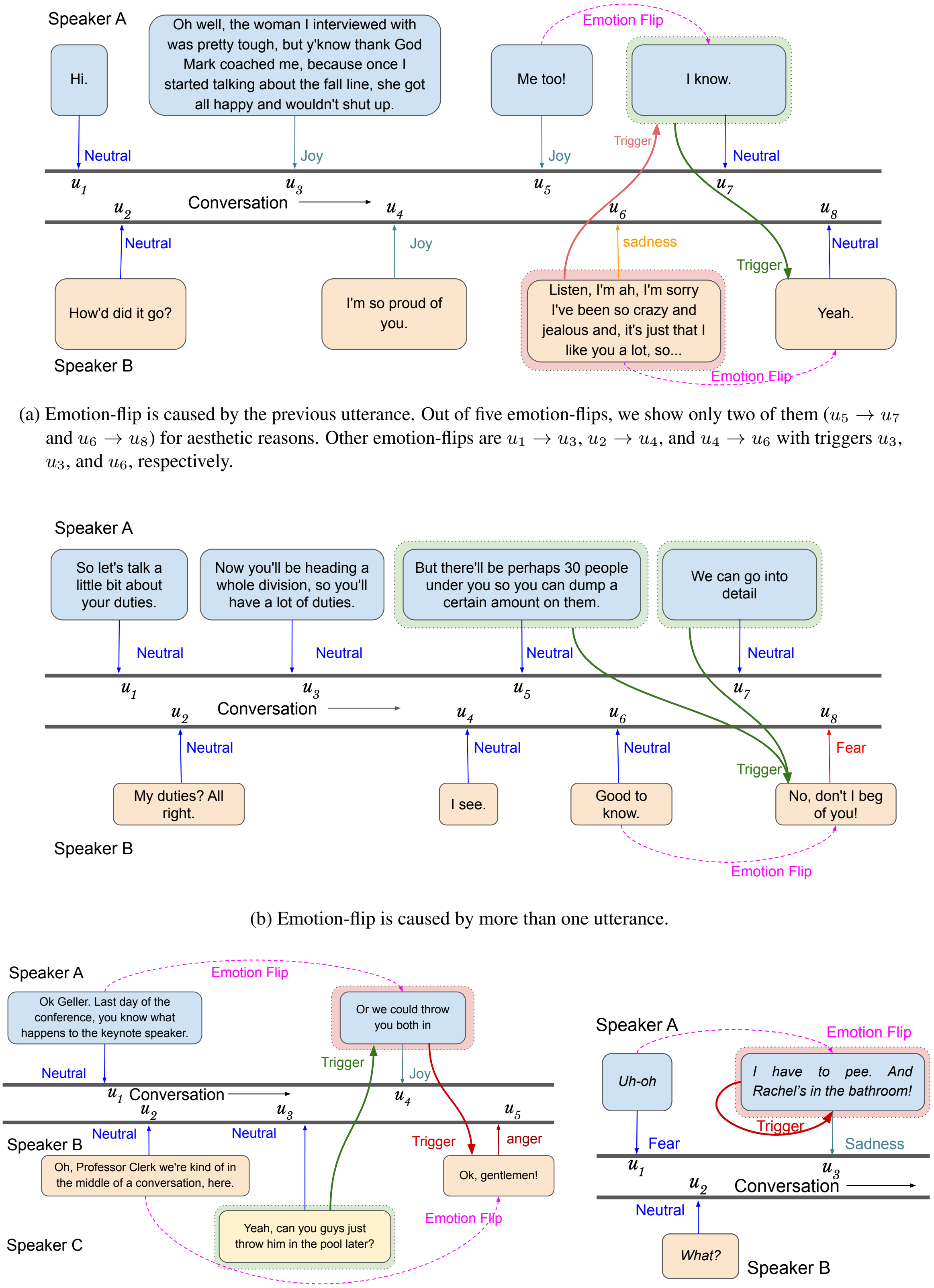
1. Given a sequence of dialogue utterances with emotion labels, the first stage of EFR identifies the utterance where a speaker experienced a flip of emotion.
2. In the second stage, EFR identifies utterances or triggers responsible for the emotion flip.
3. Finally, EFR assigns psychologically motivated (132; 133) instigator labels to triggers to explain the emotion flip.

Problem Definition The first stage can be effortlessly executed from a dialogue with emotion labels and this chapter focuses on the second and third phase of EFR. The second phase of EFR aims to find all utterances that trigger a flip in emotion of a speaker within a dialogue. A few example scenarios are presented in Figure 3.1. The first dialogue in Figure 3.1a exhibits five emotion-flips, i.e., u_1 (*neutral*) $\rightarrow u_3$ (*joy*), u_2 (*neutral*) $\rightarrow u_4$ (*joy*), u_4 (*joy*) $\rightarrow u_6$ (*sadness*), u_5 (*joy*) $\rightarrow u_7$ (*sadness*), and u_6 (*sadness*) $\rightarrow u_8$ (*neutral*); and the utterances that trigger the emotion flips are u_3, u_3, u_6, u_6 and u_7, respectively. Note that some emotion-flips might not be triggered by other speakers in the dialogue; instead, the target utterance can act as a *self-trigger*. We show such a scenario in Figure 3.1d in which utterance u_3 is the only reason for the emotion-flip observed. On the other hand, Figure 3.1b shows a case where more than one trigger is instigating an emotion-flip while Figure 3.1c presents an example where more than two speakers are involved in the conversation. In such a case, the trigger can come from any of the speakers' utterances.

On similar lines, the third phase of EFR aims to find psychologically motivated instigators for an emotion-flip in a conversation. Formally, given a sequence of n tuples of the form $\langle u_i, s_j, e_k \rangle$ in a multiparty conversation, where $s_j \in S$ is a speaker from a predefined speaker set S, $e_k \in E$ is a set of emotion labels, and $u_i \in D$ is an utterance of the dialogue D, we associate psychologically motivated instigator label(s) with a trigger utterance u_l if it causes a flip/change in emotion of a speaker $s_m \in S$ in their consecutive utterances in the conversation. Following the cognitive appraisal theory (133), we define a finite set of 27 instigators to reason out flips. Here, we do not account for implicit emotion flips, i.e., emotion flips due to the absence of explicit instigator (e.g., verbal articulation). For instance, emotion flips due to reminiscence can be regarded as implicit. On the other hand, an external trigger is associated with an emotion flip that occurs due to something mentioned in the text (e.g., a person getting scolded).

Figure 3.2a illustrates an example of emotion flip with corresponding instigators. It shows a multiparty scenario where three speakers are engaged in a conversation. There are two emotion flips – $\langle u_1, Ross, fear \rangle \rightarrow \langle u_3, Ross, joy \rangle$ and $\langle u_3, Ross, joy \rangle \rightarrow \langle u_5, Ross, anger \rangle$. The first flip occurs due to two trigger utterances, u_2 and u_3, each evoking the feeling of nervousness and adoration in the speaker (Ross). Consequently, the instigator labels for the concerned flip would be *Nervousness* and *Adoration*. On the other hand, the trigger for the second flip is a single utterance (u_4), and the corresponding instigator labels are *Annoyance* and *Challenge* as the trigger instigates the notion of annoyance and challenge in the speaker (Ross). This example highlights the case when more than one trigger utterance can cause an emotion flip. In Figure 3.2b, we show another example from our dataset. It shows a dyadic conversation having two emotion flips (u_3 and u_4) corresponding to two speakers (Monica and Chandler) involved in the conversation. The emotion flip at u_3 is an example of a self-trigger emotion flip – the responsible utterance (or trigger) is u_3 itself. Moreover, the same utterance u_3 acts as the trigger for both the emotion flips but causes different instigators in the two cases. It is interesting to note that the same utterance causes the emotion *sadness* in one speaker while the emotion *surprise* in another. This highlights the importance of identifying the emotion instigators to understand emotion dynamics completely. To summarise, our contributions are:

- We propose a novel task, called emotion-flip reasoning, in the conversational dialogue.
- We carefully draft a set of ground-truth labels, called instigators, to explain an emotion flip.
- We develop two new ground-truth datasets for EFR, called MELD-FR and MELD-I.
- We benchmark MELD-FR and MELD-I through Transformer and GRU based models.


Speaker A
Hi.
Oh well, the woman I interviewed with was pretty tough, but y'know thank God Mark coached me, because once I started talking about the fall line, she got all happy and wouldn't shut up.
Me too!
I know.
Emotion Flip
Neutral
Joy
Joy
Trigger
Neutral
u_1
u_3
u_5
u_7
u_2
Conversation
u_4
u_6
u_8
Neutral
Joy
sadness
Neutral
Trigger
How'd did it go?
I'm so proud of you.
Listen, I'm ah, I'm sorry I've been so crazy and jealous and, it's just that I like you a lot, so...
Yeah.
Speaker B
Emotion Flip

(a) Emotion-flip is caused by the previous utterance. Out of five emotion-flips, we show only two of them ($u_5 \rightarrow u_7$ and $u_6 \rightarrow u_8$) for aesthetic reasons. Other emotion-flips are $u_1 \rightarrow u_3$, $u_2 \rightarrow u_4$, and $u_4 \rightarrow u_6$ with triggers u_3, u_3, and u_6, respectively.

Speaker A
So let's talk a little bit about your duties.
Now you'll be heading a whole division, so you'll have a lot of duties.
But there'll be perhaps 30 people under you so you can dump a certain amount on them.
We can go into detail
Neutral
Neutral
Neutral
Neutral
u_1
u_3
u_5
u_7
u_2
Conversation
u_4
u_6
u_8
Neutral
Neutral
Neutral
Fear
Trigger
My duties? All right.
I see.
Good to know.
No, don't I beg of you!
Speaker B
Emotion Flip

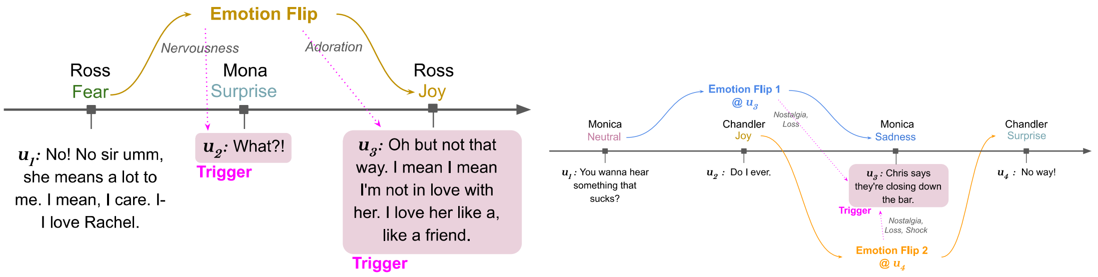
(b) Emotion-flip is caused by more than one utterance.

Speaker A
Emotion Flip
Ok Geller. Last day of the conference, you know what happens to the keynote speaker.
Or we could throw you both in
Trigger
Neutral
Joy
u_1 Conversation
u_4
u_2
u_3
u_5
Speaker B
Neutral
Neutral
Trigger
anger
Oh, Professor Clerk we're kind of in the middle of a conversation, here.
Ok, gentlemen!
Emotion Flip
Speaker C
Yeah, can you guys just throw him in the pool later?
Speaker A
Emotion Flip
Uh-oh
I have to pee. And Rachel's in the bathroom!
Trigger
Fear
Sadness
u_1
u_3
u_2
Conversation
Neutral
What?
Speaker B

(c) Emotion-flip in a multi-party conversation.

(d) Self-trigger emotion-flip.

Figure 3.1: Examples of trigger identification in emotion-flip reasoning.


Emotion Flip
Nervousness
Adoration
Ross
Fear
Mona
Surprise
Ross
Joy
u_1: No! No sir umm, she means a lot to me. I mean, I care. I- I love Rachel.
u_2: What?!
Trigger
u_3: Oh but not that way. I mean I mean I'm not in love with her. I love her like a, like a friend.
Trigger
Emotion Flip 1 @ u_3
Nostalgia, Loss
Monica
Neutral
Chandler
Joy
Monica
Sadness
Chandler
Surprise
u_1 : You wanna hear something that sucks?
u_2 : Do I ever.
u_3 : Chris says they're closing down the bar.
Trigger
Nostalgia, Loss, Shock
Emotion Flip 2 @ u_4
u_4 : No way!

(a) Example of an emotion flip with triggers and instigators Ross's emotion flipped from *Fear* (u_1) to *Joy* (u_3) due to two trigger utterances (u_2 and u_3) caused by the instigators, *nervousness* and *adoration*, respectively.

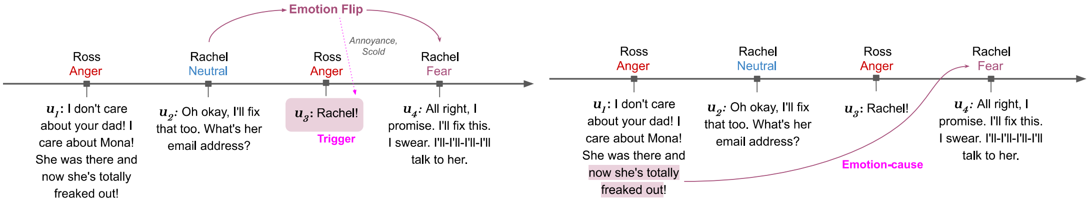
(b) Example of an emotion flip with self-trigger. Monica's emotion flipped from *Neutral* (u_1) to *Sadness* (u_3) due to one trigger utterance (u_3 itself) caused by the instigators *nostalgia* and *loss*. The other speaker's emotion then flipped from *Joy* (u_2) to *Surprise* (u_4) due to a single trigger utterance (u_3 again) caused by the instigators *nostalgia*, *loss* and *shock*.

Figure 3.2: Examples of instigator labeling in emotion-flip reasoning.

3.2 Related Work

Emotion Recognition. Earlier studies in the field of emotion analysis (29; 134; 135; 136; 137; 138) dealt with only standalone inputs. A detailed survey is provided by (139). However, these studies are performed for standalone text, which lacks any contextual information. Recently, the focus of emotion detection has shifted to conversations. It has gained significant popularity due to numerous applications. Existing literature suggests that a wide range of deep learning methods have been applied to address the ERC task (43; 44; 45; 46; 47; 49; 50; 92; 48; 80; 51; 52; 140). ICON (43) used a memory network architecture to model the interaction between self and inter-speaker states in two-party conversations. On the other hand, the use of external knowledge has also been explored (44) along with a hierarchical self-attention mechanism to detect emotions in conversation. BiERU (45) used a party ignorant framework for conversation sentiment analysis. The use of graph convolutional networks to capture the inter-speaker dynamics in a dialogue has also been explored (46). They utilized the dependencies among speakers to capture the contextual dynamics in an efficient way. In another work, AGHMN (47) proposed a hierarchical memory network with an attention mechanism to capture the essence of the dialogue in order to get a better understanding of the emotional dynamics of the speaker. TL-ERC (49) exploited the learned parameters of a dialogue generation module for emotion recognition through the transfer learning setup. Recently, DialogXL (50) adopted XLNet (141) model for ERC. They encoded the dialogue utterances and made use of dialogue-aware self-attention to exploit the dialogue semantics. A hierarchical gated recurrent unit framework involving two GRUs at different levels was employed in a study (48) where a lower-level GRU modeled the word-level inputs while an upper-level GRU captured the context at the utterance level. Further, a correction model for previous approaches called “Dialogical Emotion Correction Network (DECN)" was introduced (142). The aim of this work was to improve upon the emotion recognition performance by automatically identifying errors made by emotion recognition strategies. The authors proposed the use of a graphical network to model human interactions in dialogues. Another study (143) used graph to solve the problem of ERC. They proposed a conversational affective analysis model which combined dependent syntactic analysis and graph convolutional neural networks. A self-attention mechanism captures the most effective words in the conversation, followed by graph construction. The authors shows experiments on various datasets the report higher accuracy than previous methods.

Beyond Emotion Recognition. Most of the existing ERC systems do not account for the explainability of emotions. In an attempt to do so, the task of emotion-cause analysis was proposed (144). The task


Emotion Flip
Annoyance, Scold
Ross
Anger
Rachel
Neutral
Ross
Anger
Rachel
Fear
u1: I don't care about your dad! I care about Mona! She was there and now she's totally freaked out!
u2: Oh okay, I'll fix that too. What's her email address?
u3: Rachel!
Trigger
u4: All right, I promise. I'll fix this. I swear. I'll-I'll-I'll-I'll talk to her.
Emotion-cause

(a) Emotion flip reasoning

(b) Emotion-cause extraction

Figure 3.3: A sample dialogue to illustrate the difference between emotion-cause extraction in conversation and emotion-flip reasoning.

deals with identifying a span in the text responsible for a specific emotion. For instance, we observe two emotions in the sentence '*The queue was so long, but at last I got vaccinated*' – *joy* and *disgust*. The task aims at identifying the phrases '*the queue was long*' for *disgust* and '*I got vaccinated*' for *joy*. Following this work, a study (145) investigated the use of linguistic phenomenon by proposing an SVM-based model for emotion-cause identification. Xia et al. (146) proposed another task- emotion-cause pair extraction. This task tried to extract the potential pairs of emotions and the corresponding causes in a document. The proposed a two-step approach where, first, individual emotion extraction and cause extraction are performed via multi-task learning and then emotion-cause pairing and filtering are done. In another one of their work (147), they proposed a joint emotion-cause extraction framework which consisted of two encoders. A RNN based encoder was employed to get the word-level representations while a Transformer based encoder was applied to to learn the correlation between multiple clauses in a document. They also encoded relative position and global predication information that they claim helped capture the causality between clauses. Recently, the emotion cause extraction task has been extended for conversation (148) and the authors released a dataset called RECCON containing $1,000+$ dialogues accompanied by $10,000$ emotion-cause pairs.

How is our Task Different? EFR represents a novel paradigm in NLP as it deals explicitly and quantitatively with identifying emotion triggers and instigators. While word embeddings may contain some implicit information about utterance meaning and emotion dynamics, they provide no explainability for an emotion-flip, and hence, cannot be used as a potential feedback mechanism to a response generator. Additionally, the two tasks, namely emotion-cause extraction in conversation and EFR, may seem similar at an abstract level; nonetheless, they differ considerably at the surface level. While emotion-cause extraction in conversation aims to extract a text span that acts as grounds for the elicited emotion, EFR is a more speaker-specific task that highlights the triggers and instigators responsible for a "flip" in the speakers' emotion. In our case, the triggers come from the dialogue context, while instigators (or causes) for an emotion flip come from a finite set of predefined labels, in contrast with the infinite possibilities of a span that the emotion-cause extraction task can extract. In order to reinforce the difference between the two tasks, we show a sample dialogue in Figure 3.3 from MELD-I with annotated EFR and emotion-cause labels. It can be observed that the reason behind the emotion *fear* in utterance u_4 comes from utterances u_1 and u_3. On the other hand, the emotion flip from *neutral* to *fear* (from utterance u_2 to u_4) was triggered by the utterance u_3 because of the feelings of *annoyance* and *scold* being instigated in the target speaker.

3.3 Dataset

The task of Emotion Flip Reasoning consists of identifying trigger utterances and psychologically motivated instigator labels. To this end, we curate two specific datasets – MELD-FR and MELD-I for the task of trigger identification and instigator recognition, respectively. We explain the creation process for these datasets in this section.

3.3.1 MELD-FR

We employ a recently released dataset, called MELD (85) and extend it for our EFR task. As described in the previous chapter, it consists of $13,708$ utterances spoken by multiple speakers across $1,433$ dialogues, where each utterance has an associated emotion label representing one of Ekman's six basic emotions: *anger*, *fear*, *disgust*, *sadness*, *joy*, and *surprise* along with a label for no emotion, i.e., *neutral*. Though MELD is a multi-modal dataset, in this work, we employ textual modality only.

For EFR, we augment MELD with new ground-truth labels (dubbed as **MELD-FR**). For this, we take inspiration from the *Cognitive Appraisal Theory* by Richard Lazarus (149) which states that emotions are a result of our evaluations or appraisals of an event. We extend the concept of appraisals and try to identify these in our dialogue instances. We consider the utterances that contain possible appraisals as ***triggers***. We employed three annotators who had vast experiences in the ERC task[1]. Below, we explain the steps carried out for the EFR annotation.

1. For each speaker s_j, we identify their utterances $u_i^{s_j}$ in a dialogue where a flip in emotion has occurred, i.e., the speaker's last emotion and the current emotion are different.
2. For each identified utterance $u_i^{s_j}$, we analyse the dialogue context and mark all the utterances u_k (where $1 \leq k \leq i$) as triggers that are responsible for the emotion-flip in utterance $u_i^{s_j}$. For some of the cases, the reason for the emotion-flip of a speaker was not apparent in the dialogue, and the flip was self-driven by the speaker. We leave such cases and do not mark any triggers for them.

Annotation Guidelines

For annotating triggers, we define a set of guidelines as furnished below. We define a ***trigger*** as any utterance in the contextual history of the target utterance (the utterance for which the trigger is to be identified) that follows the following properties:

1. The whole utterance or a part of utterance directly influences a change in emotion of the target speaker.
2. The utterance can be uttered by a different speaker or the target speaker.
3. The target utterance can also be classified as a trigger utterance if it contributes to the emotion-flip of the target speaker. For example, if a person's emotion changes from *neutral* to *sad* because of some sad message that she is conveying herself, then the target utterance is the one responsible for the shift.
4. There can be more than one trigger responsible for a single emotion-flip.
5. Since we deal with textual data only, it is possible that the reason behind an emotion-flip is not evident from the data (for example, when the flip occurs due to a visual stimulus). In such cases, no utterance can be marked as a trigger.

We calculate the alpha-reliability inter-annotator agreement (108) between each pair of annotators, $\alpha_{AB} = 0.824$, $\alpha_{AC} = 0.804$, and $\alpha_{BC} = 0.820$. To find out the overall agreement score, we take the average score, $\alpha = 0.816$. Figure 3.1 shows a few example scenarios from our dataset. While Figures 3.1a, 3.1b, and 3.1d illustrate the case when two participants are involved in a conversation, Figure 3.1c shows an example where more than two speakers are involved. For our work, we considered only those dialogues where speakers experience at least one emotion-flip. After removing dialogues with no emotion-flip, we were left with 834 dialogues in the training set, which account for $4,001$ utterances with emotion flips. These dialogues were annotated by three annotators according to the above guidelines for identifying triggers. Among three annotators, two of the annotators were male, and one was female. All of them were

[1] However, the annotation process can easily be carried out by anyone if one follows the annotation guidelines we have provided.

researchers with 3-10 years of experience. They belong to the age group of 30-40. Though we employed three expert annotators in our annotation phase, the process does not require experts (linguistics, social scientists, etc.). Since the annotation guidelines that we provide above are very generic, they can easily be extended to other dialogue datasets by crowdsourcing. We wanted to prepare our labeled dataset as accurately as possible as it would be the first dataset of its kind.

Split	#Dialogue with Flip	#Utterance with Flip	#Triggers
Train	834	4001	6740
Dev	95	427	495
Test	232	1002	1152

Table 3.1: Statistics for MELD-FR dataset for EFR. We only consider those dialogues from the original MELD dataset where there is at least one emotion-flip. This step removed 271 dialogues from MELD, resulting $1,161$ dialogues.

Similarly, we obtained the trigger annotations for the development and test sets. We show a brief statistic of the datasets in Table 3.6. The resultant dataset, called MELD-FR, contains $8,387$ trigger utterances for $5,430$ emotion-flips. We also show the EFR trigger distribution considering their distance from the target utterance in Figure 3.4. We observe that for the majority of the emotion-flips, the triggers appear in the past few utterances only.

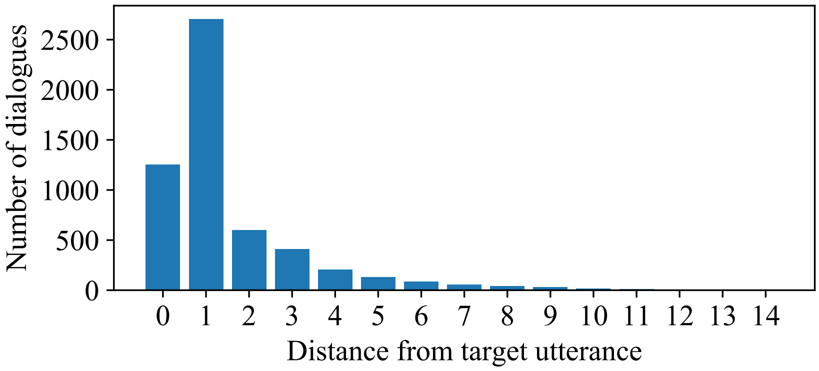


Figure 3.4: Distribution of triggers w.r.t their distance from the target utterance.

After the annotation process, we analyze the directionality of the emotion-flips in MELD-FR. Table 3.2 shows the statistics of the emotion-flip from the source emotion (row) to the target emotion (column). We make some interesting observations. We consider the set of emotion- *joy* and *surprise* as 'positive' and the set- *disgust*, *fear*, *anger*, and *sadness* as 'negative' emotions. We analyse the emotion flips based on these positive and negative emotion sets. There are in total 2612 emotion flips which result in an emotion from the positive set, whereas 2818 emotion flips result in an emotion from the negative set. Out of these flips, the most prominent emotion-flip pairs are *neutral* to *joy* (616) when the resultant (or target) emotion is positive, and *neutral* to *anger* (370) when the target is negative. We also analyse the flips occurring from a positive to positive emotions (e.g. *joy* to *surprise*) or from negative to negative emotions (e.g. *anger* to *fear*). Within the positive class, the most emotion flips are observed for the pair *surprise* to *joy* (186), while *anger* to *sadness* flip (99) prevails the negative class. Most of the emotion flips that result in a positive emotion originate from *neutral* (1103), while most emotions flips that result in a negative emotion originate from *joy* (907). When the target emotion of the emotion-flip is a positive emotion, it is mostly *joy* (1020), whereas for the negative case, it is mostly *anger* (757). We also observe that the most frequent reasons for positive emotion-flip are excitement, cheer, or being impressed by someone else. For negative emotion-flip, awkwardness, loss, or being annoyed are the frequent reasons.

		Target							
		Disgust	**Joy**	**Surprise**	**Anger**	**Fear**	**Neutral**	**Sadness**	**Total**
Source	**Disgust**	0	24	30	47	6	76	13	196
	Joy	34	0	169	86	42	665	81	1077
	Surprise	39	186	0	137	32	400	70	864
	Anger	37	96	104	0	20	318	99	674
	Fear	7	20	23	45	0	87	27	209
	Neutral	84	616	487	370	103	0	257	1917
	Sadness	17	78	60	72	28	238	0	493
	Total	218	1020	873	757	231	1784	547	

Table 3.2: Directionality for emotion flips. The value at cell (i, j) denotes the number of emotion-flip from emotion i to emotion j.

3.3.2 MELD-I

We further extend the MELD dataset to augment it with intigator labels for identified emotion flips and corresponding triggers. Specifically, we propose new labels and convert the original MELD dataset into instances based on emotion-flips resulting in 1161 dialogue instances. We manually annotate all these instances with the proposed instigator labels. However, since we use the dialogues and emotion labels from the MELD dataset, we keep the name of the new dataset derived from it- MELD-I.

		Instigator
Sentiment	**Positive**	adoration, benefit, calmness, cheer, desire, excitement, humour, impressed, relief, satisfaction
	Negative	abuse, annoyance, guilt, horror, loss, nervousness, pain, scold, shock, sympathy, threat
	Ambiguous	awkwardness, boredom, challenge, confusion, curiosity, nostalgia

Table 3.3: Division of 27 instigators into positive, negative, and ambiguous instigators.

Instigator	**Coarse-grained**	Annoyance		Awkwardness	Benefit	Cheer		Confusion
	Fine-grained	Annoyance	Pain	Awkwardness	Benefit	Cheer	Humour	Confusion
	Coarse-grained	Curiosity	Ease		Excitement			
	Fine-grained	Curiosity	Calmness	Relief	Excitement	Satisfaction	Desire	
	Coarse-grained	Threat			Others			
	Fine-grained	Threat	Horror	Abuse	Boredom	Sympathy	Challenge	Nostalgia
	Coarse-grained	Dazzle		Loss	Nervousness			Shock
	Fine-grained	Adoration	Impressed	Loss	Nervousness	Scold	Guilt	Shock

Table 3.4: Instigator labels with their definitions.

Instigator Labels

To understand the emotional dynamics of the speakers in a conversation, it is imperative to reason out any change/flip of the emotion of any speaker. Following the Cognitive Appraisal Theory by Lazarus et al. (133), which states that emotions are a result of appraisals, we aim to identify these appraisals for each emotion flip in the dialogue. These instigators follow the following properties:

- The instigators need not be unique to an emotion flip. For example, *threat* can instigate the emotion flip *joy* → *fear* as well as the flip *anger* → *fear*.

	Speaker	Utterance	Emotion	
u_1	Ross	No! No sir umm, she means a lot to me. I mean, I care I-I love Rachel.	Fear	
u_2	Mona	What?!	Surprise	
u_3	Ross	Oh but not that way. I mean I mean I'm not in love with her. I love her like a, like a friend.	Joy	Emotion flip - 1
u_4	Dr. Green	Oh really? That's how you treat a friend? You get her in trouble and then refuse to marry her?	Anger	
u_5	Ross	Hey! I offered to marry her!	Anger	Emotion flip - 2

(a) An example dialogue from MELD.

	Speaker	Utterance	Emotion
u_1	Ross	No! No sir umm, she means a lot to me. I mean, I care I-I love Rachel.	Fear
u_2	Mona	What?!	Surprise
u_3	Ross	Oh but not that way. I mean I mean I'm not in love with her. I love her like a, like a friend.	Joy

(b) Instance - 1.

	Speaker	Utterance	Emotion	Trigger	Instigator
u_1	Ross	No! No sir umm, she means a lot to me. I mean, I care I-I love Rachel.	Fear	No	-
u_2	Mona	What?!	Surprise	Yes	Nervousness
u_3	Ross	Oh but not that way. I mean I mean I'm not in love with her. I love her like a, like a friend.	Joy	Yes	Adoration

(c) MELD-I Annotation: Trigger/Instigator.

	Speaker	Utterance	Emotion
u_1	Ross	No! No sir umm, she means a lot to me. I mean, I care I-I love Rachel.	Fear
u_2	Mona	What?!	Surprise
u_3	Ross	Oh but not that way. I mean I mean I'm not in love with her. I love her like a, like a friend.	Joy
u_4	Dr. Green	Oh really? That's how you treat a friend? You get her in trouble and then refuse to marry her?	Anger
u_5	Ross	Hey! I offered to marry her!	Anger

(d) Instance - 2.

	Speaker	Utterance	Emotion	Trigger	Instigator
u_1	Ross	No! No sir umm, she means a lot to me. I mean, I care I-I love Rachel.	Fear	No	-
u_2	Mona	What?!	Surprise	No	-
u_3	Ross	Oh but not that way. I mean I mean I'm not in love with her. I love her like a, like a friend.	Joy	No	-
u_4	Dr. Green	Oh really? That's how you treat a friend? You get her in trouble and then refuse to marry her?	Anger	Yes	Annoyance, Challenge
u_5	Ross	Hey! I offered to marry her!	Anger	No	-

(e) MELD-I Annotation: Trigger/Instigator.

Table 3.5: Dataset development for an instance shown in Figure 3.2a. (a) Original dialogue from MELD; (b,d) Two instances corresponding to the two emotion flips in (a); (c,e) Trigger and instigator annotations for both instances.

- An emotion flip need not necessarily arise from the same set of instigators. For example, the emotion flip *neutral* → *fear* can be caused by *threat* and *challenge* in different situations.
- There can be more than one instigator corresponding to a single emotion flip. For example, for the emotion flip *neutral* → *fear*, the instigator can be both *threat* and *challenge*.
- The instigators cannot be emotions themselves. For example, for the emotion flip *neutral* → *surprise*, the instigator cannot be *joy*.

We organize these instigators in a 2-level hierarchy. The first level presents a coarser representation of the instigators with 14 labels, while the second level defines all 27 instigators as fine-grained representation. Table 3.4 presents the hierarchy of instigators and their definitions. Further, these instigators can be divided into three sets, based on the target emotion they can instigate- positive, negative, and neutral. Division of the instigator labels into the set of positive, negative, and ambiguous can be seen in Table 3.3.

Annotation Process

The first step in our annotation process is the instance creation for each emotion flip, followed by the trigger identification and instigator labeling. Table 3.5 presents the outcome of the annotation process for the example shown in Figure 3.2a. We explain these steps in detail below.

1. **Instance creation:** For each emotion flip of a speaker, we create a new instance. The instance contains the utterances from the beginning of the dialogue till the target utterance (emotion flipped utterance). Utterances u_3 and u_5 are the target utterances in Table 3.5, and utterances $\langle u_1, u_2, u_3 \rangle$ and $\langle u_1, u_2, u_3, u_4, u_5 \rangle$ are the respective candidate triggers for the target utterances. Among these candidates, u_2, u_3 are the triggers for the target u_3, while utterance u_4 instigates the emotion flip in the target u_5. Intuitively, the last utterance of each instance is the target utterance – the location of

emotion flip. Note that we remove all such dialogues from MELD that do not contain an emotion flip which removed 271 dialogues from the set.

2. **Trigger identification:** After creating an instance for each emotion flip, we identify a set of trigger utterances that cause the emotion to flip at the target. We mark each utterance that acts as a trigger as 'Yes' and the ones not contributing as 'No'. The two instances in Table 3.5 have utterances $\langle u_2, u_3 \rangle$ and $\langle u_4 \rangle$ as triggers for the target utterances u_3 and u_5, respectively.
3. **Instigator labeling:** Finally, we assign one or more instigator labels to each trigger utterance corresponding to the target utterance. For example, as presented in Table 3.5, we assign '*nervousness*' and '*adoration*' instigators to the trigger utterances u_2 and u_3, respectively, for the target u_3. Similarly, for the target utterance u_5 in Table 3.5, we annotate the trigger u_4 with two instigator labels '*annoyance*' and '*challenge*'. It is evident that the instigator identification is a multi-label problem.

We employ the services of three annotators[2] to annotate MELD-I – two of them in the first stage of annotation, while the service of the third expert was sought to resolve any disagreement. We compute Krippendorff's Alpha inter-annotator agreement (108) to measure the consistency in the annotation. For trigger identification, we obtain the inter-annotator agreement between annotators A and B as $\alpha_{AB} = 0.817$, between annotators B and C as $\alpha_{BC} = 0.820$, and between annotators A and C as $\alpha_{AC} = 0.811$. We take the average of these three to get the overall agreement rating, i.e., $\alpha = 0.816$. For the instigator annotation, $\alpha_{AB} = 0.511$, $\alpha_{BC} = 0.545$, and $\alpha_{AC} = 0.540$. The average agreement comes out to be $\alpha = 0.532$. The low value for the latter is attributed to the multi-label characteristic of the task.

Dataset Statistics

We show a brief statistic of MELD-I in Table 3.6 along with the distribution of emotion flips. We also show the distribution of instigators in Figure 3.5. We can observe from Figure 3.5b that the distribution of fine-grained instigator labels is skewed towards a few of the instigators. As an attempt to reduce the skewness, we group similar instigator labels and obtain a reduced set of 14 instigators in the coarse-grained setup (c.f. Figure 3.5a).

Split	#Dialogue with Flip	#Utterance with Flip	#Triggers
Train	834	4001	5262
Dev	95	427	495
Test	232	1002	1152

(a) MELD-I dataset for EFR

		Target						
		Disgust	**Joy**	**Surprise**	**Anger**	**Fear**	**Neutral**	**Sadness**
Source	**Disgust**	0	24	30	47	6	76	13
	Joy	34	0	169	86	42	665	81
	Surprise	39	186	0	137	32	400	70
	Anger	37	96	104	0	20	318	99
	Fear	7	20	23	45	0	87	27
	Neutral	84	616	487	370	103	0	257
	Sadness	17	78	60	72	28	238	0

(b) Frequency of emotion flips with respect to emotions

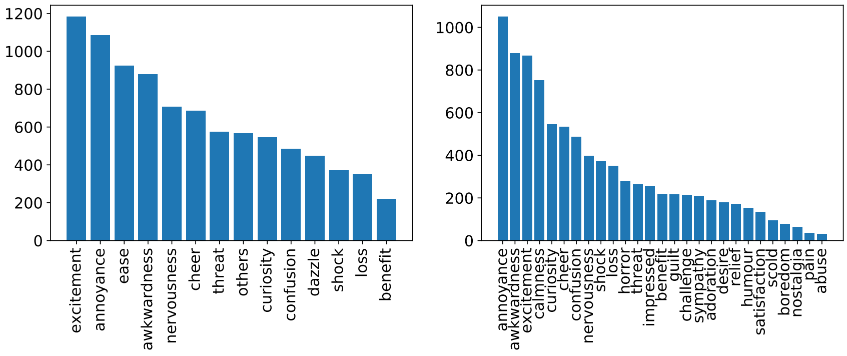
Table 3.6: Statistics of the dataset, MELD-I.

It is interesting to note that emotion flips can be divided into two categories – **positive** emotion flips ({*anger*, *fear*, *disgust*, *sadness*} → {*joy*, *surprise*, *neutral*}) and **negative** emotion flips ({*joy*, *surprise*} → {*anger*, *fear*, *disgust*, *sadness*, *neutral*}). The flips {*neutral*} → {*joy*, *surprise*} are also considered as positive emotion flips whereas {*neutral*} → {*anger*, *fear*, *disgust*, *sadness*} are considered as negative emotion flips. Considering the above categorization of emotion flips, we observe that not all

[2] They are NLP researchers or linguistics by profession; their age ranges between $30 - 45$ years.

instigators can result in all emotions flips. For example, it is improbable for a person to feel *joy* because of *guilt* – for an emotion flip with the target emotion *joy*, the instigator can almost never be *guilt*. Our observation of the annotated dataset is in line with this phenomenon.

Consequently, we divide our instigator labels into three sets – positive, negative, and ambiguous. We observe that for a positive emotion flip, only the instigators belonging to the positive and ambiguous set of instigators are responsible. Similarly, for a negative emotion flip, the negative and ambiguous sets are applicable.



(a) Coarse-grained

(b) Fine-grained

Figure 3.5: Distribution of EFR instigators in MELD-I.

3.4 Methodology

3.4.1 EFR-TX

For the EFR task, we make a sequence of predictions corresponding to each previous utterance u_i, where $i \leq t$, denoting the trigger/reason behind emotion-flip at the target utterance u_t. We employ an instance-level Transformer-based (150) encoder. Figure 3.6 presents a high-level architecture of our model.

We model the emotion-flip reasoning task as an instance-based classification problem. At first, we define an EFR instance as a sequence of utterances, $u_1, u_2, ..., u_t$, where the aim is to identify a set of triggers for the target (last) utterance u_t. Intuitively, it can be observed that the triggers for the target utterance u_t must belong to the utterance set $u_1, u_2, ..., u_t$. Thus, we classify each utterance in the instance as trigger/non-trigger for the target utterance u_t. We repeat the same process for each utterance in the dialogue; however, if no emotion-flip is observed for the target utterance u_t in the training set, we do not process the error gradients in the backward pass.

	Target u_t	**Instance**	**Trigger labels**
Figure 3.1a	u_3	$\{u_1, u_2, u_3\}$	{0, 0, 1}
	u_4	$\{u_1, u_2, u_3, u_4\}$	{0, 0, 1, 0}
	u_6	$\{u_1, u_2, u_3, u_4, u_5, u_6\}$	{0, 0, 0, 0, 0, 1}
	u_7	$\{u_1, u_2, u_3, u_4, u_5, u_6, u_7\}$	{0, 0, 0, 0, 0, 1, 0}
	u_8	$\{u_1, u_2, u_3, u_4, u_5, u_6, u_7, u_8\}$	{0, 0, 0, 0, 0, 0, 1, 0}
Figure 3.1b	u_8	$\{u_1, u_2, u_3, u_4, u_5, u_6, u_7, u_8\}$	{0, 0, 0, 0, 1, 0, 1, 0}
Figure 3.1c	u_4	$\{u_1, u_2, u_3, u_4\}$	{0, 0, 1, 0}
	u_5	$\{u_1, u_2, u_3, u_4, u_5\}$	{0, 0, 0, 1, 0}
Figure 3.1d	u_3	$\{u_1, u_2, u_3\}$	{0, 0, 1}

Table 3.7: Instance creation corresponding to the dialogues in Figure 3.1 for the EFR task.

We show the instance creation process for all the example dialogues (as shown in Figure 3.1) in Table 3.7. In the first example (Figure 3.1a), there are five utterances, u_3, u_4, u_6, u_7 and u_8, where the emotion of the speakers has flipped; thus, we have created five instances for each target emotion-flip; the corresponding trigger utterances are u_3, u_3, u_6, u_6 and u_7, respectively. In the second one (Figure 3.1b), we show an instance where triggers can come from multiple utterances. Here, the emotion of the speaker has flipped once in utterance u_8, and the triggers were identified as the utterances u_5 and u_7. We also show an example when more than two interlocutors are involved in a conversation (Figure 3.1c). In the particular example, there are two emotion flips at utterances u_4 and u_5. The triggers utterances for the same are u_3 and u_4, respectively. In the last example (Figure 3.1d), we show an instance where the emotion-flip is a result of a self trigger, i.e., for the emotion-flip observed at utterance u_3, the trigger is the same utterance itself.

After compiling the EFR instances, we employ the Transformer model for classification. We obtain the encoder output h_i for each utterance of an instance, and concatenate it with the encoder output of the target utterance h_t, i.e., $\forall i, \hat{h}_i = h_i \oplus h_t$. Since emotion-flip reasoning has a strong correspondence with the emotion label, we supplement each contextual utterance with its emotion label to learn an appropriate representation for the trigger classification. Subsequently, we classify each utterance as trigger/non-trigger for the target utterance.

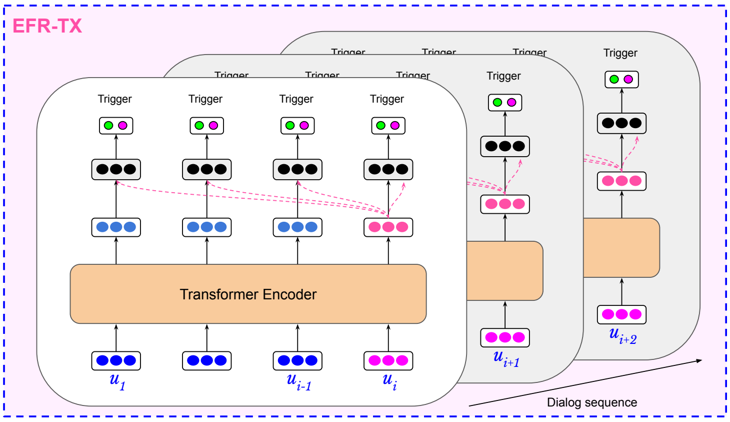


Figure 3.6: The proposed Transformer-based (EFR-TX) model for EFR. EFR-TX models each instance as a tuple <*past utterances as trigger candidates*, *target utterance as the location of emotion-flip*>.

3.4.2 TGIF

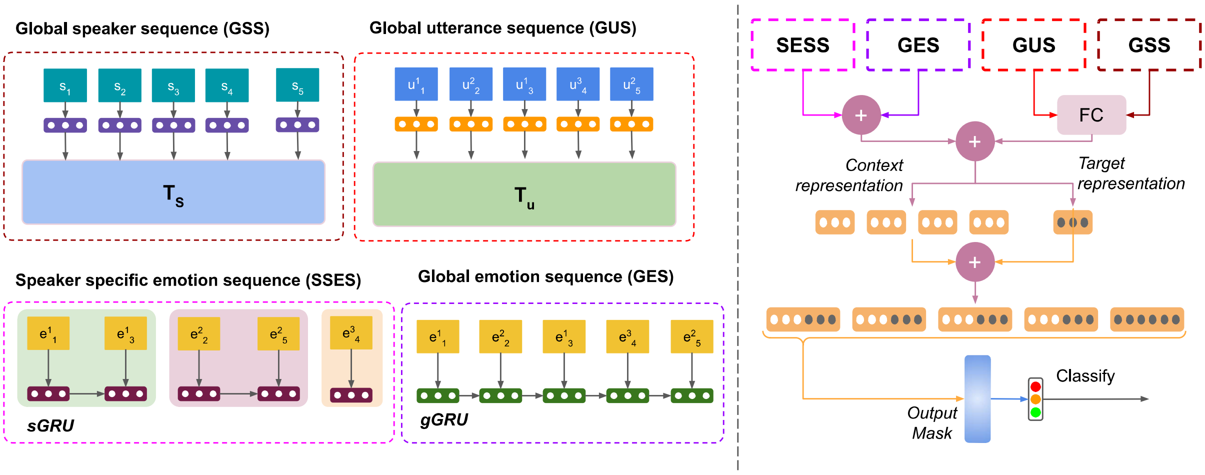

Global speaker sequence (GSS)
s_1 s_2 s_3 s_4 s_5
T_s
Global utterance sequence (GUS)
u^1_1 u^2_2 u^1_3 u^3_4 u^2_5
T_u
Speaker specific emotion sequence (SSES)
e^1_1 e^1_3 e^2_2 e^2_5 e^3_4
sGRU
Global emotion sequence (GES)
e^1_1 e^2_2 e^1_3 e^3_4 e^2_5
gGRU
SESS
GES
GUS
GSS
FC
Context representation
Target representation
Output Mask
Classify

Figure 3.7: The proposed TGIF architecture. Input: $\{\langle u_1, s_1, e_1\rangle, \langle u_2, s_2, e_2\rangle, \langle u_3, s_1, e_3\rangle, \langle u_4, s_3, e_4\rangle, \langle u_5, s_2, e_5\rangle\}$, where $\langle u_i, s_j, e_i\rangle$ represents the utterance u_i by a speaker s_j and its associated emotion e_i. Target (emotion flipped) utterance: $\langle u_5, s_2, e_5\rangle$ as $e_2 \neq e_5$.

The instigator identification task is a multi-label instance classification problem, as more than one instigator is possible for each trigger. TGIF models the global utterance sequence (*aka.* dialogue context) and speaker dynamics to capture the underlying semantics in the dialogue. Moreover, considering the strong relationship of emotion with our task, we also encode the emotion sequence of the utterances in TGIF. In total, TGIF has four submodules that exploit the global and speaker-specific dialogue and emotion dynamics – Global Utterance Sequence (GUS), Global Speaker Sequence (GSS), Global Emotion Sequence (GES), and Speaker-Specific Emotion Sequence (SSES). Finally, we combine the outputs of these four modules through a series of fully-connected layers followed by a 14/27 neurons sigmoid layer for both coarse-grained and fine-grained instigator identifications. Furthermore, at the penultimate layer, we apply an output mask to filter out the improbable instigator labels for the underlying emotion flip. The output mask assists the model in focusing on the probable labels and blocks the gradients for the unlikely labels to propagate back to the network. Below, we describe each module of TGIF in detail. Figure 5.5 presents the architecture of TGIF.

Global Utterance Sequence (GUS). The principle information about a dialogue lies in the utterances spoken in it. Thus, we employ GUS to encode the utterance sequence. We use a Transformer (151) encoder to extract a hidden representation h_i^u for each utterance u_i. For each, utterance, $u_i^{s_i}$, in the dialogue, we get an encoded vector, h_i^u, after this state, i.e. $\forall u_i^{s_i}, h_i^u = T_u(u_i^{s_i})$. Thus, h_i^u represents the context aware representation of the i^{th} utterance of the dialogue D.

Global Emotion Sequence (GES). In this module, we employ a single-layer GRU (152), $gGRU$, to capture the global emotion sequence of the dialogue. We hypothesize that the knowledge of emotion sequence would assist the model in capturing a high-level snapshot of the emotion flow among speakers. We feed the emotion sequence of the dialogue, $\{e_1, e_2, ..., e_t\}$, as input to the GRU where each emotion e_i is represented by a one-hot vector of dimension 7. As a result, we obtain the hidden representation as follows: $[h_1^e, .., h_2^e, .., h_t^e] = gGRU(e_1, e_2, .., e_t)$.

Speaker-Specific Emotion Sequence (SSES). Each emotion flip is associated with a speaker. Thus, we hypothesize that the sequence of emotions at the speaker level is crucial and would exploit the emotion dynamics of each speaker considering the target speaker. Moreover, it would distinguish between the emotional states of the target speaker and other speakers. To achieve this, we employ separate GRUs for each speaker in an instance.

For example, if there are three distinct speakers in an instance (c.f. instance 2 in Table 3.5), we learn three GRUs. For each speaker, we extract its emotion from the dialogue and create the input for GRUs as follows. Let an instance with five utterances of three distinct speakers be given as $\{\langle u_1, s_1, e_1\rangle$, $\langle u_2, s_2, e_2\rangle$, $\langle u_3, s_1, e_3\rangle$, $\langle u_4, s_3, e_4\rangle$, $\langle u_5, s_2, e_5\rangle\}$, where u_i and e_i denote the utterance and associated emotion at turn i by speaker s_j. We compile three inputs for each speaker as $\{e_1, e_3\}$, $\{e_2, e_5\}$, and $\{e_4\}$, and feed them to three speaker-specific GRUs (sGRU).

$$\begin{aligned}
[\hat{h}_1^e, \hat{h}_3^e] &= sGRU_1(e_1, e_3)\\
[\hat{h}_2^e, \hat{h}_5^e] &= sGRU_2(e_2, e_5)\\
[\hat{h}_4^e] &= sGRU_3(e_4)
\end{aligned}$$

Finally, we combine the hidden representations of GRUs by arranging them in the dialogue order for further processing, i.e., $\hat{H} = [\hat{h}_1^e, \hat{h}_2^e, \hat{h}_3^e, \hat{h}_4^e, \hat{h}_5^e]$.

Global Speaker Sequence (GSS). To explicitly capture the speaker information, their reactions with respect to other speakers, and their relationships, we also propose to encode the speaker sequence in TGIF. To capture the different reactions of a speaker with respect to the utterance of other speakers, we capture the speaker sequence by employing another Transformer encoder, T_s, which takes as input the sequence of speakers where each speaker is represented by a one-hot encoded vector. Each speaker goes through the Transformer encoder to give a speaker sequence aware representation, h_i^s, i.e. $\forall s_i, h_i^s = T_s(s_i)$. After this, we have a speaker sequence aware representation for each speaker, h_i^s, in the dialogue.

Fusion. We fuse the outputs of the above four submodules in two steps. In the first step, we combine the dialogue-level utterance and speaker sequence to obtain a global view of the conversation through a fully-connected layer. In parallel, we combine the dialogue and speaker-level emotion dynamics to get the essence of the flow of emotions in the conversation. Subsequently, in the second step, we concatenate the two representations. As the effect of an utterance on the final emotion changes with the change of the target utterance, we append the target representation to each utterance before feeding it to the output layer for prediction. We can justify the appending operation through the example shown in Table 3.5. It can be observed that utterance u_2 is present in both instances; however, it is the trigger only in the first instance. Moreover, in Figure 3.2b, the same trigger utterance u_3 resulted in an emotion flip from *neutral* to *sadness* in the first instance, while it causes the speaker's emotion change to *surprise* from *joy* for the second instance. Finally, we apply gradient masking for the improbable instigators.

3.5 Experiments and Results

We evaluate our models on MELD-FR and MELD-Ifor the task of trigger identification and instigator recognition, respectively. We will tackle both the models – EFR-TX and TGIF in different subsections in this section.

3.5.1 Evaluating EFR-TX

Figure 3.4 shows the distribution of triggers based on their distance from the target utterance. We notice that most of the triggers are the utterances that are spoken just before the target. This phenomenon corresponds to the natural conversations where an emotion-flip occurs immediately after a trigger statement is said. However, there are cases where the trigger lies beyond the last utterance of the target speaker. After analyzing the distribution carefully, we restrict the context_size = 5 for the experiments involving instance-level EFR classification.

Baseline Methods We employ a set of following baseline methods for a comparative study:

- **CMN** (20)**:** It utilizes memory networks to store the speaker-level contextual history within a dialogue. The authors showed that maintaining the conversational history in a memory helped CMN in predicting emotions more precisely. They also used these memories in capturing inter-speaker dependencies.
- **ICON** (115)**:** It maintains a memory network to preserve the interaction between the *self* and *inter-speaker* influences in dyadic conversations. It models this interaction into the global memory in a hierarchical way. Finally, the memory is used as a contextual summary which aid in predicting the emotional labels.
- **DGCN** (116)**:** It models the inter-speaker dynamics in a dialogue via a graph convolutional network. This work also leverages the self and inter-speaker dependencies of the participants for modeling conversations. By using graphs, the authors claim to have modeled context propagation in an efficient way.
- **AGHMN** (117)**:** It incorporates an attention GRU mechanism that controls the flow of information through a modified GRU cell based on the attention-weights, computed over the historical utterances in a dialogue.
- **Pointer Network** (118) **:** They are often used to generate output sequence when the length of output sequence depends on the length of the input sequence. Pointer networks have been applied to solve various combinatorial optimization and search problems such as Convex hull, and travelling salesman problem. Here, we use it to map our input sequence of utterances of a dialogue into a sequence of emotions or triggers.

These baselines are readily suitable for ERC, and a few of them reported their performance on the MELD dataset. On the other hand, by definition, for each utterance u_i, EFR aims to predict a classification (trigger/non-trigger) label for each of the previous utterances ($u_1, ..., u_i$), i.e., the model has to predict a vector of labels of length i. Since EFR is a new task and has no direct baseline model, we extend the above ERC baselines to predict a vector of labels. We augment the output layer with i independent softmax functions corresponding to each contextual utterance to achieve this. We keep the rest of the architecture as the original.

Results

The tasks of EFR and ERC are highly correlated and thus, in this section, we study the results obtained from EFR and its' impact on the ERC task. In this section, we present our comparitive study for EFR and ERC task by using the EFR-TX and the ERC-MMN(c.f. Chapter 2). We analyse the tasks in standalone as well as pipeline fashion. More details can be found below.

System	ERC (F1)								EFR (F1)
	Dg	Jy	Sr	An	Fr	Ne	Sa	W-Avg	Trigger
CMN†	0.0	48.6	**54.0**	33.7	8.6	**75.9**	19.9	51.7	37.5
ICON†	0.0	36.8	45.5	37.0	0.0	69.6	11.0	50.1	37.3
DGCN†	0.0	48.1	52.9	31.6	4.5	75.8	15.5	51.8	52.9
DGCN$^\dagger_{\text{multi}}$	0.0	39.2	43.7	37.1	0.0	71.7	12.1	51.1	53.0
AGHMN†	0.0	40.1	43.1	11.7	0.0	63.0	25.0	44.2	52.3
Pointer Network†	3.0	15.1	17.0	13.1	0.0	63.2	7.0	35.1	49.0
(ERC/EFR)-MMN	**20.2**	**48.7**	50.4	42.9	9.8	71.9	29.6	**55.7**	33.4
(ERC/EFR)-TX	0.0	4.0	5.0	1.9	0.0	61.2	0.0	29.5	44.8
EFR-ERC$_{\text{multi}}$	18.8	48.6	49.3	**43.7**	**11.2**	72.1	**32.0**	**55.7**	34.8
ERC^{True} →EFR	-	-	-	-	-	-	-	-	**53.9**

Table 3.8: Comparative analysis for ERC and EFR. (ERC/EFR)-MMN represents ERC-MMN for the emotion recognition task and EFR-MMN for the emotion-flip reasoning task (Dg: disgust, Jy: joy, Sr: surprise, An: anger, Fr: fear, Ne: neutral, Sa: sadness). †Performance on the MELD-FR dataset.

- **Single-task Learning Framework:** In this setup, both tasks are trained and evaluated separately. We evaluate two tasks on both Transformer-based (TX) and masked memory network-based (MMN) architectures. Table 3.8 summarizes the results. The MMN based systems, i.e., ERC-MMN and EFR-MMN (jointly denoted as (ERC/EFR)-MMN in Table 3.8), obtain F1-scores of 55.78% and 33.42%, respectively. The modeling of EFR-MMN as utterance-level classification follows the same procedure adopted for the baselines of EFR (c.f. Baseline section). Though the MMN architecture yields moderate performance on ERC, it underperforms on the EFR task, possibly due to the way the task was modeled. This motivated us to model EFR as an instance-level classification, as mentioned in Table 3.7 and the methodology section. Subsequently, we train a Transformer-based architecture EFR-TX and obtain a 44.79% F1-score on the test set. The improvement of more than 11% in F1-score justifies our EFR modeling as instance-level classification. We argue that reasoning the flip requires the information of the emotional states of the speakers. To support this hypothesis, we propose the ERC^{True} →EFR architecture, where we provide the true emotion labels in the EFR-TX architecture. The results obtained from this model support our hypothesis as we obtain a 53.9% F1-score. That is, an improvement of 9% over the EFR-TX model is observed. Following the success of EFR-TX, we also experiment with ERC-TX; however, the performance degrades significantly.
- **Multitask Learning Framework:** Since ERC and EFR depend on each other, we design a joint learning approach where two tasks are learned simultaneously in a unified manner. This setting follows a pipeline where the ERC-MMN model is extended to first detect the ERC labels, and if there is a flip observed in

a speaker's emotion, the EFR task is performed. Both the tasks share ERC-MMN till the penultimate layer. Subsequently, we add two parallel fully-connected layers – one for each task. We train the model by accumulating the losses incurred at the output layers of ERC and EFR and backpropagate employing Adam optimizer. We call this model EFR-ERC$_{\text{multi}}$. Unfortunately, EFR-ERC$_{\text{multi}}$ does not benefit much for ERC and a slight improvement of $\sim 1.4\%$ F1-score for EFR compared to EFR-MMN. However, the obtained multitask performance on EFR is below-par compared to EFR-TX. Similar to the earlier case, we attribute this performance drop to the different ways in which we model the tasks. We also perform multitasking on the best baseline, DGCN, where we observe that EFR slightly improves (from 52.9% to 53.0%), whereas we observe a 0.7% drop in ERC (51.8% to 51.1%).

- **Other Cascade Models:** Along with the experiments mentioned in Table 2.4, we also perform other experiments to play around with our architectures, specifically for the EFR task. Previously, we have explained two types of cascade models for the EFR task (ERC→EFR$_{cas}$ and ERC^{True} →EFR). Here, we will show two more cascade models that we tried for the EFR task.
  - **Early-fusion cascade:** In this setting, we introduce emotion labels in the input layer of our model. We concatenate the emotion representation (a 7-dimensional one-hot vector) with the utterance representation (a 768-dimensional BERT vector) and then feed it to the transformer-based network. The first row of Table 3.9 shows the results obtained using this model.
  - **Late-fusion cascade:** In this setting, we introduce emotion labels in the penultimate layer of our model. We concatenate the emotion representation (a 7-dimensional one-hot vector) with the representation obtained from the transformer encoder. We then feed this representation to the classification layers for classification. The second row of Table 3.9 shows the results obtained using this model.

In comparison to the single-task model `ERC-TX` with 44.79% F1, two of the baselines (CMN with 37.5% F1 and ICON with 37.3% F1) obtain lesser F1-scores, while the other three baselines, DGCN, AGHMN, and pointer network report improved results of 52.93%, 52.30%, and 49.0% F1-scores, respectively. However, all five baselines are outperformed by our final model (ERC^{True} →EFR) which reports an improvement of $1 - 17\%$ in F1-score for the trigger label.

	Early Fusion	***Late Fusion***
Trigger F1	35.1	51.5

Table 3.9: Experimental results for early- and late-fusion of emotion labels in the ERC^{True} →EFR model.

Generalizability

To analyze the performance of our model on an out-of-distribution generalization test set, we consider another dataset, IEMOCAP (119). It contains crowdsourced conversations revolving around 16 topics. For the construction of our test set, we randomly pick two conversations from each topic. We then create instances from these dialogues as illustrated in Table 3.7 and manually annotate them with EFR labels (inter-annotator agreement, $\alpha = 0.813$; $\alpha_{AB} = 0.818$, $\alpha_{AC} = 0.808$, and $\alpha_{BC} = 0.820$. To measure the overall agreement score, we take the average of these values, $\alpha = 0.813$.). Table 3.10 gives us a brief statistics of the IEMOCAP-FR dataset. We test our model trained on MELD-FR on IEMOCAP-FR and report the results in Table 3.11. For ERC, our model produces the best results. However, the results are significantly less than the results obtained on MELD-FR. This reduction can be attributed to the inherent differences in the dialogues present in the two datasets. IEMOCAP contains more than 50 utterances in a dialogue on average whereas MELD contains an average of 9 utterances per dialogue. Secondly, the emotion distribution between the two sets also differ in a major way. IEMOCAP does not contain any *disgust* emotion, and the *neutral* emotion is not as commonly present in it as it is in MELD-FR. Consequently, the task of emotion recognition becomes challenging for this dataset. On the other hand, our model and the baselines perform surprisingly well for the task of EFR, comparable to the EFR performance on MELD-FR. This performance can be attributed to the fact that even if the emotion distribution differs in IEMOCAP-FR, the distribution of triggers is still very similar to MELD-FR.

ERC								EFR		
Disgust	**Joy**	**Surprise**	**Anger**	**Fear**	**Neutral**	**Sadness**	**Total**	**#Diag with Flip**	**#Utt with Flip**	**#Triggers**
0	671	44	1413	25	552	407	3112	32	965	1388

Table 3.10: Statistics of the IEMOCAP-FR dataset.

System	**ERC (F1)**								**EFR (F1)**
	Dg	**Jy**	**Sr**	**An**	**Fr**	**Ne**	**Sa**	**W-Avg**	**Trigger**
CMN	0.0	7.1	0.0	**56.4**	0.0	2.1	2.3	28.2	35.6
ICON	0.0	14.2	0.0	49.7	0.0	9.1	8.0	28.4	36.1
DGCN	0.0	15.5	2.3	54.2	0.0	6.0	11.5	30.8	49.6
DGCN$_{\text{multi}}$	0.0	11.3	2.2	53.4	0.0	5.1	13.1	29.2	48.4
AGHMN	0.0	7.3	0.0	45.8	0.0	0.0	11.2	24.4	49.3
Pointer Network	0.0	12.4	0.0	32.2	0.0	2.6	6.0	18.1	44.7
(ERC/EFR)-MMN	0.0	**19.3**	**3.2**	52.7	0.0	**10.2**	**17.1**	**33.7**	31.2
(ERC/EFR)-TX	0.0	11.9	1.3	36.5	0.0	4.2	9.5	21.2	40.1
EFR-ERC$_{\text{multi}}$	0.0	17.5	2.2	51.5	0.0	8.3	**17.7**	31.4	32.8
ERC^{True} →EFR	-	-	-	-	-	-	-	-	**52.8**

Table 3.11: Comparative analysis for ERC and EFR. (ERC/EFR)-MMN represents ERC-MMN for the emotion recognition task and EFR-MMN for the emotion-flip reasoning task (Dg: disgust, Jy: joy, Sr: surprise, An: anger, Fr: fear, Ne: neutral, Sa: sadness). Trained on MELD-FR; Tested on IEMOCAP-FR dataset.

Error Analysis

This section presents both quantitative and qualitative analysis *w.r.t.* the confusion matrix and misclassification examples. We also supplement our analysis of the proposed systems with DGCN (the best baseline). Table 3.12 show the confusion matrix for EFR.

		Predicted	
		Non-Trigger	**Trigger**
Actual	**Non-Trigger**	2144	1359
	Trigger	226	926

(a) ERC^{True} →EFR

		Predicted	
		Non-Trigger	**Trigger**
Actual	**Non-Trigger**	2913	590
	Trigger	525	627

(b) Baseline: DGCN

Table 3.12: Confusion matrix of our best model and the best baseline for the EFR task.

The baseline DGCN reports comparable performance with ERC^{True} →EFR. However, while analysing the confusion matrices in Table 3.12, we observe that the number of *true-positives* considering the *trigger* class is much higher for ERC^{True} →EFR (926) than DGCN (627). Also, ERC^{True} →EFR reports lesser *false-negatives*. On the other hand, however, the *false-positives* are more, and due to which our proposed model fails to leverage the higher *true-positives* to a full extent and reports only 1% improvement over DGCN.

We also perform error analysis on the predictions of proposed systems. For illustration, we present one representative dialogue with its gold and predicted labels (ours and DGCN) for the EFR task. Table 3.13

shows a dialogue with two speakers, Ross and Rachel, and we observe an emotion-flip (*neutral*→*anger*) for Rachel in utterance u_5 considering her previous utterance u_2. For the target utterance u_5, actual trigger utterances are u_3 and u_4. Our proposed model, ERC^{True} →EFR, correctly identifies both triggers; however, it also misidentifies one utterance, i.e., u_5, as trigger. For the same dialogue, DGCN misclassifies one *trigger* utterance as *non-trigger* and one *non-trigger* utterance as *trigger*.

#	Speaker	Utterance		Actual		Prediction	
				Emotion (MELD)	**Trigger (MELD-FR)**	ERC^{True} →EFR	DGCN
u_1	Ross	Okay	Context Flip	*neutral*	*N-Trigger*	*N-Trigger*	*N-Trigger*
u_2	Rachel	Ross didn't you say that there was an elevator in here?		***neutral***	*N-Trigger*	*N-Trigger*	*N-Trigger*
u_3	Ross	Uhh yes I did but there isn't okay here we go!		*sadness*	*Trigger*	*Trigger*	*N-Trigger*
u_4	Ross	Okay go left left left		*surprise*	*Trigger*	*Trigger*	*Trigger*
u_5	Rachel	Okay y'know what there is no more left left!	Target	*anger*	*N-Trigger*	*Trigger*	*Trigger*

Table 3.13: Actual and predicted labels of triggers for a dialogue having five utterances (u_1, ..., u_5) from the test set. There is an emotion-flip for Rachel (*neutral*→*anger*) in u_5 and its triggers are u_3 and u_4. We mark them as triggers because Ross tricked her into believing that his apartment had an elevator and still acted like nothing happened, thus instigating an emotion-flip.

3.5.2 Evaluating TGIF

We perform experiments for both the granularity levels – coarse-grained and fine-grained. In the fine-grained setup, we observe a few instigator labels with a very low count. Since these labels are few in number, the model does not have sufficient evidence to learn a mapping from the input to such labels. Consequently, we compile another coarse-grained setup where we merge all instigator labels with count < 250 into a set, called '*other*'. As a result, in total, we have three setups – one fine-grained with 27 instigator labels) and two coarse-grained (definition-based and count-based) with 14 instigator labels each. In all three setups, we employ sigmoid neurons with focal loss (153) for multi-label classification. We select the traditional precision, recall, and F1-score as our metrics of choice thus ensuring that our evaluation is consistent with existing practices and establishes a universal benchmark.

Development Phase

Model	**Coarse-grained**		**Fine-grained**
	Defn-based	**Count-based**	
GUS	41.4	37.0	29.9
+ GES	41.9	37.1	31.5
+ GSS	42.1	37.8	32.1
+ SESS (TGIF)	**42.7**	**38.5**	**33.1**

Table 3.14: Results (W-F1) of fine-tuning on the development set. It shows the effect of each module when incorporated in TGIF. We obtain the best results when all four submodules are employed (last row).

To find the best configuration for TGIF, we investigate the effect of each module in the development phase. We start with the GUS module as the backbone network and subsequently introduce other modules (GES, GSS, and SESS) in an incremental fashion. Table 3.14 illustrates the results we obtain. Looking at the fine-grained setup, we notice a performance increase of 1.6% in weighted F1 when we add the GES module to the backbone model. This performance increase is coherent with our argument that the inclusion of emotional information will help the model in learning a better mapping function. Additionally, the incorporation of speaker-specific modules (GSS and SESS) gives a performance boost of 0.6% and 1.6%, respectively. We use the model consisting of all four submodules as our final architecture since it yields the best performance on the development set. After fine-tuning the hyperparameters during the development phase, we fix the configuration and evaluate TGIF on the test set.

Baselines and Comparative Study

Since the problem of instigator classification for EFR is novel, we adapt various related existing systems for comparison. Note that all these systems are recent state-of-the-arts and designed especially for the task of emotion recognition in conversation (ERC).

1. **DialogueGCN** (46): Uses GRUs and a graph convolution network (GCN) for emotion recognition by considering self and inter-speaker dependency.
2. **AGHMN** (47): Employs attention-based GRU to monitor information flow through hierarchical memory networks and calculate attention weights for classification.
3. **TL-ERC** (49): Adopts transfer learning from a dialogue generation model, leveraging its weights for emotion classification.
4. **DialogXL** (50): Customizes XLNet with utterance-level recurrence and dialogue-specific self-attention to recognize emotions in conversations.
5. **BERT** (110): Utilizes the transformer architecture (150) as an encoder stack for various NLP tasks.

Similar to TGIF, we perform instance-wise experiments with output masking for each baseline. That is, all the improbable instigators are masked. Moreover, since we provide emotion labels as input to our model, we do the same with the baselines.

Model	**Coarse-grained**						**Fine-grained**		
	Defn-based			**Count-based**					
	Pre	**Rec**	**F1**	**Pre**	**Rec**	**F1**	**Pre**	**Rec**	**F1**
AGHMN	7.6	20.4	11.07	8.5	25.4	12.73	15.1	17.6	16.3
TL-ERC	9.6	49.0	16.6	14.4	54.8	21.7	7.1	33.0	12.8
DGCN	12.5	67.0	19.8	18.5	70.2	27.5	10.5	67.2	17.5
DialogXL	7.3	37.5	12.22	8.8	43.7	14.64	9.8	34.2	15.3
BERT	18.3	62.9	27.2	17.5	59.2	26.3	14.8	55.1	21.7
TGIF	**24.3**	58.6	**31.6**	**28.3**	63.4	**37.5**	**26.5**	55.6	**33.3**

Table 3.15: Comparative results on coarse-grained and fine-grained instigators. All the metrics are weighted average over all instigator classes.

Table 3.15 shows that TGIF outperforms all baselines with reasonable margin across all setups. In the definition-based coarse-grained setup, we obtain $11.07\%, 16.6\%, 19.8\%$, 12.2%, and 27.2% weighted-F1 for AGHMN, TL-ERC, DGCN, DialogXL, and BERT respectively. In comparison, TGIF yields 31.6% W-F1 in the same setup – an increment of 4.4 points over the best performing baseline (BERT). We observe a similar trend for the count-based coarse-grained setup with TGIF and the best baseline (DGCN) reporting 37.5% and 27.5% W-F1, respectively – a difference of 10 points. In the fine-grained setup, the performance of the baselines (ranging between $12.8\% - 21.7\%$) are significantly inferior to TGIF (33.3%). It suggest that TGIF also accounts for the increase in instigator labels more efficiently than the existing baselines. TGIF beats all considered baselines in every setting but at the same time reports a weighted F1 score on the lower side indicating the difficulty of the problem statement.

Error Analysis

As we can observe from the distribution of EFR instigators (c.f. Figure 3.5), there is a significant label skewness. To inspect the learning of TGIF for individual labels, we analyze the results of top-3 (majority) and bottom-3 (minority) instigator labels w.r.t. the number of training instances in MELD-I. The top-

3 labels are *nervousness*, *awkwardness*, and *excitement* in the coarse-grained setup, and *annoyance*, *awkwardness*, and *excitement* in the fine-grained setup. Similarly, The bottom-3 labels in the coarse-grained setup are *shock*, *dazzle*, and *threat*, while instigators *nostalgia*, *pain*, and *boredom* are the three least occurring labels in the fine-grained setup.

Tables 3.16 report the results of TGIF and baselines for the majority and minority classes, respectively. As expected, the performance of each model for the majority classes is comparatively on the higher side of the spectrum than the performance on minority classes. Except for the instigators, *nostalgia*, *pain*, *boredom* in the fine-grained minority cases and *excitement* in the coarse-grained majority case, TGIF reports the best weighted-F1 for each case. The observed behaviour can be attributed to the fact that Transformer based architectures are data-hungry models, and thus they learn a better mapping for majority classes.

Model	Top-3 Majority						Bottom-3 Minority					
	Coarse-grained			Fine-grained			Coarse-grained			Fine-grained		
	Ner	Awk	Exc	Ann	Awk	Exc	Shk	Daz	Tht	Nos	Pain	Bor
AGHMN	11.2	10.1	8.6	17.8	12.8	15.3	3.9	4.2	3.2	**13.4**	7.2	**16.9**
TL-ERC	23.0	23.3	18.8	14.9	23.5	14.6	2.2	10.6	5.4	6.6	0.6	1.8
DGCN	28.9	28.4	28.3	24.8	26.9	23.2	10.5	9.8	6.0	0.0	**7.6**	2.1
DialogXL	12.2	14.7	12.6	13.9	11.9	11.1	2.1	3.6	4.2	6.4	5.2	2.9
BERT	36.0	26.8	**35.3**	38.9	26.8	36.1	10.7	2.5	0.0	3.2	2.0	0.0
TGIF	**37.8**	**35.7**	28.4	**53.5**	**35.8**	**56.7**	**35.1**	**18.6**	**9.8**	12.5	5.5	0.0

Table 3.16: Class-wise comparative analysis (F1-score) for the top-3 (majority) and bottom-3 (minority) classes. ⟨*Ner: Nervousness*, *Awk: Awkwardness*, *Exc: Excitement*, *Ann: Annoyance*, *Shk: Shock*, *Daz: Dazzle*, *Tht: Threat*, *Nos: Nostalgia*, *Bor: Boredom*⟩.

Qualitative Error Analysis. In order to perform qualitative error analysis, we take a sample dialogue from our test set and show the gold and predicted labels in Table 3.17 for the fine-grained setup. For the target utterance u_5, TGIF predicts *confusion* and *shock* instigators against the gold labels *confusion* and *curiosity* instigated by the trigger utterance u_4. Similarly, for the the trigger u_5, TGIF identifies two correct (*confusion* and *shock*) and one incorrect label (*curiosity*). An abstract view of the prediction suffices that the set of instigator labels for the emotion flip target u_5 (without regarding the triggers separately) is same as the set of gold labels. On the other hand, BERT (best baseline) commits many mistakes in both cases. It predicts one correct label for the trigger u_5 but no correct instigator for the trigger u_4. It can be observed that BERT gives precision scores of 0% for the first trigger while a precision of 50% is observed for the second trigger. Recall value also comes out to be 0% and 50% for the two triggers, respectively. In comparison, TGIF obtains moderate scores in both cases, i.e., recall $= 50.0\%$; precision $= 50.0\%$ in the first case and recall $= 100.0\%$; precision$= 66.7\%$ in the second case. A similar trend is observed for coarse-grained instigator labels. We note that, for a given target emotion, the BERT method consistently identifies the same instigators, giving little heed to the conversation context. In contrast, our approach takes both the target emotion and conversation context into account when identifying instigators, resulting in more accurate and nuanced predictions. Further, we show the zero-shot results of the proposed method in the supplementary.

	Speaker	Utterance	Emotion	Trigger	Instigator		
					Gold	Prediction	
						TGIF	BERT
u_1	Monica	Yeah, but without the costumes.	neutral	No	-	-	-
u_2	Phoebe	Oh.	neutral	No	-	-	-
u_3	Joey	And it's not fake, it's totally brutal.	neutral	No	-	-	-
u_4	Chandler	Yeah, it's two guys in a ring, and the rules are: They are no rules.	neutral	Yes	confusion, curiosity	confusion, shock	excitement, nervousness, shock
u_5	Monica	So you can like, bite, and pull people's hair and stuff?	surprise	Yes	confusion, shock	confusion, shock, curiosity	curiosity, shock

Table 3.17: Fine-grained analysis: Actual and predicted instigator labels for an EFR instance. TGIF predicts one and two correct instigator labels for the two trigger utterances, u_4 and u_5, respectively. In each case, it wrongly predicts one instigator. In contrast, BERT reports a high percentage of *false positives*.

Directionality of Triggers. In this work, we consider Ekman's emotion labels along with a label for no emotion (neutral). That is, we have six emotion labels, namely *disgust*, *joy*, *surprise*, *anger*, *fear*, and *sadness*. An emotion flip for a target speaker can occur between any two pair of emotions. In other words, we can have 42 possible emotion flips in a dialogue. Based on the source-target emotion pairs, we analyse the effect of directionality of emotion flips in MELD-I. We show the frequency of emotion flips with respect to the source-target emotion pairs in Table 3.6. Cell (i, j) in the table represents the number of flips in MELD-I where the source emotion is e_i, and the resultant or target emotion is e_j. As discussed in Section 3.3.2, there can be two types of emotion flips – *positive* and *negative*. Here, we see how the emotion flips frequency and instigators are dependant on the type of flips. Based on our ground-truth EFR labels, there are a total of 2612 positive emotion flips and 2818 negative emotion flips. Out of these flips, the flip *neutral* → *joy* is the most prominent positive emotion flip whereas the flip, *joy* → *neutral* is the most prominent negative emotion flip. It is crucial to note that *joy* → *neutral* considered as a negative flip because it involves a shift from a highly positive emotional state (joy) to a less positive or neutral state. While neutral isn't inherently negative, it represents a decrease in positive affect, which can be interpreted as a move towards a less positive or even somewhat negative emotional state, depending on the context and interpretation.

Apart from the emotion flips which have opposite polarities at both ends, we can also have intra-polarity flips. For instance, flips like *anger* → *fear* is a negative to negative emotion flip, while *joy* → *surprise* is a positive to positive emotion flip. We see, for the intra-polarity cases, that the flip *surprise* → *joy* and the flip *anger* → *sadness* are the most prominent intra-positive and intra-negative flips, respectively. We also observe that most of the flips that result in a negative emotion (*anger*, *disgust*, *fear*, *sadness*) originate from *joy*. On the other hand, the flips that result in a positive emotion (*joy*, *surprise*) originate from *neutral*. We also observe that, for positive emotion flips, the top-3 frequent instigators are *excitement*, *cheer*, or *impressed*. For negative emotion flip, *awkwardness*, *loss*, or *annoyance* are the more frequent instigators. In addition, we check the performance of our models on the two most prominent types of flip directions – positive to negative and negative to positive. We show these results in Table 3.18 and observe that positive to negative flips are better predicted by our model for all the classes of instigators. This can be attributed to the fact that our data contains more negative emotions, thus containing more negative instigators. Consequently, our model is able to learn those instigators in a better fashion. This result is encouraging as it is an indication that with more data, our model will be able to learn the instigators in a better way.

Type of Flip	**Coarse-grained**						**Fine-grained**		
	Defn-based			**Count-based**					
	Pre	**Rec**	**F1**	**Pre**	**Rec**	**F1**	**Pre**	**Rec**	**F1**
Negative to Positive	**27.3**	54.6	33.9	26.1	52.5	32.7	19.3	52.2	26.2
Positive to Negative	26.4	**61.2**	**35.1**	**28.0**	**59.1**	**35.4**	**32.2**	**58.1**	**38.8**

Table 3.18: Result analysis on directionality of flips for coarse-grained and fine-grained instigators. All the metrics are weighted average over all instigator classes.

Generalizability of TGIF

In order to emphasise the relevance of EFR and evaluate the generalizability of the proposed methodology, TGIF, we perform a zero-shot experiment. We consider IEMOCAP (1) which consists of emotion annotated conversations on 16 topics. We randomly sample 15 conversations to construct emotion flip instances, with triggers identified as shown in Table III of the main text. We then task TGIF and BERT, the best baseline, with predicting

	Correctness	Completeness	Prefered Instigator set
BERT	2.67	3.42	25%
TGIF	**3.21**	**3.44**	**75%**

Table 3.19: Human Evaluation Results on IEMOCAP (1) in a zero-shot setting. Scores are average across all evaluators.

the emotion flip instigators. After collecting the predictions, we asked 20 human annotators to rate them based on correctness, completeness, and preference (TGIF vs. BERT). The cumulative results in Table 6.16 indicate that while TGIF outperforms BERT, the latter is comparable in terms of completeness.

3.5.3 Evaluating LLMs

Similar to the evaluation conducted in Section 2.5.3, we assess the efficacy of Llama, a large language model, across the tasks and datasets outlined in this chapter. Our primary objective is to compare the performance of Llama against our top-performing systems, namely EFR-TX and TGIF. The results are summarized in Table 3.20. Notably, while Llama demonstrates proficiency in instigator identification against TGIF, its performance in trigger discovery falls short when compared with EFR-TX. This discrepancy can be attributed to the relatively sparse data available for trigger detection, which may not suffice for a large-scale model to learn meaningful insights. Consequently, a lighter model exhibits superior performance in this scenario.

F1-scores	EFR-TX	TGIF		
	53.9	31.6	37.5	33.3
Llama	51.4	**33.8**	**39.2**	**36.1**

Table 3.20: Performance of Llama when compared with our proposed methodologies for EFR.

3.6 Emotion Flip Reasoning in Hindi-English Code-mixed Conversation

So far in this chapter, we have focused on the emotion dynamics of monolingual English dialogues. However, emotions also play a pivotal role in Hindi-English code-mixed conversations. Consequently, we have organised a shared task in SemEval 2024[3] called EDiREF[4]. The EDiReF shared task is an amalgamation of three subtasks tasks- (i) ERC in Hindi-English code-mixed conversations, (ii) EFR in Hindi-English code-mixed conversations, and (iii) EFR in English conversations. For code-mixed tasks, we follow the same guidelines as English and curated an EFR dataset from our code-mixed conversation dataset, E-MaSaC. Specifically, we annotated $11,908$ utterances in 449 dialogues with eight emotion labels (we added '*contempt*' in addition to the six basic emotions and *netural*) and 7550 trigger utterances for 5873 emotion-flips. Similar to the English dataset, we acquired the service of experts who are native speakers of Hindi and well-versed in English. As a quality assurance, the Krippendorff alpha-reliability inter-annotator agreement (108) is computed as $\alpha = 0.853$. A brief statistics of code-mixed EFR dataset is shown in Table 3.21.

Split	Emotions								Total
	Disgust	Joy	Surprise	Anger	Fear	Neutral	Sadness	Contempt	
Train	127	1646	444	856	530	4091	572	549	8815
Dev	21	242	68	122	91	652	132	75	1403
Test	21	382	57	150	129	697	167	87	1690

(a) ERC – Hindi

Split	#D with Flip	#U with Flip	#Triggers
Train	344	4406	5565
Dev	47	686	959
Test	58	781	1026

(b) EFR – Hindi

Table 3.21: Statistics of the Code-mixed E-MaSaCdataset for ERC and EFR.

[3] https://semeval.github.io/SemEval2024/

[4] https://lcs2.in/SemEval2024-EDiReF/

3.7 Conclusion

This chapter addressed the critical need to delve deeper into the emotional dynamics of speakers within conversational dialogues. While identifying emotions in isolation was undoubtedly valuable, it fell short in providing a comprehensive understanding of the intricate and speaker-specific changes in emotions that transpired throughout a conversation. To bridge this gap in the emotional discourse analysis, we introduced a novel task of EFR. EFR is designed to unravel the triggers and instigators that underlay shifts or flips in a speaker's emotional state during a dialogue. These instigators could range from external factors like threats, which might have caused a shift from joy to anger, to a myriad of other emotional catalysts. To support the pursuit of EFR, we presented two meticulously crafted datasets, MELD-FR and MELD-I, each adorned with ground-truth labels for EFR triggers and instigators. These labels aligned with well-established principles in emotional psychology, lending credibility to our research. Furthermore, we harnessed the power of advanced neural architectures, namely the EFR-TX and TGIF, incorporating Transformer encoders and stacked GRUs to adeptly capture the rich nuances within dialogue context, speaker dynamics, and the evolving emotional sequences. The results of our rigorous evaluation spoke volumes, showcasing the state-of-the-art performance of EFR-TX and TGIF, which surpassed the effectiveness of five baseline models designed for this challenging task. Additionally, we proved the adaptability and robustness of our neural architectures in handling unseen datasets, thereby demonstrating their generalizability in a zero-shot setting. Additionally, we explore the task of EFR in the code-mixed setting by organising a shared task in SemEval 2024. Although the emotional comprehension of English and Hindi-English code-mixed conversations improves the understanding of dialogues, they do not provide a holistic view of the discourse. In order to gain a complete grasp of the conversation, we also need the perception of other affective cues, such as humour and sarcasm, which we explore in the subsequent part.

Part II

Sarcasm and Humour Analysis

4. Sarcasm and Humour Detection

Sarcasm detection and humour classification are inherently subtle problems, primarily due to their dependence on the contextual and non-verbal information. Furthermore, existing studies in these two topics are usually constrained in non-English languages such as Hindi, due to the unavailability of qualitative annotated datasets. In this chapter, we make two major contributions considering the above limitations: (1) we develop a Hindi-English code-mixed dataset, MaSaC, for the multi-modal sarcasm detection and humour classification in conversational dialogue, which to our knowledge is the first dataset of its kind; (2) we propose MSH-COMICS, a novel attention-rich neural architecture for the utterance classification. We learn efficient utterance representation utilizing a hierarchical attention mechanism that attends to a small portion of the input sentence at a time. Further, we incorporate dialogue-level contextual attention mechanism to leverage the dialogue history for the multi-modal classification. We perform extensive experiments for both the tasks by varying multi-modal inputs and various submodules of MSH-COMICS. We also conduct comparative analysis against existing approaches. We observe that MSH-COMICS attains superior performance over the existing models by >1 F1-score point for the sarcasm detection and 10 F1-score points in humour classification. We diagnose our model and perform thorough analysis of the results to understand the superiority and pitfalls. Further, we consider two monolingual datasets for humour and sarcasm detection in monolingual English using our proposed architecture and attain valuable improvements.

4.1 Introduction

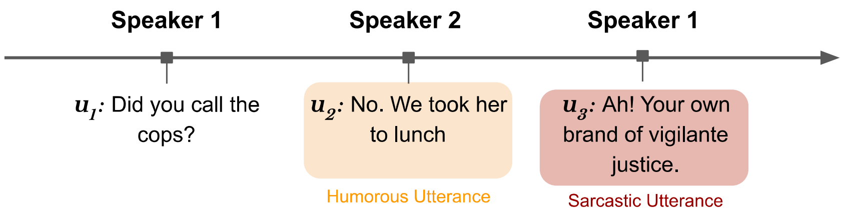

Speaker 1
Speaker 2
Speaker 1
u_1: Did you call the cops?
u_2: No. We took her to lunch
u_3: Ah! Your own brand of vigilante justice.
Humorous Utterance
Sarcastic Utterance

Figure 4.1: Example of sarcasm and humour identification in conversation.

Unlike traditional sentiment (154; 155; 156; 157) or emotion classification (158; 159; 75; 160; 161), sarcasm or humour detection in a standalone textual input (e.g., a tweet or a news headline) is a non-trivial task due to its below-the-surface semantics. Most of the time, the surface-level words carry sufficient cues in the text to detect the expressed sentiment or emotion. However, sarcastic or humorous inputs do not offer such simplistic information for classification. Instead, the expressed semantic information in sarcastic or humorous inputs often have dependency on the context of the text, and it is important to leverage the contextual information for the identification task as shows in Figure 4.1, where the previous utterances are required to figure out whether an utterance is sarcastic/humorous or not. Moreover, in many cases, the presence of multi-modal signals, such as visual expression, speech pattern, etc., provide auxiliary but crucial cues for sarcasm or humour detection. At times, they are the only cues that support a sarcastic/humorous expression. For example, it is extremely difficult (or nearly impossible) to detect sarcasm in the text '*Thanks for inviting me!*' without any context or other information. However, the same is less challenging if multi-modal signals accompany the text (e.g., a disgusting facial expression, gaze movement, or intensity/pitch of the voice while uttering the text) or the context (e.g., the text was preceded by a dispute/argument/insult).

A conversational dialogue records the exchange of utterances among two or more speakers in a time series fashion. Thus, it offers an excellent opportunity to study the sarcasm or humour in a context. A few previous attempts (162; 2; 163) on sarcasm classification involved multi-modal information in a conversation to leverage the context and extract the incongruity between the surface and expressed semantics. Similarly, many studies (57; 164) employed images and visual frames along with the text to detect humour. Surveys on multi-modal analysis (165; 166; 167; 168; 169; 170) reveal two prime objectives while handling multi-modal contents: (a) to leverage the distinct and diverse information offered by each modality, and (b) to reduce the effect of noise among the multi-modal information sources.

Usually, the solution to a natural language processing task handles only a single language. However, with the globalization of languages, many applications demand for solutions that can handle more than one language at a time. Thus, a new frontier of multi-lingual processing has emerged. India is a multi-lingual country, and a vast population are comfortable with more than one language. Their comfort is apparent in the regular usage of words from multiple languages to form a single sentence in both writing and speaking. For example, the text '*Sachin ne 21 years pehle apna debut match khela tha.*' ('*Sachin played his debut match twenty one years ago.*') has three English words (i.e., 'years', 'debut', and 'match') and one named-entity (i.e., 'Sachin'), while the rest of the words are part of romanized Hindi language. Similarly, it is common to switch languages for the consecutive sentences as well. For example, two sentences in Table 4.1 are in two different languages – not only their words are in different languages, but also they follow language-specific syntactic structure. These two variants are usually termed as the *code-mixed* and *code-switched* inputs, respectively.

	Code-mixed
Original:	*Sachin ne 21* **years** *pehle apna* **debut match** *khela tha.*
Translation:	Sachin played his debut match 21 years ago.

	Code-switched
Original:	*Agle hafte meri garmi ki chuttiyan shuru hone wali hain.* **I am planning to go to Europe during my vacation.**
Translation:	My summer vacation is starting next week. I am planning to go to Europe during my vacation.

Table 4.1: Examples for the code-mixed and code-switched inputs. Bold text represents English words and Italic texts signify Hindi words or named entity.

Though the code-mixed and code-switched inputs are natural in a multi-lingual culture, they offer a significant challenge in the automatic processing of such text. The foremost task in handling code-mixed input is the language identification of each word. Dictionary-based lookup is a trivial solution to identify language-specific words; however, the complexity escalates when a token (in transliterated form) is a valid word in more than one language. For example, the word '*main*' has the meaning '*important*' in English, while it also means '*I*' in Hindi. Once the language is identified for each word, literature suggests language-specific processing for the downstream tasks in a trivial setup. Recently, the quest of handling multi-lingual inputs in a deep neural network architecture has paved the way for the development of more sophisticated multi-lingual/cross-lingual word representation techniques (171; 172).

Most of the existing datasets for the multi-modal sarcasm and humour detection involve only monolingual data (primarily English). To explore the challenges of code-mixed scenarios, in this paper, we introduce MaSaC[1], a new multi-modal contextual sarcasm and humour classification dataset in English-Hindi code-mix environment. MaSaC comprises ∼1,200 multi-party dialogues extracted from a popular Indian television show '*Sarabhai vs. Sarabhai*'[2]. It contains ∼15,000 utterance exchanges (primarily in Hindi) among the speakers. We manually analyze all the utterances and mark the presence/absence of sarcasm and humour for each of them.

[1] MaSaC can be vaguely pronounced as *Mazaak* (Joke) in Hindi.

[2] https://www.imdb.com/title/tt1518542/

To evaluate MaSaC dataset, we propose MSH-COMICS[3], a multi-modal hierarchical attention framework for the utterance classification in conversational dialogues. At first, we encode the textual utterance representation using a hierarchy of localized attention over the tokens in a sentence. In the next step, we learn the modality-specific dialogue sequence using LSTM (173) layers. Further, to leverage the contextual information, we employ three attention mechanisms that learn the importance of preceding utterances with respect to each of the textual, acoustic, and textual+acoustic modalities. Since one of the prime concerns in multi-modal analysis is to counter the presence of noise among modalities, we employ a simple gating mechanism that aims to filter the noise in accordance with the interactions among the modalities. Finally, we utilize the filtered representations for the sarcasm and humour classification. For a complete evaluation, we also check the performance of our model on two multimodal monolingual English datasets – MUStARD (174) and MUMOR-EN (175), for the task of sarcasm detection and humour classification, respectively.

Experimental results suggest significant performance for both the sarcasm and humour classification tasks. We also evaluate MaSaC and the monolingual English datasets on the existing multi-modal contextual sentence classification systems. The comparative study reveals that MSH-COMICS yields superior performance compared to the baselines for both the tasks. The contributions of the current work are as follows:

- We develop MaSaC, a qualitative multi-modal dataset for the sarcasm detection and humour classification.
- We propose a novel architecture for the multi-modal contextual sentence classification.
- We provide strong baselines for the two tasks on the proposed dataset.
- We report detailed analysis of the experimental results and the reported errors.
- Through our developed MaSaC dataset, we offer an opportunity to the community to carry forward the research on the code-mixed environment in Indian context.

4.2 Related Work

In this section, we present a survey of the literature on the sarcasm detection and humour classification focusing on the following three dimensions – context, multi-modality, and Indian languages.

Sarcasm Detection: Sarcasm detection is an interesting as well as a challenging task. It has gained significant attention in the last few years (58; 176; 59; 177; 178; 179). Earlier work on sarcasm detection involved investigation on the lexical aspects of the text expressing sarcasm (58). More specifically, the authors studied the influence of adjectives, adverbs, interjections, and punctuation marks in sarcasm detection, and showed that their presence have positive correlation (though small) with the sarcastic text. Tsur et al. (59) proposed a semi-supervised approach for sarcasm discovery in Amazon product reviews. The authors employed punctuation and pattern-based features to classify the unseen samples using a kNN classifier. A similar study on tweet was proposed in (176). Other works claimed the presence of sentiment shift or the contextual incongruity to be an important factor in accurate sarcasm prediction (177). Son et al. (180) proposed a hybrid Bi-LSTM and CNN based neural architecture for the sarcasm detection.

Most of the above studies involve sarcasm discovery in the standalone input - which are reasonably adequate for the sentence with explicit sarcastic clues. However, for the implicit case, more often than not, the context in which the sarcastic statement was uttered is of utmost importance (181; 3; 182). Joshi et al. (181) exploited the historical tweets of a user to predict sarcasm in his/her tweet. They investigated the sentiment incongruity in the current and historical tweets, and proposed it to be a strong clue in sarcasm

[3] MSH-COMICS is short for **M**ulti-modal **S**arcasm Detection and **H**umor Classification in **CO**de-**MI**xed **C**onversation**S**.

detection. In another work, Ghosh et al. (183) employed an attention-based recurrent model to identify sarcasm in the presence of a context. The authors trained two separate LSTMs-with-attention for the two inputs (i.e., sentence and context), and subsequently, combined their hidden representations during the prediction. The availability of context was also leveraged by (3). The authors learned a CNN-BiLSTM based hybrid model to exploit the contextual clues for sarcasm detection. Additionally, they investigated the psychological dimensions of the user in sarcasm discovery using 11 emotional states (e.g., *upbeat*, *worried*, *angry*, *depressed*, etc.).

Although a significant number of studies on sarcasm detection have been conducted in English, only a handful attempts have been made in Hindi or other Indian languages (184; 185). One of the prime reasons for limited works is the absence of sufficient dataset on these languages. Bharti et al. (185) developed a sarcasm dataset of 2,000 Hindi tweets. For the baseline evaluation, they employed a rule-based approach that classifies a tweet as sarcastic if it contains more positive words than the negative words, and vice-versa. In another work, Swami et al. (184) collected and annotated more than 5,000 Hindi-English code-mixed tweets. They extracted n-gram and various Twitter-specific features to learn SVM and Random Forest classifiers. Though the dataset proposed by Swami et al. (184) and MaSaC involve Hindi-English code-mixed inputs, they differ on the contextual dimensions, i.e., the instances in their dataset are standalone and do not have any context associated with them, whereas, the sarcastic instances in MaSaC are a part of the conversational dialog. Moreover, MaSaC also includes multi-modal information for each dialog.

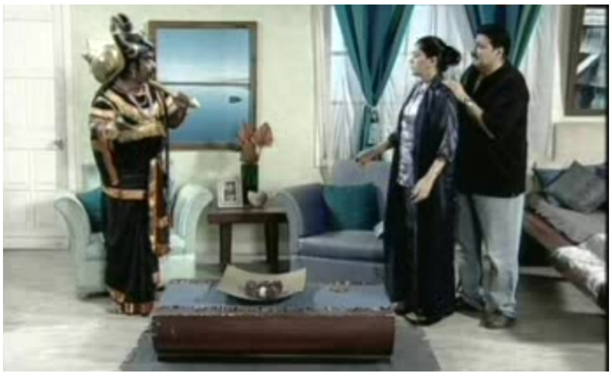
Recently, the focus on sarcasm detection has shifted from the text-based uni-modal analysis to the multi-modal analysis (162; 2). Cai et al. (162) proposed a hierarchical fusion model to identify the presence of sarcasm in an image in the pretext of its caption. The authors exploited the incongruity in the semantics of the two modalities as the signals of sarcasm. Another application of the multi-modal sarcasm detection is in the conversational dialog system. During the conversation, it is crucial for a dialog agent to be aware of the sarcastic utterances and respond accordingly. Castro et al. (2) developed a multi-speaker conversational dataset for the sarcasm detection. For each sarcastic utterance in the dialog, the authors identified a few previous utterances as the context for sarcasm. The dataset developed in the current work is on the similar line except two major differences: (a) MaSaC contains Hindi-English code-mixed utterances, which is the first dataset of its kind; and (b) instead of defining the explicit context, we let the model learn the appropriate context during training.

Humour Detection: Like sarcasm detection, computational humour analysis is a fascinating but subtle task in the domain of natural language processing. Recent literature suggests that contextual information plays an important role in computational humour detection (164; 57). However, due to the complexity in processing the contextual information, many of the earlier studies aim to identify humorous contents in standalone text without consulting the context (54; 186; 55; 56). Their prime inputs are *one-liners* or *punchlines* - which usually have rich comic or rhetoric content to attract someone's attention. Though the strategy of detecting humour in standalone texts seem appealing, often the absence of context makes it extremely difficult (even for humans) to interpret the humorous content. Moreover, the textual form of the humorous contents are complemented with other crucial non-verbal signals such as animated voice, impersonation, funny facial expression, etc. This difference in acoustic features between humorous and non-humorous utterances is validated by Amruta et al. (187) Many researchers have exploited these meta-data for humour classification (57; 164; 188). Hasan et al. (57) extended humour classification in punchlines by considering both the contextual and multi-modal information. The authors utilized Transformer's (189) encoder architecture to model the contextual information in addition to the memory fusion network (190) for combining the multi-modal signals. Bertero and Fung (164) relied on the text and acoustic features for contextual humour classification. Dario et al. (188) treated the humour classification task as sequence labelling and employed conditional random field to get the output. In the context of

Indian languages, the study on humour classification, like any other NLP task, is limited. To the best of our knowledge, Khandelwal et al. (191) is one of the first studies that involve humour classification in Hindi-English code-mixed language. They developed a dataset of ∼3,500 tweets with almost equal number of humorous and non-humorous tweets. The authors bench-marked the dataset on SVM classifier using bag-of-word features. Sane et al. (192) improved the state-of-the-art on the same dataset using neural models. In comparison with MaSaC, the dataset of Kandelwal et al. (191) lacks both the contextual as well as multi-modal information. Furthermore, MaSaC has significantly more number of instances, and annotations for two tasks, i.e., sarcasm and humour detection.

Problem Statement. Sarcasm is defined as an expression meant to criticize, taunt, or hurt someone's feeling in a sober and explicitly non-disrespectful manner. On the other hand, humorous statements aim to incite amusing or comic feelings with the intention to make their audience laugh. A light-hearted sarcastic statement which does not offend the target can be interpreted as humorous. However, it is important to note that all sarcastic statements may not be amusing, whereas a humorous expression is always intended to amuse the listeners.

In the current work, our objective is to identify all the instances of sarcastic or humorous utterances in a multi-speaker conversational dialog. Given a sequence of utterance $U = (u_1, u_2, ..., u_n)$ in a dialog video, we wish to classify each utterance into – (i) *sarcastic* or *non-sarcastic*, and (ii) *humorous* or *non-humorous*. Each utterance u_i has multiple representations corresponding to the available modalities, i.e., visual frames of the utterance u_i^V, acoustic signals of the utterance u_i^A, and the utterance transcripts u_i^T. In our study, we do not account for the visual frames while learning the model. A valid explanation for leaving out the visual modality is due to the presence of multiple actors in a frame, and most of them do not offer any constructive assistance to the model. We argue that the inclusion of the visual frames in the model would defile the learning process by attending to irrelevant content (or noise).To support our claim, in Figure 4.2, we show one of many such scenarios. Therefore, we employ only the textual and acoustic features in our model.



Kyun? Mene to apna bhensa building ke bahar park kiya hai... [Humour]
(Why? I have parked my bull outside the building...)

Figure 4.2: An example frame highlighting the irrelevant visual content considering the humour (or sarcasm) prediction, and the model may defile the learning process by attending to irrelevant contents (or noise).

4.3 Dataset

4.3.1 Monolingual English

For the task of monolingual English humour classification and sarcasm detection, we use the prevalent MUMOR-EN (175) and MUStARD (174) datasets. We explain them below.

- **MUMOR-EN** (175): MUMOR-EN is constructed based on the MELD dataset (85). The authors of MUMOR-EN discarded dialogues from MELD which contained less than three utterances in them since the purpose of the dataset is to recognize humour in long conversations rather than short texts. The MUMOR dataset contains humour, emotion, and sentiment labels for each utterance. The authors asked three annotators to watch the video clips with subtitles of each utterance, and let them decide whether this utterance is humorous or not and which kind of emotion it belongs to. For humour label, two categories were considered – humorous and non-humorous. The overall Fleiss' kappa score of humour annotation process was 0.81, which indicates a decent agreement among annotators.

- **MUStARD** (174): Similar to MUMOR-EN, MUStARD also contains conversations from popular English sitcoms including Friends[4] and Big Bang Theory[5]. MUStARD contains multimodal cues along with the textual dialogue utterances. Each utterance in the data is marked with a binary label indicating the presence of sarcasm in the utterance. The dataset consists of 690 samples, where 345 samples are sarcastic and 345 samples are non-sarcastic in nature.

4.3.2 Code-mixed Hindi-English

Our multi-modal sarcasm and humour classification dataset is based on the video clips of the popular Indian comedy TV show 'Sarabhai vs. Sarabhai'[6]. The show resolves around the day-to-day life of five family members, namely, Indravardan (aka Indu), Maya, Saahil, Monisha, and Roshesh, with a few infrequent characters. Each scene of the show involves conversation among two or more speakers, and based on the speaker, we split the conversation into utterances. In all, we extract more than 15K utterances from 400 scenes spread across 50 episodes. We refer to the conversation (or sequence of utterances) in each scene as a standalone dialogue. For each utterance in the dialogue, we assign appropriate sarcasm (*sarcastic* or *non-sarcastic*) and humour (*humorous* or *non-humorous*) labels. The context for any utterance is restricted to the conversation in the current dialogue only. We employed three annotators, all fluent in English and Hindi with age between 20-35 years, for assigning sarcasm and humour labels to each utterance and aggregate the annotations using majority voting. We calculate the Cohen Kappa inter-rater agreement score for the annotations. The average score for humour classification is 0.654, whereas for sarcasm detection it is 0.681 signifying a sufficient agreement.

Data Preprocessing. The multi-modal information extraction from a comedy video poses two primary challenges: (1) alignment of the multi-modal signals, and (2) laughter removal from the acoustic signal. For the alignment, we mark the boundary of each utterance on the time spectrum for mapping the corresponding speech and visual frames. This was performed by detecting a prolonged silence in the video, and subsequently, discarding the silence portions on the time spectrum. As a consequence, we obtain the boundary for each utterance in the dialogue. Subsequently, we extract the speech signals employing the Google Speech API-based automatic speech recognition tool, called Gnani.ai.[7]

Like many other comedy shows, our input video also contains audiences' laugh as they react to the scene. It is a popular practice to highlight the comic or humorous situation in the video. Since one of our target tasks is humour classification, we remove laughter from the audio signal to avoid the model to overfit on the audience laugh. We employ open source Audacity[8] tool for the laughter and background noise removal. Audacity's algorithm works as follows — It initially identifies different sound bands corresponding to the laughter frequency range. It then suppresses the audio frequency signals above the threshold of the laughter sample frequency. Then a sampling function is applied to smooth the suppressed audio, resulting in an audio file with reduced laughter frequency bands.

[4] https://www.imdb.com/title/tt0108778/

[5] https://www.imdb.com/title/tt0898266/

[6] https://www.imdb.com/title/tt1518542/

[7] Gnani.ai

[8] https://github.com/audacity/audacity/blob/master/src/effects/NoiseReduction.cpp

Data Statistics. In Table 4.3, we list the dataset statistics along with the annotated class label counts. We split the dataset into train and test set with $1,100$ and 90 dialogues, respectively. Furthermore, we use 10% of train set as the validation set during experiments. Out of $14,000$ utterances in the train set, the number of sarcastic and humorous utterances are $2,748$ and $5,054$, respectively. Similarly, the test set comprises 391 sarcastic and 740 humorous utterances. Table 4.3 also lists the word distribution for the Hindi-English code-mixed input. MaSaC consists of $\sim$36,000 Hindi and $\sim$3,000 English words.

	Speakers (Characters)				
	Indravardan	Maya	Saahil	Monisha	Roshesh
Sarcastic	1383	826	692	115	123
Humorous	2391	1733	733	769	168

Table 4.2: Speaker-wise sarcastic and humorous utterance distribution in MaSaC.

We also present the speaker-wise sarcastic and humorous statistics in Table 4.2. Out of the five speakers, one speaker stands out in both sarcastic and humorous utterances, i.e., Indravardan, followed by Maya, Saahil, and others.

	#dialogue	#Utterance	#Speaker/dialogue	Utterance Len		Vocab		Labels	
				Avg	Max	Hindi	English	Sarcastic	Humorous
Train	1100	14000	3 (Avg)	20	128	27574	2462	2748	5054
Test	90	1576				8664	669	391	740

Table 4.3: Statistics of MaSaC for code-mixed sarcasm and humour classification. For each utterance, we extract the visual, acoustic, and transcript of the dialogue.

Feature Extraction. We employ pre-trained FastText multilingual word embedding model (171; 172) and Librosa (193) tool for the textual and acoustic representations, respectively. For each token in the utterance, we extract a 300-dimensional word vector. Following the uncased version, we obtain embedding coverage for more than 90% of vocabulary words. For the acoustic representation, we use Librosa (193) tool to extract the acoustic features for each frame – we extract the maximum possible (128) MFCCs (Mel-frequency cepstral coefficients) for every frame. To obtain the utterance-level acoustic representation, we follow the standard acoustic feature extraction technique (194; 195; 196) by utilizing a time distributed 1D convolution layer on top of MFCCs of all frames. We do not use visual signals in our models because the quality of visual frames present in our dataset was not good enough, and thus the features extracted from these frames were acting as noise.

4.4 Methodology

In this section, we describe MSH-COMICS, our proposed system for sarcasm and humour classification. Figure 4.3 presents a high-level architectural overview of MSH-COMICS. It takes a sequence of utterances (a dialogue) as input and produces a corresponding label for each utterance.

To learn the context of the dialogue, we employ two LSTMs on top of textual (u_i^T) and acoustic (u_i^A) representations of dimensions d^T and d^A, respectively.

$$h_i^A = LSTM^A(u_i^A, h_{i-1}^A) \tag{4.1}$$

$$h_i^T = LSTM^T(u_i^T, h_{i-1}^T) \tag{4.2}$$

where $h_i^A \in \mathbb{R}^{d^A}$ and $h_i^T \in \mathbb{R}^{d^T}$ are the learned hidden representations for acoustic and textual modalities, respectively. The textual representation (u_i^T) is computed through an application of the utterance-level hierarchical attention module (discussed later), whereas, the acoustic representation (u_i^A) is obtained through an audio processing tool, Librosa (193) (c.f. Section 4.3.2).

Dialogue-level contextual attention (*C-ATN*^{D}): Subsequently, we employ three separate attention modules that compute attention weights (i.e., α, β, and γ) of the contextual (preceding) utterances considering the acoustic pattern, textual pattern, and cross-modality pattern.

$$\alpha_i = \frac{\exp(h_i^A)}{\sum_{j=1}^{i} \exp(h_j^A)} \tag{4.3}$$

$$\beta_i = \frac{\exp(h_i^T)}{\sum_{j=1}^{i} \exp(h_j^T)} \tag{4.4}$$

$$\gamma_i = \frac{\exp(h_i)}{\sum_{j=1, X\in(A,T)}^{i} \exp(h_j^X)} \tag{4.5}$$

These attention weights signify the importance of contextual utterances $u_1, ..., u_{i-1}$ for the classification of utterance u_i. Therefore, we compute the mean of the contextual attended vectors for each hidden representation h_i. Further, we utilize the residual skip connection (197) to form the final attended representations $\hat{h}_i^A$, $\hat{h}_i^T$, and $\hat{h}_i^A T$ corresponding to the acoustic, textual, and cross-modal attention modules, respectively as follows:

$$\hat{h}_i^A = \sum_{k}^{i} \alpha_k h_k^A / i \oplus h_i^A \tag{4.6}$$

$$\hat{h}_i^T = \sum_{k}^{i} \beta_k h_k^T / i \oplus h_i^T \tag{4.7}$$

$$\hat{h}_i^{AT} = \sum_{k, X\in(A,T)}^{i} \gamma_k h_k^X / 2i \oplus h_i^A \oplus h_i^T \tag{4.8}$$

where $\oplus$ is the concatenation operator. Collectively, we term these three attention modules as the dialogue-level contextual attention module *C-ATN*^{D}, i.e., *C-ATN*$^D = [\hat{h}_i^A, \hat{h}_i^T, \hat{h}_i^{AT}]$. In the subsequent steps, we consume these representations for the final classification.

Filtering: Prior to feeding these representations to the fully-connected layers, we incorporate a noise filtering mechanism (198) to enhance the representation for each modality. The intuition behind the filtering mechanism is to learn the interaction among the available modalities, which has not been incorporated in the model so far, and subsequently, filter the noise in correspondence with the other modalities. We argue that the filtering mechanism provides assistance to the model to pass only the relevant features such that the filtered representations of different modalities can complement each other in retaining the diverse and distinct features. For each modality, we implement filtering as follows:

$$h'^A_i = tanh(\hat{h}_i^A) \cdot \sigma(\hat{h}_i^{AT}) \tag{4.9}$$

$$h'^T_i = tanh(\hat{h}_i^T) \cdot \sigma(\hat{h}_i^{AT}) \tag{4.10}$$

where $\sigma(\cdot)$ refers to the sigmoid function and is learned during the training. Since $\sigma(\cdot)$ lies in the range [0, 1], it controls the amount of information that can pass through the filter, i.e., a value close to 0 signifies extremely irrelevant information and is blocked, whereas, for a value approaching 1, all the information can be forwarded to the upper layers. Finally, we take the filtered representations along with the cross-modal attended vector for the final classification.

Hierarchical attention module *H-ATN*^{U}: One of the crucial aspects of a deep neural architecture for any natural language processing task is the efficient input representation. Literature suggests the availability of many techniques to obtain an efficient sentence vector from the word-level embeddings, e.g., mean of constituent word embeddings, the last time-step representation in a recurrent layer, etc. However, a significant challenge in such approaches is to reduce the effect of irrelevant words and to find relations among the far apart words in the sentence.

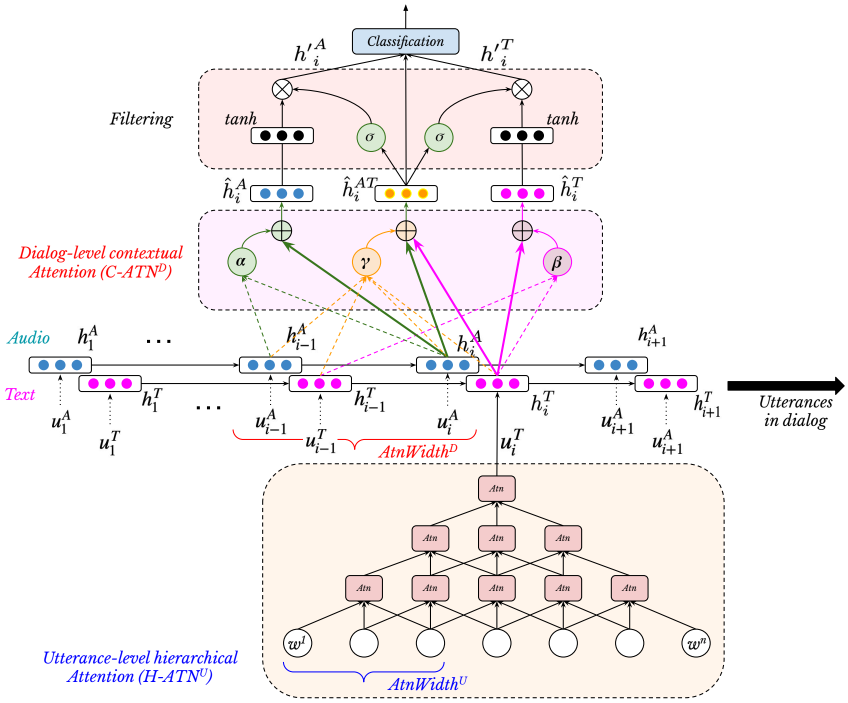


Figure 4.3: System architecture of MSH-COMICS. Each instance is a sequence of utterances in a conversational dialogue, and the classification is performed for each utterance. *H-ATN*^{U} computes efficient textual representation for each utterance in the dialogue. *C-ATN*^{D} learns attention weights of the contextual (preceding) utterances considering the acoustic (α), textual (β), and cross-modal (γ) patterns. Filtering mechanism reduces the effect of noise in the learned representation of individual modality.

In this paper, we propose a hierarchical attention module *H-ATN*^{U} to learn the significance of constituent words in the final sentence vector. We apply a series of localize attentions, each one attending to a small portion of the sentence. For example, $AtnWidth^U$=3 signifies that each attention mechanism attends to a sequence of three words only, and a context vector is obtained by taking a mean of the attended vectors followed by a linear layer with ReLU activation. As a consequence, we obtain N - $AtnWidth^U$ + 1 context vectors at the first hierarchical level l=1, where N is the number of words in a sentence. Similarly, we apply localized attentions at the second hierarchical level l+1, i.e., on N - $AtnWidth^U$ + 1 context vectors. Following this process, we compute localized attention for $\lceil \frac{N-1}{AtnWidth^U - 1} \rceil$ hierarchical levels, and at the final level, we obtain a single context vector representing the entire sentence. It is to be observed that, as we go higher in the hierarchy, *H-ATN*^{U} attends to a wider sequence of words, thus offers a mechanism to extract long-term relations. We formulate the utterance-level hierarchical attention mechanism in Algorithm 1.

4.5 Experiments and Results

We implemented our model in Python-based PyTorch deep learning library. For the evaluation, we compute precision, recall, F1-score, and accuracy for both the tasks. Though we compute and report both accuracy and F1-score for the sake of completeness, our preferred evaluation criteria is F1-score for the MaSaC dataset due to the unbalanced label distribution of classification labels (e.g., *sarcastic/non-*

Algorithm 1 Utterance-level Hierarchical Attention (*H-ATN*^{U})

```
procedure H-ATN$^U$($[w_1, ., w_N] = W, X =$ AtnWidth$^U$)
    for $k \in 1, ..., N$ do
        $CV_{(0,k)} = ReLU(w_l)$
    $M = ceiling(\,(N-1)/(X-1)\,)$
    for $l \in 1, ..., M$ do
        $Q = N - (l * X) + l$
        for $k \in 1, ..., Q$ do
            $\zeta_{l,k} = Attention(CV_{(l-1,k)}, ..., CV_{(l-1,k+X-1)})$
            $\phi_{l,k} = \left(\sum_i^X \zeta_{l,k} \cdot CV_{(l-1,k+i-1)}\right)\Big/X$
            $CV_{(l,k)} = ReLU(\phi_{l,k})$
    return $CV_{M,1}$
procedure $Attention(CV_{(1)}, ..., CV_{(X)})$
    for $i \in 1, ..., X$ do
        $\zeta_i = \frac{\exp(CV_{(i)})}{\sum_j^X \exp(CV_{(j)})}$
    return $\zeta$
```

sarcastic: $391/1185$) in it. We employ forward LSTM (173) to learn the contextual pattern of the dialogue, where each state of the recurrent layer learns a 128 dimensional hidden vector. We set *dropout*$= 40\%$ (199), *batch size*$=$ 32, and *ReLU* (200) as the activation function for the experiments. At the output, we employ *sigmoid* with *binary cross-entropy* to compute the loss. Subsequently, the computed loss is backpropagated utilizing the *Adam* (201) optimizer.

4.5.1 Results

For the utterance-level localized hierarchical attention mechanism, we experiment with varying attention widths *AtnWidth*^{U} in the range $[2, 5]$, and obtain *AtnWidth*^{U} = 3 to be the optimal value. Similarly, for the dialogue-level modality-specific attention mechanism, we observe that *AtnWidth*^{D} = 5 is best suited for the sarcasm and humour classification. We also experimented with visual features. We used the Affectiva API[9] and the GoogleNet Model (202) to obtain the visual expression features. Model *LSTM*(V_a) and *LSTM*(V_g) in Table 4.4 represent the case when only Affectiva and GoogleNet visual features are used for classification respectively for the MaSaC. The last row of the table illustrates the case when all three modalities of MaSaC are used in our model. It can be observed that the results using only visual features are far from satisfactory. This behavior can be attributed to the fact that the video frames present in MaSaChave low quality frames. The last row of each modality type illustrates the results obtained on the monolingual English datasets – MUStARD and MUMOR-EN, using the best performing system for the modality, for the tasks of sarcasm detection and humour classification, respectively. Table 4.4 reports the experimental results for both the tasks. It is to be noted that we train and evaluate all the models for both tasks separately.

Uni-modal evaluation – Acoustic

The first five rows of Table 4.4 list the results where we classify the utterance employing the acoustic signals only. We obtain F1-scores of 21.6% and 21.5% using *LSTM*(A) model for the sarcasm and humour classification, respectively for MaSaC. The possible explanation for low F1 score would be the absence

[9] https://github.com/cosanlab/affectiva-api-app

Modality	Model	Sarcasm Detection				Humor Classification			
		Pre	Rec	F1	Acc	Pre	Rec	F1	Acc
Acoustic (A)	LSTM(A)	0.419	0.146	0.216	0.738	0.475	0.139	0.215	0.523
	LSTM(*H-ATN*^{A})	0.383	0.129	0.193	0.537	0.445	0.171	0.247	0.463
	LSTM(A) + *C-ATN*^{D}	0.422	0.222	0.290	0.628	0.400	0.570	0.470	0.851
	LSTM(*H-ATN*^{A}) + *C-ATN*^{D}	0.254	0.207	0.228	0.537	0.273	0.597	0.375	0.619
	†‡LSTM(*H-ATN*^{A}) + *C-ATN*^{D}	0.710†	0.726†	0.718†	0.784†	0.743‡	0.697‡	0.719‡	0.762‡
Text (T)	LSTM(T_{avg})	0.569	0.558	0.563	0.669	0.753	0.617	0.678	0.867
	LSTM(T_{BERT})	0.646	0.524	0.579	0.774	0.707	0.667	0.687	0.717
	LSTM(*H-ATN*^{U})	0.862	0.573	0.688	0.871	0.711	0.724	0.717	0.735
	LSTM(*H-ATN*^{U}) + *C-ATN*^{D}	0.833	0.601	0.698	0.871	0.760	0.830	0.793	0.797
	†‡LSTM(*H-ATN*^{U}) + *C-ATN*^{D}	0.652†	0.634†	0.642†	0.714†	0.725‡	0.736‡	0.730‡	0.774‡
Visual (V)	LSTM(V_a)	0.113	0.084	0.096	0.178	0.161	0.105	0.127	0.226
	LSTM(V_g)	0.318	0.127	0.182	0.310	0.387	0.179	0.245	0.491
	†‡LSTM(V_g)	0.349†	0.334†	0.336†	0.378†	0.289‡	0.247‡	0.266‡	0.302‡
Text+Acoustic (T+A)	LSTM(A) + LSTM(T_{avg})	0.571	0.401	0.563	0.789	0.528	0.662	0.587	0.628
	LSTM(A) + LSTM(*H-ATN*^{U})	0.801	0.586	0.674	0.865	**0.809**	0.801	0.805	0.818
	LSTM(A) + LSTM(*H-ATN*^{U}) + *C-ATN*^{D}	**0.865**	0.555	0.676	0.868	0.755	0.832	0.797	0.807
	LSTM(A) + LSTM(*H-ATN*^{U}) + *C-ATN*^{D} + Filter	0.811	**0.636**	**0.711**	**0.873**	0.785	**0.858**	**0.820**	**0.823**
	†‡LSTM(A) + LSTM(*H-ATN*^{U}) + *C-ATN*^{D} + Filter	0.697†	0.682†	0.689†	0.731†	0.776‡	0.764‡	0.769‡	0.820‡
Text+Acoustic+Visual (T+A+V)	LSTM(V_a) + LSTM(A) + LSTM(*H-ATN*^{U}) + *C-ATN*^{D} + Filter	0.695	0.596	0.642	0.726	0.800	0.747	0.773	0.810
	LSTM(V_g) + LSTM(A) + LSTM(*H-ATN*^{U}) + *C-ATN*^{D} + Filter	0.748	0.571	0.647	0.774	0.762	0.775	0.768	0.792
	†‡LSTM(V_g) + LSTM(A) + LSTM(*H-ATN*^{U}) + *C-ATN*^{D} + Filter	0.729†	0.716†	0.722†	0.761†	0.794‡	0.786‡	0.789‡	0.831‡

Table 4.4: Experimental results for the sarcasm detection and humour classification. All models of each task are separately trained and evaluated. A → Acoustic features from Librosa; *H-ATN*^{U} → Utterance-level hierarchical attention mechanism over textual modality ; *H-ATN*^{A} → Utterance-level hierarchical attention mechanism over acoustic modality; *C-ATN*^{D} → Dialogue-level contextual attention mechanism; T_{avg} → Textual Utterance embedding computed as an average of the constituents word embeddings; V_a → Visual features from Affectiva; V_g → Visual features from GoogleNet. †‡ → Experiments on Monolingual English Dataset – † → Experiments on MUStARD, ‡ → Experiments on MUMOR-EN.

of any semantic entity in the representation – acoustic feature mainly captures the intensity, excitation mode, pitch, etc. Together with the textual feature, which contains semantic entities (words), the acoustic feature assists the model in leveraging the acoustic variations (e.g., excitement) for sarcasm and humour classification. Subsequently, we incorporate a dialogue-level contextual attention mechanism *C-ATN*^{D} over the LSTM layer (i.e., *LSTM*(A) + *C-ATN*^{D} model) and observe performance improvements of $\sim$8 and $\sim$26 points in F1-scores for the sarcasm detection and humour classification, respectively. The performance difference between the two models for both tasks is primarily due to the reduction in false negatives (and thus improvements in the recall values). Moreover, we credit the improvement to the attention module, which provides crucial assistance to the model in identifying the underrepresented sarcastic and humour classes. In other words, it helps the model to exploit the semantics of the relevant (attended) context in classifying the utterance as sarcastic or humorous. We also experiment with acoustic feature obtained from utterance-level hierarchical attention module, *H-ATN*^{A}. We observe a performance decrease of $\sim$7 and $\sim$10 in F1 scores for sarcasm and humour classification respectively when we use *H-ATN*^{A} instead of A. Thus, we continue with using A as our acoustic features. Another important observation suggests that the width of contextual attention (i.e., the number of contextual utterances considered in the attention computation) has an effect on the systems' performance. As we increase the attention width (*AtnWidth*^{D}) beyond five utterances, the performance of the systems begins to degrades. It suggests that the context of sarcasm or humour usually resides in the close proximity of the target utterance - which intuitively follows the real world as both sarcasm and humour lose their effect and relevancy, if delayed for a longer period. On the other hand, smaller *AtnWidth*^{D} does not offer sufficient context for the model to learn. Hence, for rest of the experiments, we choose *AtnWidth*^{D}=5. Additionally, for the monolingual English dataset, MUStARD, we obtain an F1-score of 71.8% when only audio is used as feature for detecting sarcasm, and an F1-score of 71.9% is obtained for the task of humour classification on the MUMOR-EN dataset.

Uni-modal evaluation – Textual

Similar to the acoustic modality, we also perform experiments with only the textual modality on code-mixed and English datasets. In total, we perform four variants, i.e., *LSTM*(T_{avg}), *LSTM*(*H-ATN*^{U}), *LSTM*(T_{BERT}) and *LSTM*(*H-ATN*^{U}) + *C-ATN*^{D} on MaSaC and *LSTM*(*H-ATN*^{U}) + *C-ATN*^{D} on English datasets. The first variant is a vanilla LSTM based classification model trained on the textual utterance embeddings - which is computed as the mean of FastText multilingual embeddings of constituent words, and is represented as T_{avg}. The second variant, *LSTM*(*H-ATN*^{U}), is similar to the first except that the textual utterance embeddings is computed utilizing the utterance-level hierarchical attention module. The third variant is also similar to the first two with the difference of type or utterance embedding used. In this variant, we experimented with BERT (110) to get the utterance embeddings. The fourth variant is an extension of the second where we also incorporate the dialogue-level contextual attention in classifying the utterances. We evaluate all these variants on both tasks and report the results in Table 4.4. For code-mixed sarcasm detection, we obtain F1-scores of 56.3%, 68.8%, and 69.8% for the three variants, respectively. Similarly, the models yield 67.8%, 71.7%, and 79.3% F1-scores for code-mixed humour classification. We can observe that the incorporation of utterance-level hierarchical and dialogue-level contextual attention mechanisms have positive effect on the overall performance in both tasks. Further, F1-scores of 64.2% and 73.0% are obtained for monolingual English sarcasm detection and humour classification, respectively.

Bi-modal evaluation – Textual + Acoustic

Next, we leverage the availability of both modalities (i.e., text and acoustic) for training MSH-COMICS. We learn two separate LSTMs for each modality, and at each step, we combine the two representations together utilizing the three dialogue-level contextual attention modules, i.e., text-specific contextual attention, acoustic-specific contextual attention, and cross-modal contextual attention on both text and acoustic signals. Further, we incorporate a gating mechanism to filter out the noise from the learned representations. Similar to the earlier case, we also experiment with the two variants of the textual representations, i.e., a mean vector T_{avg} and the vector computed by employing hierarchical attention *H-ATN*^{U}. The Text+Acoustic part of Table 4.4 report the ablation results for the different combinations of individual components - with the last row representing the complete model, as depicted in Figure 4.3.

The first model under the bi-modal inputs (i.e., *LSTM*(A)+*LSTM*(T_{avg}) model) yields 56.3% and 58.7% F1-scores for the sarcasm and humour classification, respectively. It can be observed that the simple addition of the acoustic information to the textual information does not effect the performance of the system (*LSTM*(A)+*LSTM*(T_{avg})) in a positive way, as compared to the system (*LSTM*(T_{avg})) with textual information only (c.f. row three of Table 4.4). We observe a performance drop of 9 points in F1-score in humour classification and no changes in case of sarcasm detection. Similarly, we see $\sim$2% drop in the accuracies values with the simple incorporation of acoustics signal for both tasks. This phenomenon can be attributed to the fact that the two modalities does not complement each other in the feature space and treat each other as the potential noise. We argue that the fusion of two modalities should be performed in an intelligent way such that they complement each other in the model training, and our incorporation of the filtering mechanism in the proposed model, indeed, assists the system to extract the complementary features only. The second model, *LSTM*(A) + *LSTM*(*H-ATN*^{U}), and the third model *LSTM*(A) + *LSTM*(*H-ATN*^{U}) + *C-ATN*^{D}) with bi-modal inputs, reflect the incorporation of utterance-level hierarchical and the dialogue-level contextual attention modules. However, similar to the previous case, acoustic signal does not have a positive influence on the results for the sarcasm detection. Finally, the performance on monolingual English shows complimentary performance when compared to the text only model. However, along our previous observations, the performance for the bimodal system is less than the scores obtained for acoustic only model due to potential noise while mixing information.

In the subsequent experiment, we evaluate our complete model on the two tasks, i.e., with the incorporation of filtering mechanism to dictate the complementary feature extraction. The proposed model yields the best F1-scores of 71.1% and 82.0% for the sarcasm and humour classification in the code-mixed setting. Moreover, it is also evident that the filtering mechanism leverages the acoustic signals in association with the textual information with a $\sim$2% jump in F1-scores compared to the text-based model. We also observe improvements in accuracy values for the two tasks as well. Further, F1-scores of 72.2% and 78.9% are obtained for monolingual English setting. In summary, we observe the following phenomena:

- As evident from the obtained results, the textual utterance embedding computed using hierarchical attention mechanism extracts richer features than the mean vector.
- The dialogue-level contextual attention module learns relevant context to conceive below-the-surface semantic for the target utterance.
- The filtering mechanism helps the system to extract relevant information from a modality in the proximity of others.

Joint-learning of Sarcasm and Humor

Since the two tasks are related in the problem space, i.e., both are classification tasks and both have dependencies on the context to extract the hidden semantics, we learn the sarcasm and humour classification tasks in a joint framework. The base architecture (till the filter module) for the joint-learning remains the same as earlier. We only add task-specific layers at the output, i.e., the architecture is extended with two branches corresponding to the two tasks after the filter module. During training, we compute loss at both the branches and propagate them back to the network. The results obtained using the joint-learning approach are reported in Table 4.5. We repeat the same set of experiments as in the case of separate learning (c.f. Table 4.4) for the MaSaC since it contains annotations for both, humour and sarcasm.

We can observe that the obtained results follow the same trend as in the case of separate learning. The usage of hierarchical attention, contextual attention, and the filtering modules help the system to obtain the F1-scores of 68.6% and 81.4% for the sarcasm detection and humour classification, respectively. However, excluding the filtering module, the system yields inferior F1-scores of 65.4% and 80.7%, respectively. Moreover, the incorporation of acoustic information without filtering mechanism also degrades the performance of the system. In comparison with the separate learning of two tasks, the joint-learning architecture yields lesser performance by 2.5 and 0.6 points in F1-scores; however, it requires lesser (approximately half) parameters to learn, and hence is about 50% less complex than the two separate models combined.

Comparative Analysis

We also perform comparative analysis by evaluating the existing systems on MaSaC. In particular, we evaluate MaSaC dataset on the following baseline models.

- **SVM** (203): We incorporate an SVM classifier on standalone utterances (without any context) as the baseline system. Depending on the textual representation, we evaluate two variants: a) on the average of the constituent word embeddings (T_{avg}), and b) on the embedding computed using the hierarchical attention module $\textit{H-ATN}^{U}$. For the acoustic signal, we utilize the raw feature representation as mentioned in Section 4.3.2.
- **MUStARD** (2): It is an SVM-based system that takes an utterance and its contextual utterances for the classification. For the evaluation, we define previous five utterances as the context and learn the sarcastic and humorous utterance classification. We experiment with the publicly available

Modality	Model	Sarcasm Detection				Humor Classification			
		Pre	Rec	F1	Acc	Pre	Rec	F1	Acc
Acoustic (A)	LSTM(A)	0.457	0.107	0.174	0.747	0.543	0.291	0.379	0.552
	LSTM(*H-ATN*^{A})	0.297	0.126	0.177	0.611	0.391	0.255	0.309	0.442
	LSTM(A) + *C-ATN*^{D}	0.520	0.263	0.350	0.757	0.565	0.381	0.455	0.572
	LSTM(*H-ATN*^{A}) + *C-ATN*^{D}	0.432	0.211	0.255	0.635	0.498	0.356	0.415	0.557
Text (T)	LSTM(T_{avg})	0.803	0.488	0.607	0.843	0.749	0.818	0.782	0.786
	LSTM(*H-ATN*^{U})	**0.843**	0.506	0.633	0.854	0.754	0.832	0.791	0.794
	LSTM(*H-ATN*^{U}) + *C-ATN*^{D}	0.695	**0.634**	0.663	0.840	0.745	0.877	0.806	0.801
Text+Acoustic (T+A)	LSTM(A) + LSTM(T_{avg})	0.675	0.537	0.598	0.821	0.718	0.764	0.740	0.748
	LSTM(A) + LSTM(*H-ATN*^{U})	0.799	0.540	0.644	0.852	**0.776**	0.824	0.799	0.806
	LSTM(A) + LSTM(*H-ATN*^{U}) + *C-ATN*^{D}	0.800	0.552	0.654	0.855	0.766	0.851	0.807	0.808
	LSTM(A) + LSTM(*H-ATN*^{U}) + *C-ATN*^{D} + Filter	0.785	0.609	**0.686**	**0.862**	0.756	**0.882**	0.814	**0.811**

Table 4.5: Experimental results for the *joint-learning* of sarcasm detection and humour classification. *H-ATN*^{U} $\rightarrow$ Utterance-level hierarchical attention mechanism over textual modality; *C-ATN*^{D} $\rightarrow$ Dialogue-level contextual attention mechanism. T_{avg}: Textual utterance embedding computed as an average of the constituents word embeddings.

	Systems	Sarcasm Detection				Humor Classification			
		Pre	Rec	F1	Acc	Pre	Rec	F1	Acc
Text	SVM (T_{avg})	0.170	0.332	0.225	0.618	0.284	0.475	0.356	0.492
	SVM (*H-ATN*^{U})	0.320	0.343	0.331	0.656	0.658	0.299	0.411	0.598
	MUStARD (2)	0.510	0.404	0.451	0.756	0.673	0.538	0.598	0.661
	Ghosh et al. (3)	0.595	0.432	0.500	0.786	0.648	0.518	0.576	0.644
	DialogRNN (4)	0.751	**0.604**	0.670	0.759	**0.730**	0.698	0.714	**0.764**
	MSH-COMICS	**0.833**	0.601	**0.698**	**0.871**	0.711	**0.724**	**0.759**	0.735
	†‡MUStARD (2)	0.609†	0.596†	0.598†	0.659†	0.674‡	0.696‡	0.685‡	0.713‡
	†‡**MSH-COMICS**	0.652†	0.634†	0.642†	0.714†	0.725‡	0.736‡	0.730‡	0.774‡
Text + Audio	SVM (T_{avg}+A)	0.217	0.281	0.245	0.571	0.389	0.370	0.379	0.431
	SVM (*H-ATN*^{U}+A)	0.274	0.384	0.320	0.595	0.429	0.397	0.413	0.469
	MUStARD (2)	0.520	0.458	0.487	0.761	0.692	0.546	0.610	0.673
	DialogRNN (4)	0.725	**0.690**	0.708	0.761	0.714	0.725	0.720	0.749
	MSH-COMICS	**0.853**	0.636	**0.711**	**0.873**	**0.785**	**0.858**	**0.820**	**0.823**
	†‡MUStARD (2)	0.639†	0.647†	0.643†	0.701†	0.735‡	0.719‡	0.727‡	0.763‡
	†‡**MSH-COMICS**	0.697†	0.682†	0.689†	0.731†	0.776‡	0.764‡	0.769‡	0.820‡

Table 4.6: Comparative study against existing approaches. MUStARD (2): SVM-based system with pre-defined context; Ghosh et al. (3): CNN-BiLSTM with pre-defined context; DialogRNN (4): Recurrent model for the classification of each utterance in the conversational dialogue. †‡ $\rightarrow$ Experiments on Monolingual English Dataset – † $\rightarrow$ Experiments on MUStARD, ‡ $\rightarrow$ Experiments on MUMOR-EN.

implementation[10] provided by Castro et al. (2).

- **Ghosh et al.** (3): The underlying architecture of Ghosh et al. (3) also incorporates the contextual information while classifying an utterance. The authors proposed a deep neural network architecture that models the contextual and target utterances using two separate CNN-BiLSTM layers. Further, the learned representations are combined in DNN for the classification[11]. Similar to the earlier case, for the evaluation, we define previous five utterances as context. The implementation of the model was adopted from (3)[12].
- **DialogRNN** (4): The DialogRNN (DRNN) (4) is one of the recent classification models capable of handling conversational dialogue. It was originally proposed for the ERC task; however, it is the closest approach considering our modeling of the two tasks, i.e., classifying each utterance in the

[10] `https://github.com/soujanyaporia/MUStARD`

[11] Gosh et al. (3) also employed authors' profile information for the modeling; however, we did not utilize such information during the evaluation.

[12] `https://github.com/AniSkywalker/SarcasmDetection`

conversational dialogue. The DRNN architecture encodes speaker-specific utterances independent of other speakers, and subsequently, incorporates each speaker-specific sequence to maintain the dialogue sequence. We utilize the implementation[13] of DRNN (4) for the evaluation.

In Table 4.6, we report the results of above comparative systems. For each comparative system, we evaluate on both uni-modal[14] *textual* and bi-modal *textual+acoustic* information. In text modality, SVM on T_{avg} reports mediocre F1-scores of 22.5% and 35.6% for the sarcasm and humour classification, respectively. In contrast, the same SVM classifier improves the performance of two tasks (11% and 6%, respectively) by utilizing the embeddings of hierarchical attention module. In comparison, the contextual models (MUStARD (2) and Ghosh et al. (3)) yield decent F1-scores of 45.1% and 50.0% in sarcasm detection. Similarly, the two comparative systems obtains 59.8% and 57.6% F1-scores for the humour classification. Finally, we evaluate DialogRNN (4) for both sarcasm and humour classification, and obtains the best comparative F1-scores of 67.0% and 71.4%, respectively. In comparison, for the same input (i.e., textual modality), our proposed system reports $\sim$3% and $\sim$4.5% improvement over the best comparative system.

We observe similar trends with the bi-modal *textual+acoustic* inputs for both the tasks under consideration. The SVM-based system records the least F1-scores of 24.5% and 37.9%, while DialogRNN (4) reports the best performance among the comparative systems with 70.8% and 72.0% F1-scores for the sarcasm and humour classification tasks, respectively. Comparison shows the superiority of the proposed system over the comparative system with >1 and 10 points improvement in the F1-scores. Finally, last two lines of text and text+audio represents the experiments on the monolingual English dataset.

4.5.2 Error Analysis

Though MSH-COMICS performs better than the existing systems, it did misclassify some utterances as well. In this section, we report our quantitative and qualitative analysis on the errors. At first, we analyze the system's performance in terms of confusion matrix, as depicted in Table 4.7.

	Sar	Non-Sar
Sar	249	142
Non-Sar	58	1127

	Hum	Non-Hum
Hum	635	105
Non-Hum	174	662

Table 4.7: Confusion matrix for MSH-COMICS.

Next, we choose a dialogue (consisting of 10 utterances) from the test set and present system's prediction in Table 4.8. It reports code-mixed utterances (with English translation), its speakers, and its actual and predicted labels for both sarcasm and humour classification tasks. Across 10 utterances in the dialogue, two of them are labeled as sarcastic in the gold set, while, the count of humorous utterances is 4. The dialogue in Table 4.8 also exhibits the contextual and/or multi-modal dependencies for an utterance to be labeled as sarcastic/humorous. For example, humorous utterances u_2 and u_3 do not convey any explicit textual semantics on their own; instead, they rely on the contextual utterances, i.e., u_1 for u_2 and $u_1 \& u_2$ for u_3. Moreover, utterance u_2 also depends on the multi-modal information, i.e., the excited voice of the speaker (Indu) along with the context signals the presence of humour in u_2. We also highlight the English words (blue colored text) along with its count for each utterance in the dialogue which constitute approximately 15% of the complete text. Out of these English words, some word plays crucial role in the identification of sarcasm/humour in the utterance. For example, the metaphorically used English word '*vegetable*' in utterance u_6 is the prime clue for the utterance to be identified as humorous.

[13] `https://github.com/declare-lab/conv-emotion`

[14] We do not report uni-modal *acoustic* results due to extremely poor performance.

#	Speaker	Utterance	Sarcasm		Humor	
			Actual	Pred	Actual	Pred
u_1	Maya:	*Viren ka phone aaya tha, Los Angeles se. mere popat kaka.* [Eng words: 1] I got a call from Viren from Los Angeles. My Popat uncle.	✗	✗	✗	✗
u_2	Indu:	*gaye kya?* [Eng words: 0] Did he die?	✗	✗	✓	✓
u_3	Maya:	*nahin. tayaari mein hain* [Eng words: 0] No, preparing for it.	✗	✗	✓	✗
u_4	Indu:	*come on come on, maya. don't cry. tum janti ho tum roti ho aur bhi acchi nahin lagti* [Eng words: 6] Come on Come on, Maya. Don't cry. You know when you cry, you look even worse.	✓	✓	✓	✓
u_5	Maya:	*Indravardan! Please! you know, viren keh raha tha ki unhone bilkul bistar pakad liya hai. chal phir bhi nahi sakte bechare* [Eng words: 3] Indravardan! Please! you know, Viren was saying he is completely bed-ridden, the miserable man can't even walk.	✗	✗	✗	✓
u_6	Indu:	*come on maya vo navve saal ke hain. is umr mein koi bhi insaan vegetable jaisa ho jaata hai* [Eng words: 3] Come on Maya. He is 90 years old. At his age, every one becomes miserable (seems like a vegetable).	✗	✓	✓	✓
u_7	Maya:	*I know, I know, darling. I mean, apni monisha ko hi dekh lo, itni choti umr mein bilkul vegetable jaisi ho gayi hai. din bhar sofa par padi rehti hai. dopahar ki t.v. serial dekhti rehti hai. sabzi bhi wahin lete lete kaatti hai. chai piti hai to glass uthakar phir kitchen nahin le rakhti. tel laga sir sofa ke cushions mein rakh deti hai* [Eng words: 15] I know, I know, darling. I mean, look at our Monisha, she looks so miserable at this young age. She spend her whole day on the sofa watching daytime T.V. serial. She chops vegetable while reclining there. She does not put the cup back in the kitchen after having a cup of tea. She puts her oily hair on sofa's cushion.	✓	✓	✗	✗
u_8	Indu:	*popat kaka. maya, hum popat kaka ki baat kar rahe the, na* [Eng words: 0] Popat uncle! Maya, were'nt we talking about Popat uncle?	✗	✗	✗	✗
u_9	Maya:	*haan, viren keh raha tha ki din bhar mujhe yaad karte rehte hain. maya, maya, maya ko bulao* [Eng words: 0] Yes, Viren was mentioning that he remembers me the whole day. Maya, Maya, ask Maya to come.	✗	✗	✗	✓
u_{10}	Indu:	*accha to kab jana ho raha hai los angeles tumhara?* [Eng words: 0] Great, when are you leaving for Los Angeles?	✗	✗	✗	✗

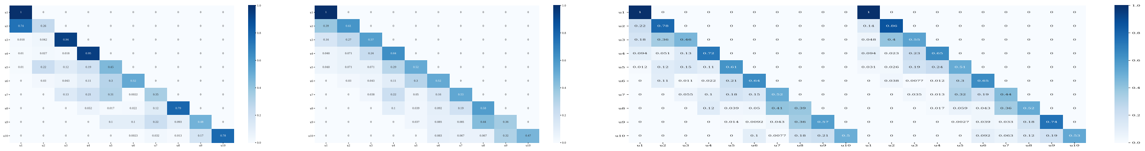
Table 4.8: Actual and predicted labels for sarcasm detection and humour classification for a dialogue having 10 utterances ($u_1, ..., u_{10}$) in MaSaC dataset. Blue-colored texts represent English words, while black-colored texts are either romanized Hindi or named entities. For sarcasm detection, MSH-COMICS yields 66% precision and 100% recall. Similarly, we obtain 60% precision and 75% recall for the humour classification.

On evaluation, our system predicts three utterances as sarcastic (i.e., u_4, u_7, and u_7) and 5 utterances as humorous (i.e., u_2, u_4, u_5, u_6, and u_9). In both cases, it makes some correct predictions as well as some incorrect predictions. For the sarcasm detection, our system yields *precision* of 66% (i.e., two out of three predictions are correct) and *recall* of 100% (i.e., both sarcastic utterances are correctly predicted). On the other hand, we obtain *precision* and *recall* of 60% and 75%, respectively, for the humour classification, i.e., we observe two *false-positives* and one *false-negatives* along with three *true-positives*.

Model	Data	Sarcasm Detection				Humor Classification			
		Prec	Rec	F1	Acc	Prec	Rec	F1	Acc
MSH-COMICS	*Set1*	**0.67**	**0.79**	**0.72**	**0.83**	**0.92**	0.91	**0.91**	**0.87**
	Set2	0.62	0.62	0.62	0.82	0.51	**0.96**	0.67	0.76

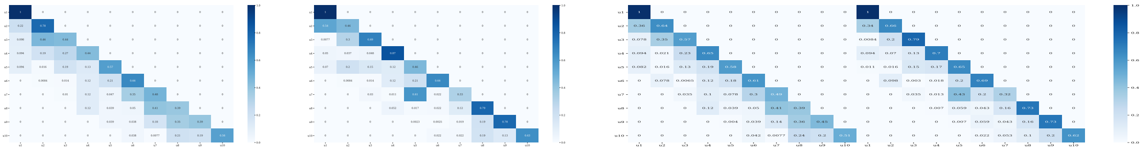
Table 4.9: Experimental results on sampled data from test set to analyze the effect of the extent of code-mixing in our model. *Set1* contains 100 utterances containing ∼18 English words (on avg) per sentence, whereas *Set2* contains 100 utterances with ∼1 English word per sentence.

We also analyse the effect of the level of code-mixing for both the tasks. In the given example, there is on an average one English word in every six words. For sarcasm classification, our model only returns one false positive. It is for the case where the utterance contains three English words out of the total 19 words, i.e., one English word for every 6.3 words. Whereas it gives correct predictions for utterances 4 and 7 which contains one English word for every three words and one English word for every 4.3 words, respectively. For the humour classification task, our model misclassifies for utterances u3, u5 and u9, having 0, 3 and 0 English words respectively. The ratio for English words to total words then turns out to be 0:4, 1:3 and 0:18. Whereas it predicts humour correctly for utterances u2, u4 and u6, having 0, 6 and 3 English words respectively. The ratio of English words to total being 0:2, 1:3 and 1:6.3 respectively. Looking at these results, we hypothesize that our model performs better for the utterances containing comparatively more English words. To validate this hypothesis, we sample two sets from our test set. Both the set has 100 utterances. One of the set contains the utterances having the most number of English words (∼18 English words per utterance) and another contains utterances containing the least number of English words (∼1 English word per utterance). We evaluate our final model on these two sets and report the result in Table 4.9. It can be easily seen from the table that the our model performs better when there are more English words in the utterance thus validating our hypothesis.



(a) Textual attention. (b) Acoustic attention. (c) Textual and acoustic cross-modal attention.

Figure 4.4: Humor Classification: Heatmap analysis of the dialogue-level contextual attention module for the dialogue presented in Table 4.8. For each utterance u_i on the y-axis, we compute attention weights for the 5 utterances, i.e., the current and previous four utterances (*AtnWidth*^{D} = 5). The cell values $(i, i-4), (i, i-3), (i, i-2), (i, i-1)$, and (i, i) represents the attention weights of utterances $u_{i-4}, u_{i-3}, u_{i-2}, u_{i-1}$, and u_i, respectively. The colormap signifies the amount attention weight for the respective utterances. The darker the shade, higher the attention weight assigned by MSH-COMICS.



(a) Textual attention. (b) Acoustic attention. (c) Textual and acoustic cross-modal attention.

Figure 4.5: Sarcasm Detection: Heatmap of the dialogue-level contextual attention module for the dialogue presented in Table 4.8.

We also perform heatmap analysis of the attention weights as computed by the system. For the analysis, we take the same dialogue as presented in Table 4.8 and depict the heatmaps of dialogue-level contextual attention *C-ATN*^{D} in Figures 4.4 and 4.5 for the humour classification and sarcasm detection, respectively. For each case, we show three separate heatmaps of the attention matrices corresponding to the textual, acoustics, and cross-modal attention modules. Each row i represents an utterance for which we compute attention weights for the five (current and four previous) utterances (i.e., *AtnWidth*^{D}=5) and the color shade signifies the amount of attention the model assigns to the corresponding utterances - darker shades represent higher weight, while lighter shades signify lower weights. Rest of the entries have attention weight zero, as they don't participate in the computation.

From the heatmaps, we can observe that the attention modules assign different wights to the contextual utterances depending upon their importance. For example, the system assigns higher weight on the previous textual content (c.f. Figure 4.4a $\beta_1 = 0.74 \& \beta_2 = 0.26$) and the current acoustic context (c.f.

Figure 4.4b $\alpha_1 = 0.39 \& \alpha_2 = 0.61$) for utterance u_2 in the humour classification. The distribution of attention weights can be justified by manual observation as well. The textual content of u_2 (i.e., *gaye kya?*|Did he die?) does not offer significant information about being humorous and one has to consider the context for the semantic. On the other hand, the audio signal reveals the excitement and tone in the voice of the speaker, and thus validates the higher attention weights by the system. Similarly, we observe many scenarios in other dialogs as well where the attention weights have high correlation with the contextual and multi-modal semantics of the utterances.

4.5.3 Evaluating LLMs

In order to see the performance of large language models such as Llama when compared with the proposed MSH-COMICS, we finetune the LLM with our MaSaC dataset and show its performance alongside the performance by MSH-COMICS in Table 4.10 for code-mixed conversations. For the English counterpart, we employ the MUMOR-EN and the MUStARD dataset for humour and sarcasm detection, respectively. As expected, Llama, being a larger model, performs better than the proposed methodologies, for the two tasks of sarcasm detection and humour identification.

		Sarcasm			**Humour**		
		Prec	**Rec**	**F1**	**Prec**	**Rec**	**F1**
Code-mixed	**MSH-COMICS**	0.811	0.636	0.711	0.785	0.858	0.820
	Llama	**0.904**	**0.711**	**0.796**	**0.853**	**0.871**	**0.862**
English	**MSH-COMICS**	**0.729**	0.716	**0.722**	0.794	0.786	0.789
	Llama	0.711	**0.731**	0.721	**0.795**	**0.789**	**0.792**

Table 4.10: Performance of Llama when compared with our proposed methodologies for sarcasm and humour identification.

4.6 Conclusion

In conclusion, this chapter addressed the challenges inherent in sarcasm detection and humour classification, recognizing their nuanced nature reliant on contextual and non-verbal cues. The limitation of existing studies, particularly in non-English languages like Hindi, prompted two significant contributions – Firstly, the creation of MaSaC, a novel Hindi-English code-mixed dataset, marked a pivotal step forward for multi-modal sarcasm detection and humour classification in conversational dialogues. It stood as the first dataset of its kind. Secondly, the introduction of MSH-COMICS, an innovative attention-rich neural architecture, elevated the realm of utterance classification. The model efficiently learned utterance representation through a hierarchical attention mechanism, focusing on specific segments of the input sentence. Additionally, a dialogue-level contextual attention mechanism capitalized on the dialogue history for more robust multi-modal classification. Through extensive experiments for both tasks, varying multi-modal inputs, and exploring different submodules of MSH-COMICS, the chapter demonstrated its superiority over existing models. A comprehensive diagnostic analysis of the model provided insights into its strengths and weaknesses. Furthermore, the extension of our proposed architecture to two monolingual datasets for humour and sarcasm detection in monolingual English yielded valuable improvements. This comprehensive exploration not only advanced the state-of-the-art in multi-modal sarcasm detection and humour classification but also contributed insights that may shape future developments in this nuanced field.

5. Sarcasm Explanation

Conversations emerge as the primary media for exchanging ideas. Identifying various affective qualities, such as sarcasm, humour, and emotions, is paramount for comprehending the true connotation of the emitted utterance. However, one of the major hurdles faced in learning these affect dimensions is the presence of figurative language *viz.* irony, metaphor, or sarcasm. Though sarcasm identification has been a well-explored topic in dialogue analysis, for conversational systems to truly grasp a conversation's innate meaning and generate appropriate responses, simply detecting sarcasm is not enough; it is vital to explain its underlying sarcastic connotation to capture its true essence. In this chapter, we study the discourse structure of sarcastic conversations and propose a novel task – *Sarcasm Explanation in Dialogue (SED)*. Set in a multimodal and multilingual setting, the task aims to generate natural language explanations of satirical conversations. To this end, we curate WITS, a new dataset to support our task. We propose MAF and MOSES, two deep neural networks, which take multimodal (sarcastic) dialogue instance as an input and generates a natural language sentence as its explanation. Subsequently, we leverage the generated explanation for various natural language understanding tasks in a conversational dialogue setup, such as *sarcasm detection*, *humour identification*, and *emotion recognition*. Our evaluation shows that MAF and MOSES outperforms the state-of-the-art system for SED on different evaluation metrics, such as ROUGE, BLEU, and BERTScore. Further, we observe that leveraging the generated explanation advances three downstream tasks for affect classification – an average improvement of $\sim 14\%$ F1-score in the sarcasm detection task and $\sim 2\%$ in the humour identification and emotion recognition task. We also perform extensive analyses to assess the quality of the results.

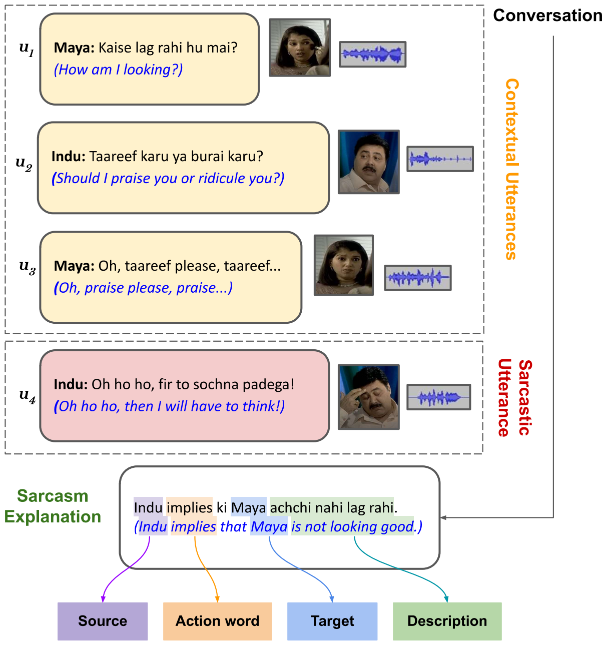
5.1 Introduction

Expressing oneself eloquently to our conversation partner requires employing multiple affective components such as emotion, humour, and sarcasm. All such attributes interact with each other to present a concrete definition of an uttered statement (204). While affects such as emotion and humour deem easier to comprehend, sarcasm, on the other hand, is a challenging aspect to understand (205). The use of figurative language serves many communicative purposes and is a regular feature of both oral and written communication (204). Predominantly used to induce humour, criticism, or mockery (206), paradoxical language is also used in concurrence with hyperbole to show surprise (207) as well as highlight the disparity between expectations and reality (208). While the use and comprehension of sarcasm is a cognitively taxing process (205), psychological evidence advocate that it positively correlates with the receiver's theory of mind (ToM) (209), i.e., the capability to interpret and understand another person's state of mind. Thus, for NLP systems to emulate such anthropomorphic intelligent behavior, they must not only be potent enough to identify sarcasm but also possess the ability to comprehend it in its entirety. To this end, moving forward from sarcasm identification, we propose the novel task of **Sarcasm Explanation in Dialogue** *aka* **SED**.

For dialogue agents, understanding sarcasm is even more crucial as there is a need to normalize its sarcastic undertone and deliver appropriate responses. Conversations interspersed with sarcastic statements often use contrastive language to convey the opposite of what is being said. In a real-world setting, understanding sarcasm goes beyond negating a dialogue's language and involves the acute comprehension of audio-visual cues. Additionally, due to the presence of essential temporal, contextual, and speaker-dependent information, sarcasm understanding in conversation manifests as a challenging problem. Consequently, many studies in the domain of dialogue systems have investigated sarcasm from textual, multimodal, and conversational standpoints (210; 211; 212; 105). However, baring some exceptions (213; 214; 215), research on figurative language has focused predominantly on its identification rather than its comprehension and normalization. This paper addresses this gap by attempting to generate natural language explanations of satirical dialogues.

To illustrate the proposed problem statement, we show an example in Figure 5.1. It contains a dyadic conversation of four utterances $\langle u_1, u_2, u_3, u_4 \rangle$, where the last utterance (u_4) is a sarcastic remark. Note that in this example, although the opposite of what is being said is, *"I don't have to think about it,"* it is not what the speaker means; thus, it enforces our hypothesis that sarcasm explanation goes beyond simply negating the dialogue's language. The discourse is also accompanied by ancillary audio-visual markers of satire such as an ironical intonation of the pitch, a blank face, or roll of the eyes. Thus, conglomerating the conversation history, multimodal signals, and speaker information, SED aims to generate a coherent and cohesive natural language explanation associated with sarcastic dialogues.

For the task at hand, we extend our MASAC dataset – a sarcasm detection dataset for code-mixed conversations – by augmenting it with natural language explanations for each sarcastic dialogue. We name the dataset WITS[1]. The dataset is a compilation of sarcastic dialogues from a popular Indian TV show. Along with the textual transcripts of the conversations, the dataset also contains multimodal signals of audio and video.



Maya: Kaise lag rahi hu mai?
(How am I looking?)

Indu: Taareef karu ya burai karu?
(Should I praise you or ridicule you?)

Maya: Oh, taareef please, taareef...
(Oh, praise please, praise...)

Indu: Oh ho ho, fir to sochna padega!
(Oh ho ho, then I will have to think!)

Indu implies ki Maya achchi nahi lag rahi.
(Indu implies that Maya is not looking good.)

Figure 5.1: SED: Given a sarcastic dialogue, the aim is to generate a natural language explanation for the sarcasm in it. *Blue text* represents the English translation for the text.

We experiment with unimodal as well as multimodal models to benchmark WITS. Text, being the driving force of the explanations, is given the primary importance, and thus, we compare a number of established text-based sequence-to-sequence systems on WITS. To incorporate multimodal information, we propose a unique fusion scheme of *Multimodal Context-Aware Attention* (MCA2). Inspired by (114), this attention variant facilitates deep semantic interaction between the multimodal signals and textual representations by conditioning the key and value vectors with audio-visual information and then performing dot product attention with these modified vectors. The generated audio and video information-informed textual representations are then combined using the *Global Information Fusion Mechanism* (GIF). The gating mechanism of GIF allows for the selective inclusion of information relevant to the satirical language and also prohibits any multimodal noise from seeping into the model. We further propose MAF (*Modality Aware Fusion*) module and the MOSES (MultimOdal Sarcasm Explanation

[1] WITS: "Why Is This Sarcastic"

with Spotlight) methodology, where the aforementioned mechanisms are introduced in the Generative Pretrained Language Models (GPLMs) as adapter modules. Our fusion strategies outperform the text-based baselines and the traditional multimodal fusion schemes in terms of multiple text-generation metrics. Finally, we conduct a comprehensive quantitative and qualitative analysis of the generated explanations.

All affective components, such as sarcasm, humour, and emotion, work in tandem to convey a statement's intended meaning (216; 217). Accordingly, we hypothesize that understanding one of the affective markers, like sarcasm, in its entirety will influence comprehending others. Consequently, in this chapter, we also deal with *leveraging sarcasm explanations* for three affect understanding tasks in dialogues, namely sarcasm detection, humour identification, and emotion recognition. The performance obtained from these tasks can be employed as a method to estimate the relevance of the SED task extrinsically. In a nutshell, our contributions are summarised below:

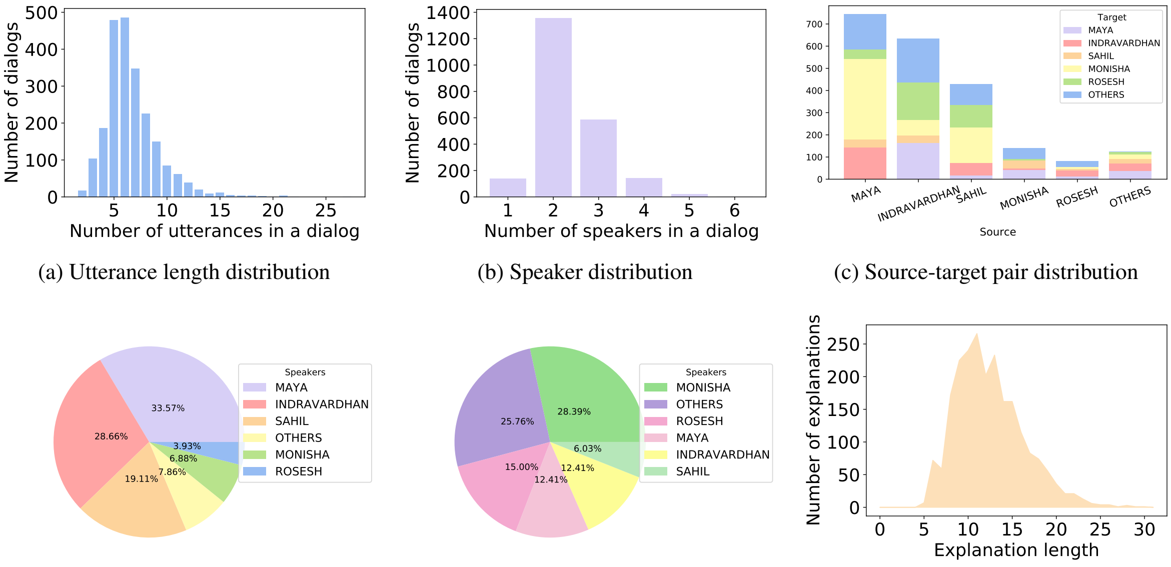
- We propose SED, a novel task aimed at generating a natural language explanation for a given sarcastic dialogue, elucidating the intended irony.
- We extend an existing sarcastic dialogue dataset, to curate WITS, a novel dataset containing human annotated gold standard explanations.
- We benchmark our dataset using MAF and MOSES variants of BART, that incorporate the audio-visual cues using a unique context-aware attention mechanism and pronunciation embeddings.
- We carry out extensive quantitative and qualitative analysis along with human evaluation to assess the quality of the generated explanations.

5.2 Related Work

Sarcasm and Text: A well-compiled survey (218) on computational sarcasm expanded on the relevant datasets, trends, and issues for automatic sarcasm identification. Early work in sarcasm detection dealt with standalone text inputs like tweets and reviews (58; 59; 177; 219). These initial works mostly focused on the use of linguistic and lexical features to spot the markers of sarcasm (58; 59). More recently, attention-based architectures are proposed to harness the inter- and intra-sentence relationships in texts for efficient sarcasm identification (60; 61; 62). Analysis of figurative language has also been extensively explored in conversational AI setting. A study (220) utilised attention-based RNNs to identify sarcasm in the presence of context. Two separate LSTMs-with-attention were trained for the two inputs (sentence and context) and their hidden representations were combined during the prediction. The study of sarcasm identification has also expanded beyond the English language. One study (221) collected a Hindi corpus of 2000 sarcastic tweets and employed rule-based approaches to detect sarcasm. Another (222) curated a dataset of 5000 satirical Hindi-English code-mixed tweets and used n-gram feature vectors with various ML models for sarcasm detection. Other notable studies include Arabic (223), Spanish (224), and Italian (225) languages.

Sarcasm and Multimodality: In the conversational setting, MUStARD, a multimodal, multi-speaker dataset (211) is considered the benchmark for multimodal sarcasm identification. A study (217) leveraged the intrinsic interdependency between emotions and sarcasm and devised a multi-task framework for multimodal sarcasm detection. Currently, the state-of-the-art on this dataset utlised a humour knowledge enriched transformer model (216). In the bimodal setting, sarcasm identification with tweets containing images has also been well explored (162; 226; 227) .

Beyond Sarcasm Identification: While studies in computational sarcasm have predominantly focused on sarcasm identification, some forays have been made into other domains of figurative language analysis. The work of converting sarcastic utterances into their non-sarcastic interpretations using deep learning has started (214). In another direction, a study (213) devised a modular unsupervised technique for sarcasm



(a) Utterance length distribution

(b) Speaker distribution

(c) Source-target pair distribution

(d) Sarcasm source distribution

(e) Sarcasm target distribution

(f) Explanation length distribution

Figure 5.2: Distribution of attributes in WITS. The number of utterances in a dialogue lies between 2 and 27. Maximum number of speakers in a dialogue are 6. The speaker 'Maya' is the most common common sarcasm source while the speaker 'Monisha' is the most prominent sarcasm target.

generation by introducing context incongruity through fact removal and incongruous phrase insertion. Following this, another research (215) proposed a retrieve-and-edit-based unsupervised framework for sarcasm generation. Their proposed model leverages the valence reversal and semantic incongruity to generate sarcastic sentences from their non-sarcastic counterparts. In summary, much work has been done in sarcasm detection, but little, if any, effort has been placed into *explaining the irony behind sarcasm*. We attempt to fill this gap by proposing a new problem definition and a supporting dataset.

5.3 Dataset

Situational comedies, or *'Sitcoms'*, vividly depict human behaviour and mannerism in everyday real-life settings. Consequently, the NLP research community has successfully used such data for sarcasm identification (211; 105). However, as there is no current dataset tailored for the proposed task, we curate a new dataset named WITS, where we augment the already existing MaSaC dataset with explanations for our task. We manually analyze the data and clean it for our task. While the original dataset contained 45 episodes of the TV series, we add 10 more episodes along with their transcription and audio-visual boundaries. Subsequently, we select the sarcastic utterances from this augmented dataset and

# Dlgs	**# Utts**	**# Eng utts**	**# Hin utts**
2240	9080	101	1453
# CM utts	**Avg. utt/dlg**	**Avg. sp/dlg**	**Avg. words/utt**
7526	4.05	2.35	14.39
Avg. words/dlg	**Vocab size**	**Eng vocab size**	**Hin vocab size**
58.33	10380	2477	7903

Table 5.1: Statistics of dialogues present in WITS.

manually define the utterances to be included in the dialogue context for each of them. Finally, we are left with 2240 sarcastic dialogues with the number of contextual utterances ranging from 2 to 27. Each of these instances is manually annotated with a corresponding natural language explanation interpreting its sarcasm. Each explanation contains four primary attributes – source and target of sarcasm, action word for sarcasm, and an optional description for the satire as illustrated in Figure 5.1. In the explanation "Indu implies that Maya is not looking good.", 'Indu' is the sarcasm source, 'Maya' is the target, 'implies' is the action word, while 'is not looking good' forms the description part of the explanation. We collect explanations in code-mixed format to keep consistency with the dialogue language. We split the data into train/val/test sets in an 80:10:10 ratio for our experiments, resulting in 1792 dialogues in the train set and 224 dialogues each in the validation and test sets. We also consider the monolingual English dataset - MUStARD, curated for the task of sarcasm identification. We follow the same guidelines as mentioned in the next subsection to annotate it with sarcasm explanations. The next section illustrates the annotation process in more detail. Table 5.1 and Figure 5.2 show detailed statistics of WITS.

5.3.1 Annotation Guidelines

Each of the instance in WITS and MUStARD is associated with a corresponding video, audio, and textual transcript such that the last utterance is sarcastic in nature. We first manually define the number of contextual utterances required to understand the sarcasm present in the last utterance of each dialogue. Further, we provide each of these sarcastic statements, along with their context, to the annotators who are asked to generate an explanation for these instances based on the audio, video, and text cues. Two annotators were asked to annotate both the datasets. The target explanation is selected by calculating the cosine similarity between the two explanations. If the cosine similarity is greater than 90% then the shorter length explanation is selected as the target explanation. Otherwise, a third annotator goes through the dialogue along with the explanations and resolves the conflict. The average cosine similarity after the first pass is 87.67% for WITS and 91.03 for MUStARD. All the final selected explanations contain the following attributes:

- **Sarcasm source:** The speaker in the dialogue who is being sarcastic.
- **Sarcasm target:** The person/ thing towards whom the sarcasm is directed.
- **Action word:** Verb/ action used to describe how the sarcasm is taking place. For e.g. mocks, insults, taunts, etc.
- **Description:** A description about the scene which helps in understanding the sarcasm.

5.3.2 Variations of WITS

In order to gauge the effect of sarcasm explanation on affective attributes, we augment WITS to perform sarcasm detection, humour identification, and emotion recognition on it. We create instances for sarcastic and non-sarcastic utterances with their context to perform sarcasm detection. We call this variation of the dataset sWITS. Adapted from MaSaC, WITS can also be mapped to annotations for humour identification, where each utterance contains a binary

	WITS	**sWITS**		**hWITS**		**eWITS**			
	#S	**#NS**	**#S**	**#NH**	**#H**	**#Ntrl**	**#Sad**	**#Joy**	**#Ang**
Train	1792	1669	1792	2795	995	1590	1147	623	429
Val	224	213	224	362	112	196	133	87	57
Test	224	218	224	367	106	195	141	70	67
Total	2240	2100	2240	3524	1213	1981	1421	780	553

Table 5.2: Statistics of the sarcasm, humour, and emotion (N: Ntrl: Neutral, Ang: Anger) datasets in consideration (number of dialogue instances marked as sarcastic (#S), non-sarcastic (#NS), non-humorous (#NH), and humorous (#H).).

marker showcasing whether the utterance is amusing or not. Consequently, we map each instance in sWITS to its corresponding humour annotation. Additionally, we determine emotion labels for the instances at hand and identify the following emotions – *sadness, joy, anger*, *and neutral.* Three annotators were involved in this phase and achieved an inter-annotator agreement of 0.86. Accordingly, we obtain four variations of the dataset:

1. WITS: It contains multimodal, multiparty, code-mixed, sarcastic instances with associated explanations.
2. sWITS: It contains sarcastic and non-sarcastic instances constructed from WITS. The last utterance of each instance is marked by a binary tag indicating whether the statement contains sarcasm or not.
3. hWITS: For each instance created in sWITS, each target utterance is marked with another binary label revealing the existence of humour in it.
4. eWITS: Similar to hWITS, this variant contains emotion labels for the target utterances.

Table 5.2 illustrates the elementary statistics for the explained dataset variations. We explain each dataset curation in detail below.

sWITS: The parent dataset, WITS contains sarcastic instances along with their explanations. Each instance of WITS contains a sequence of utterances where the last utterance is sarcastic. However, for the sarcasm detection task, we need both sarcastic as well as non-sarcastic instances. To create the non-sarcastic instances, we randomly sample utterances from the context of the instances present in WITS. Figure 5.3 illustrates the process of creating sWITS from WITS.

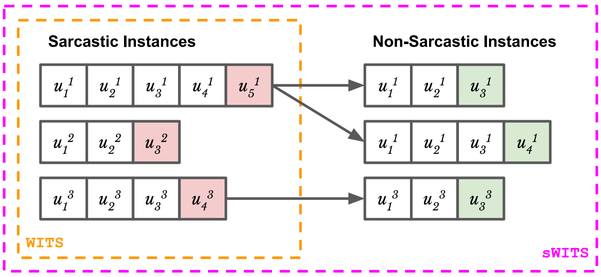


Figure 5.3: Construction of sWITS from WITS.

hWITS: To gauge the effect of sarcasm explanation on humour identification, we will need instances with humour labels. As a result, we explore the mapping existing between WITS and the MaSaC dataset. MaSaC contains binary markers identifying the presence of sarcasm and humour in all utterances. The WITS dataset is an extended version of the MaSaC dataset where all the sarcastic utterances are appended with their corresponding natural language explanation. Ergo, we map the humour labels from MaSaC to the instances present in sWITS and get hWITS.

eWITS: Sarcasm significantly affects the emitted emotion of an utterance. Wherefore, we hypothesize that emotion recognition in conversation can be improved in the presence of utterance explanations. Therefore, we need emotion labels for the instances present in sWITS to create eWITS. We annotate the instances for emotion labels following the Ekman (72) emotion scheme. Out of the seven possible emotion labels, namely *anger*, *fear*, *disgust*, *sadness*, *joy*, and *surprise*, *neutral*, we were able to identify four for our set of instances – *anger*, *sadness*, *joy*, and *neutral*. Three annotators (ABC) were involved and achieved Krippendorff's Alpha (108) inter-annotator scores as $\alpha_{AB} = 0.84$, $\alpha_{BC} = 0.88$, and $\alpha_{AC} = 0.86$ giving an average score of 0.86. We show couple of instances from all the discussed datasets in Table 5.3.

Context Speakers	**Context Utterances**	**Target Speaker**	**Target Sarcastic Utterance**	**Explanation**
INDRAVARDHAN	Accha suno Monisha tumhaare ghar mein been ya aisa kuuch hain? *(Listen Monisha, do you have a flute or something similar?)*	MAYA	Kaise hogi? Monisha aapne ghar pe dustbin mushkil se rakhti hain to snake charmer waali been kaha se rakhegi? *(How will it be there? Monisha hardly keeps a dustbin in her home so how will she has a snake charmer's flute?)*	Maya Monisha ko tana marti hai safai ka dhyan na rakhne ke liye. *(Maya taunts Monisha for not keeping a check of cleanliness)*
SAHIL MONISHA	Ab tumne ghar ki itni saaf safai ki hai and secondly us Karan Verma ke liye pasta, lasagne, caramel custard banaya. *(Now you have cleaned the house so much and secondly made pasta, lasagne, caramel custard for that Karan Verma.)* Walnut brownie bhi. *(And walnut brownie too.)*	SAHIL	Walnut brownie, matlab wo khane wali? *(You mean edible walnut brownie?)*	Sahil monisha ki cooking ka mazak udata hai. *(Sahil makes fun of Monisha's cooking.)*

(a) For WITS

Context Speakers	Context Utterances	Target Speaker	Target Utterance	sWITS	hWITS	eWITS
MONISHA SAHIL	Dukan se yaad aya, mummy ji wahan pe Southhall ya Wembley ki kisi dukan se please mere liye chandi ka mangalsutra le aaiega na *(Talking about shops, mom please get me a silver necklace from any shop from Southhall or Wembley.)* Mangalsutra London se? *(You want a necklace from London?)*	MONISHA	Haan wahan pe kali mani ke neeche Big Ben ki pendant wala mangalsutra milta hai. *(Yes, we can get a necklace of black beads from there.)*	0	1	Joy
ROSESH INDRAVARDHAN ROSESH	Momma mujhe bohot achi lagti hai. *(I like momma very much)* I know that. Momma pari hai pari! *(I know that. Your mother is like a fairy.)* Aur me? *(And me?)*	INDRAVARDHAN	Rakshas! *(Monster!)*	1	0	Anger

(b) For sWITS, hWITS, and eWITS

Table 5.3: Sample instances for WITS, sWITS, hWITS, and eWITS

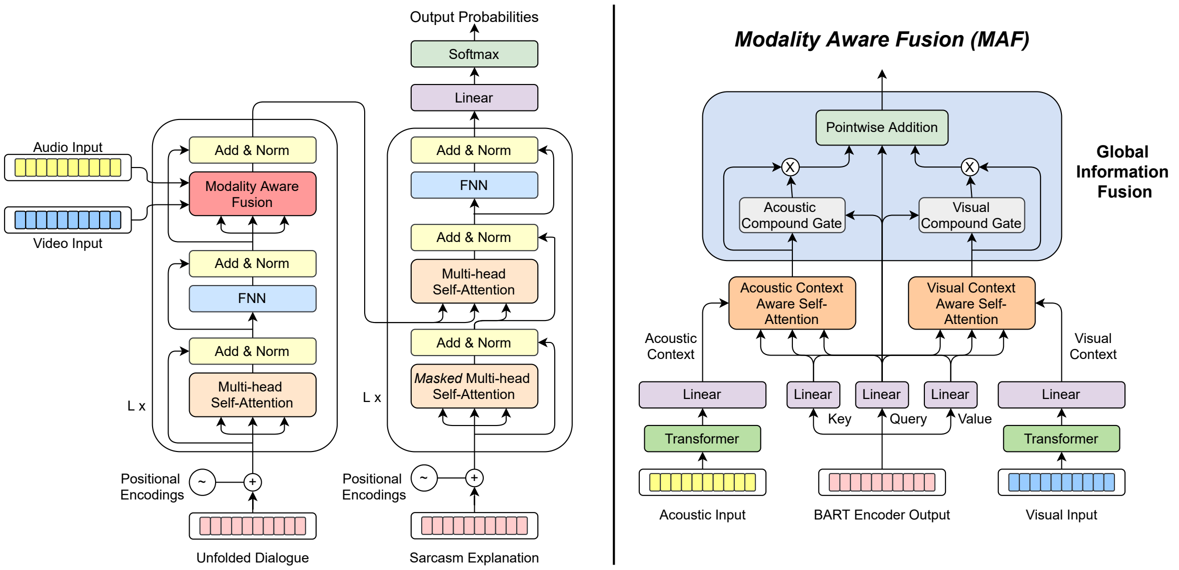


Figure 5.4: Model architecture for MAF. The proposed Multimodal Fusion Block captures audio-visual cues using Multimodal Context Aware Attention (MCA2) which are further fused with textual representations using Global Information Fusion (GIF) block.

5.4 Methodology

5.4.1 MAF

In this section, we present our model and its nuances. The primary goal is to smoothly integrate multimodal knowledge into the BART architecture. To this end, we introduce *Multimodal Aware Fusion* (MAF), an adapter-based module that comprises of *Multimodal Context-Aware Attention (MCA2)* and *Global*

Information Fusion (GIF) mechanisms. Given the textual input sarcastic dialogue along with the audio-video cues, the former aptly introduces multimodal information in the textual representations, while the latter conglomerates the audio-visual information infused textual representations. This adapter module can be readily incorporated at multiple layers of BART/mBART to facilitate various levels of multimodal interaction. Figure 5.4 illustrates our model architecture.

Multimodal Context Aware Attention

The traditional dot-product-based cross-modal attention scheme leads to the direct interaction of textual representations with other modalities. Here the text representations act as the query against the multimodal representations, which serve as the key and value. As each modality comes from a different embedding subspace, a direct fusion of multimodal information might not retain maximum contextual information and can also leak substantial noise in the final representations. Thus, based on the findings of (114), we propose multimodal fusion through *Context Aware Attention*. We first generate multimodal information conditioned key and value vectors and then perform the traditional scaled dot-product attention. We elaborate on the process below.

Given the intermediate representation H generated by the GPLMs at a specific layer, we calculate the query, key, and value vectors Q, K, and $V \in \mathbb{R}^{n \times d}$, respectively, as given in Equation 6.1, where W_Q, W_K, and $W_V \in \mathbb{R}^{d \times d}$ are learnable parameters. Here, n denotes the maximum sequence length of the text, and d denotes the dimensionality of the GPLM generated vector.

$$\left[QKV\right] = H\left[W_Q W_K W_V\right] \tag{5.1}$$

Let $C \in \mathbb{R}^{n \times d_c}$ denote the vector obtained from audio or visual representation. We generate multimodal information informed key and value vectors $\hat{K}$ and $\hat{V}$, respectively, as given by (114). To decide how much information to integrate from the multimodal source and how much information to retain from the textual modality, we learn matrix $\lambda \in \mathbb{R}^{n \times 1}$ (Equation 6.3). Note that U_k and $U_v \in \mathbb{R}^{d_c \times d}$ are learnable matrices.

$$\begin{bmatrix}\hat{K}\\ \hat{V}\end{bmatrix} = (1 - \begin{bmatrix}\lambda_k\\ \lambda_v\end{bmatrix})\begin{bmatrix}K\\ V\end{bmatrix} + \begin{bmatrix}\lambda_k\\ \lambda_v\end{bmatrix}(C\begin{bmatrix}U_k\\ U_v\end{bmatrix}) \tag{5.2}$$

Instead of making λ_k and λ_v as hyperparameters, we let the model decide their values using a gating mechanism as computed in Equation 6.3. The matrices of $W_{k_1}, W_{k_2}, W_{v_1}$, and $W_{v_2} \in \mathbb{R}^{d \times 1}$ are trained along with the model.

$$\begin{bmatrix}\lambda_k\\ \lambda_v\end{bmatrix} = \sigma(\begin{bmatrix}K\\ V\end{bmatrix}\begin{bmatrix}W_{k_1}\\ W_{v_1}\end{bmatrix} + C\begin{bmatrix}U_k\\ U_v\end{bmatrix}\begin{bmatrix}W_{k_2}\\ W_{v_2}\end{bmatrix}) \tag{5.3}$$

Finally, the multimodal information infused vectors $\hat{K}$ and $\hat{V}$ are used to compute the traditional scaled dot-product attention. For our case, we have two modalities – audio and video. Using the *context-aware attention mechanism*, we obtain the acoustic-information-infused and visual-information infused vectors H_A and H_V, respectively (c.f. Equations 5.4 and 5.5).

$$H_a = Softmax(\frac{Q\hat{K}_a^T}{\sqrt{d_k}})\hat{V}_a \tag{5.4}$$

$$H_v = Softmax(\frac{Q\hat{K}_v^T}{\sqrt{d_k}})\hat{V}_v \tag{5.5}$$

Global Information Fusion

In order to combine the information from both the acoustic and visual modalities, we design the GIF block. We propose two gates, namely the *acoustic gate* (g_a) and the *visual gate* (g_v) to control the amount of information transmitted by each modality. They are as follows:

$$g_a = [H \oplus H_a]W_a + b_a \tag{5.6}$$

$$g_v = [H \oplus H_v]W_v + b_v \tag{5.7}$$

Here, $W_a, W_v \in \mathbb{R}^{2d \times d}$ and $b_a, b_v \in \mathbb{R}^{d \times 1}$ are trainable parameters, and $\oplus$ denotes concatenation. The final multimodal information fused representation $\hat{H}$ is given by Equation 5.10.

$$\hat{H} = H + g_a \odot H_a + g_v \odot H_v \tag{5.8}$$

This vector $\hat{H}$ is inserted back into GPLM for further processing.

5.4.2 MOSES

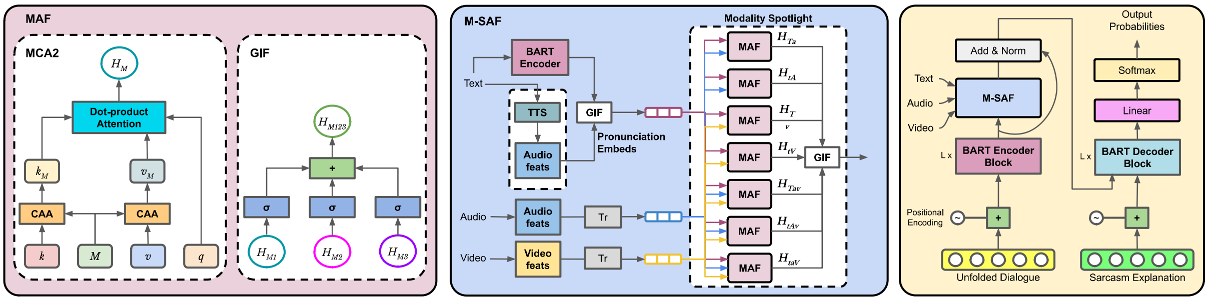


Figure 5.5: MOSES: The MAF model captures acoustic and visual hints using the Multimodal Context Aware Attention (MCA2) and combines them using Global Information Fusion (GIF). Each modality is kept in spotlight using the Modality Spotlight (MS) module. To capture the subjectivity in the code-mixed spellings, we propose *pronunciation embeddings*.

This section illustrates the working of our proposed model, MOSES as presented in Figure 5.5. The existing SED model, MAF, which uses a modified version of context-aware attention (114), takes the multimodal (audio/video) vectors as context and fuses them with the text modality to generate multimodal fused text vectors. This way of multimodal fusion makes text the primary modality and treats the other signals (acoustic and visual) as secondary. Such a fusion technique might result in the downplay of the audio and video modalities. However, in the complete duration of the discourse, modalities other than text could play the deciding role in resolving the affects in consideration. Consequently, we propose using context-aware fusion in such a way that each modality gets a chance to play a pivotal role in the fusion computation. We propose two additional modules in tandem with the previously proposed MCA2 and GIF module, which we describe below.

Modality Spotlight (MS).

We discussed how we can generate multimodal infused vector representation considering one modality as primary and the rest as context. Our work deals with three modalities – text, acoustic, and visual. The spotlight module is responsible for treating each of these modalities as the primary modality at a time and generating the corresponding fused vectors. For instance, if text is considered the primary modality, then we need to calculate two multimodal fused vectors, H_{Ta} and H_{Tv}, such as audio and video, play the role of context in the representations, respectively. Similarly, when audio and video are considered the primary source of information, H_{tA} and H_{tV} are calculated. Note that we do not calculate H_{Av} or H_{aV} because we are dealing with a textual generation task where the textual information plays the preliminary role.

Apart from bi-modal interactions, we also deal with tri-modal interactions in our work, where all three modalities are infused using the GIF module. Unlike bi-modal fusion, it is unfair to let text be the only primary modality in the tri-modal fusion. Consequently, we compute three tri-modal vectors, H_{Tav}, H_{tAv}, and H_{taV}, such that text, audio, and video individually play the primary role, respectively. The GIF module is responsible for combining the information from multiple modalities together in an efficient manner. G gates are used to control the amount of information disseminated by each modality, where $2 \leq G \leq 3$ is the number of modalities to fuse. For instance, if we calculate the interaction between the text and audio modalities with text being the primary source of information, we will first need to calculate the gated information from the audio representation using Equation 5.9.

$$g_a = [H \oplus H_a]W_a + b_a \tag{5.9}$$

where W_a and b_a are learnable matrices, and $\oplus$ denotes vector concatenation. The final representation to be passed on to the next encoder layer will be obtained using Equation 5.10.

$$H_{Ta} = H + g_a \odot H_a \tag{5.10}$$

On similar lines, if we are to calculate the tri-modal representation keeping the text as the primary modality, we first compute the gated vector for audio and video and then compute a weighted combination of the three modalities. The following sequence of equations illustrates this process,

$$\begin{aligned} g_a &= [H \oplus H_a]W_a + b_a \\ g_v &= [H \oplus H_v]W_v + b_v \\ H_{Tav} &= H + g_a \odot H_a + g_v \odot H_v \end{aligned}$$

Likewise, we calculate the following set of vectors: H_{Ta}, H_{tA}, H_{Tv}, H_{tV}, H_{Tav}, H_{tAv}, and H_{taV}. Further, another GIF module is used to conglomerate these seven vectors, as shown in Equation 5.11.

$$\begin{aligned} H_{all} = g_t \odot H + g_{Ta} \odot H_{Ta} + g_{tA} \odot H_{tA} + \\ g_{Tv} \odot H_{Tv} + g_{tV} \odot H_{tV} + g_{Tav} \odot H_{Tav} + \\ g_{tAv} \odot H_{tAv} + g_{taV} \odot H_{taV} \end{aligned} \tag{5.11}$$

Pronunciation Embedding (PE)

The textual input in WITS is present in romanised code-mixed format. Thereby, it may contain terms with the same meaning but varying spellings that are phonetically identical. For instance, the word *"main"* in Hindi (translating to *"I"* in English) can be written as *"main"* or *"mein"*. To capture the similarity between all these spelling variations, we propose using *Pronunciation Embeddings (PE)* that capture the phonetic equivalence between the words of the input text. We convert the text into a standard speech format using python's gTTS library[2]. This converted audio does not contain any tone or pitch variation for any term and thus, sounds the same for phonetically similar terms. We then extract the audio features

[2] https://pypi.org/project/gTTS/

from this converted speech. This pronunciation vector is fused with the text representation, obtained from any encoder model like BART, using the GIF module to obtain the final text representation.

5.5 Experiments and Results

This section illustrates the feature extraction strategy we use and the baseline systems to which we compare our model, followed by the results we obtain for the SED task. We use the standard generative metrics – ROUGE-1/2/L (228), BLEU-1/2/3/4 (229), and METEOR (230) to capture the syntactic and semantic performance of our systems.

5.5.1 Feature Extraction

The primary challenges for generating vector representations for the instances in WITS come from the code-mixed and multimodal aspects of the dataset. We alleviate these by proposing intelligent feature extraction methods. The same methods are used for the MUStARD dataset as well.

Audio: Acoustic representations for each instance are obtained using the openSMILE python library[3]. We use a window size of 25 ms and a window shift of 10 ms to get the non-overlapping frames. Further, we employ the eGeMAPS model (231) and extract 154 dimensional functional features such as Mel Frequency Cepstral Coefficients (MFCCs) and loudness for each frame of the instance. These features are then fed to a Transformer encoder (150) for further processing.

Video: We use a pre-trained action recognition model, ResNext-101 (232), trained on the Kinetics dataset (233) which can recognise 101 different actions. We use a frame rate of 1.5, a resolution of 720 pixels, and a window length of 16 to extract the 2048 dimensional visual features. Similar to audio feature extraction, we employ a Transformer encoder (150) to capture the sequential dialogue context in the representations.

5.5.2 Comparative Systems

We use various established sequence-to-sequence (seq2seq) models to obtain the most promising textual representations for the discourse.

- **RNN**: The openNMT4[4] implementation of the RNN seq2seq architecture is used to obtain the results.
- **Transformer** (150): Explanations are generated using the vanilla Transformer encoder-decoder model.
- **Pointer Generator Network (PGN)** (234): A combination of generation and copying mechanisms is used in this seq2seq architecture.
- **BART** (235): We use the base version of this denoising autoencoder model. It has a bidirectional encoder with an auto-regressive left-to-right decoder built on standard machine translation architecture.
- **mBART** (236): Trained on multiple large-scale monolingual corpora, mBART follows the same objective and architecture as BART[5].

[3] https://audeering.github.io/opensmile-python/

[4] https://github.com/OpenNMT/OpenNMT-py

[5] https://huggingface.co/facebook/mbart-large-50-many-to-many-mmt

5.5.3 Evaluating MAF

Text Based: As evident from Table 5.4, BART performs the best across all the metrics for the textual modality, showing an improvement of almost 2-3% on the METEOR and ROUGE scores when compared with the next best baseline. PGN, RNN, and Transformers demonstrate admissible performance considering that they have been trained from scratch. However, it is surprising to see mBART not performing better than BART as it is trained on multilingual data.

Multimodality: Psychological and linguistic literature suggests that there exist distinct paralinguistic cues that aid in comprehending sarcasm and humour (237; 238). Thus, we gradually merge auditory and visual modalities using MAF module and obtain MAF-TAV and MAF-TAV$_\mathrm{m}$ for BART and mBART, respectively. We observe that the inclusion of acoustic signals leads to noticeable gains of 2-3% across the ROUGE, BLEU, and METEOR scores. The rise in BERTScore also suggests that the multimodal variant generates a tad more coherent explanations. As ironical intonations such as mimicry, monotone, flat contour, extremes of pitch, long pauses, and exaggerated pitch (239) form a significant component in sarcasm understanding, we surmise that our model, to some extent, is able to spot such markers and identify the intended sarcasm behind them.

We notice that visual information also contributes to our cause. Significant performance gains are observed for MAF-TV and MAF-TV$_\mathrm{m}$, as all the metrics show a rise of about 3-4%. While MAF-TA gives marginally better performance over MAF-TV in terms of R1, RL, and B1, we see that MAF-TV performs better in terms of the rest of the metrics. Often, sarcasm is depicted through gestural cues such as raised eyebrows, a straight face, or an eye roll (237). Moreover, when satire is conveyed by mocking someone's looks or physical appearances, it becomes essential to incorporate information expressed through visual media. Thus, we can say that, to some extent, our model is able to capture these nuances of non-verbal cues and use them well to normalize the sarcasm in a dialogue. In summary, we conjecture that whether independent or together, audio-visual signals bring essential information to the table for understanding sarcasm.

Mode	Model	R1	R2	RL	B1	B2	B3	B4	M	BS
Textual	**RNN**	29.22	7.85	27.59	22.06	8.22	4.76	2.88	18.45	73.24
	Transformers	29.17	6.35	27.97	17.79	5.63	2.61	0.88	15.65	72.21
	PGN	23.37	4.83	17.46	17.32	6.68	1.58	0.52	23.54	71.90
	mBART	33.66	11.02	31.50	22.92	10.56	6.07	3.39	21.03	73.83
	BART	36.88	11.91	33.49	27.44	12.23	5.96	2.89	26.65	76.03
Multimodality	**MAF-TA**$_m$	39.02	15.90	36.83	31.26	16.94	11.54	7.72	29.05	77.06
	MAF-TV$_m$	39.47	16.78	**37.38**	32.44	17.91	12.02	7.36	29.74	77.47
	MAF-TAV$_m$	38.52	14.13	36.60	30.50	15.20	9.78	5.74	27.42	76.70
	MAF-TA	38.21	14.53	35.97	30.58	15.36	9.63	5.96	27.71	77.08
	MAF-TV	37.48	15.38	35.64	30.28	16.89	10.33	6.55	28.24	76.95
	MAF-TAV	**39.69**	**17.10**	37.37	**33.20**	**18.69**	**12.37**	**8.58**	**30.40**	**77.67**

Table 5.4: Experimental results. (Abbreviation: R1/2/L: ROUGE1/2/L; B1/2/3/4: BLEU1/2/3/4; M: METEOR; BS: BERT Score; PGN: Pointer Generator Network).

Ablation Study

Table 5.5 reports the ablation study. CONCAT1 represents the case where we perform bimodal concatenation $((T \oplus A), (T \oplus V))$ instead of the *MCA2* mechanism, followed by the GIF module, whereas, CONCAT2 represents the simple trimodal concatenation $(T \oplus A \oplus V)$ of acoustic, visual, and textual representations followed by a linear layer for dimensionality reduction. In comparison with *MCA2*, CONCAT2 reports a below-average performance with a significant drop of more than 14% for MAF-TAV and MAF-TAV$_m$. This highlights the need to have deftly crafted multimodal fusion mechanisms. CONCAT1, on the other hand, gives good performance and is competitive with DPA and MAF-TAV.

We speculate that treating the audio and video modalities separately and then merging them to retain the complimentary and differential features lead to this performance gain. Our proposed MAF outperforms DPA with gains of 1-3$\%$. This underlines that our unique multimodal fusion strategy is aptly able to capture the contextual information provided by the audio and video signals. Replacing the *GIF* module with simple addition, we observe a noticeable decline in the performance across almost all metrics by about 2-3$\%$. This attests to the inclusion of *GIF* module over simple addition. We also experiment with fusing multimodal information using MAF before different layers of the BART encoder. The best performance was obtained when the fusion was done before the sixth layer of the architecture.

Model	R1	R2	RL	B1	B2	B3	B4	M	BS
MAF-TAV$_m$	**38.52**	14.13	36.60	30.50	15.20	9.78	5.74	27.42	76.70
- MCA2 + CONCAT1	37.56	**14.85**	34.90	30.16	15.76	**10.12**	**6.82**	**28.59**	76.59
- MAF + CONCAT2	17.22	1.70	14.12	13.11	2.11	0.00	0.00	9.34	66.64
- MCA2 + DPA	36.43	13.04	33.75	28.73	14.02	8.00	4.89	25.60	75.58
- GIF	36.37	13.85	34.92	28.49	14.34	9.00	6.16	25.75	**76.86**
MAF-TAV	**39.69**	**17.10**	**37.37**	**33.20**	**18.69**	**12.37**	**8.58**	**30.40**	**77.67**
- MCA2 + CONCAT1	36.88	13.21	34.39	29.63	14.56	8.43	4.84	26.15	76.08
- MAF + CONCAT2	21.11	2.31	19.68	12.44	2.44	0.73	0.31	9.51	69.54
- MCA2 + DPA	38.84	14.76	36.96	30.23	15.95	9.88	5.83	28.04	77.20
- GIF	39.45	14.85	37.18	31.85	15.97	9.62	5.47	28.87	77.54

Table 5.5: Ablation results on MAF-TAV$_\mathrm{m}$ and MAF-TAV (DPA: Dot Product Attention).

5.5.4 Evaluating MOSES

Textual: Table 5.6 shows the results obtained when textual systems are used to obtain the generated explanations. We notice that while PGN delivers us with the least performance across most metrics, BART-based representations outperform the rest by providing the best performance across the majority of all evaluation metrics.

Pronunciation Embeddings (PE): Due to the subjective nature of how other languages (Hindi, in our case) are written in a romanised format, the spellings of the words come from their phonetic understanding. To resolve the ambiguity between the same words with differing spellings, we propose to use pronunciation embeddings. As illustrated in Table 5.6, we observe that by adding the PE component to the model with the help of the GIF module, the performance of text-based systems jumps by an average of $\sim 4\%$ across all evaluation metrics.

	Model	R1	R2	RL	B1	B2	B3	B4	M
Textual	**RNN**	29.22	7.85	27.59	22.06	8.22	4.76	2.88	18.45
	Transformer	29.17	6.35	27.97	17.79	5.63	2.61	0.88	15.65
	PGN	23.37	4.83	17.46	17.32	6.68	1.58	0.52	23.54
	mBART	33.66	11.02	31.5	22.92	10.56	6.07	3.39	21.03
	BART	36.88	11.91	33.49	27.44	12.23	5.96	2.89	26.65
Multimodal	**MAF-TA**	38.21	14.53	35.97	30.58	15.36	9.63	5.96	27.71
	MAF-TV	37.48	15.38	35.64	30.28	16.89	10.33	6.55	28.24
	MAF-TAV	39.69	17.1	37.37	33.2	18.69	12.37	8.58	30.4
	MOSES-TA	38.27	14.53	35.72	31.57	16.37	9.66	6.06	29.27
	MOSES-TV	39.62	16.78	37.48	32.69	17.76	11.01	6.89	31.65
	MOSES-TAV	40.88	18.33	38.38	33.27	18.87	12.6	8.8	31.41
	MOSES	**42.17**	**20.38**	**39.66**	**34.95**	**21.47**	**15.47**	**11.45**	**32.37**

Table 5.6: Experimental results (Abbreviation: R1/2/L: ROUGE1/2/L; B1/2/3/4: BLEU1/2/3/4; M: METEOR; PGN: Pointer Generator Network). Final row denotes MOSES including the pronunciation and spotlight modules.

Multimodality: After we obtain the representation for the code-mixed text by fusing textual representation with pronunciation embeddings, we move on to adding multimodality to the system. We experimented with an established SED method (MAF-TAV) to estimate the effect of multimodality. Table 5.6 exhibits that while the addition of acoustic signals does not result in a performance gain, the addition of visual cues boosts the performance by $\sim 1\%$ across all metrics. This phenomenon can be attributed to the fact that audio alone may cause confusion while understanding sarcasm, and visual hints may help in such times. Thereby, improving the visual feature representations can be one of the future directions. Finally, when we add all multimodal signals together, we observe the best performance yet with an average increase of further $\sim 1\%$ across majority metrics.

Modality Spotlight: As hypothesised, we obtain the best performance for sarcasm understanding when all the three modalities are used in tandem. We argue that especially in the case of sarcasm, multimodal signals such as audio and video might play the principal role in many instances. To comprehend this rotating importance of modalities, we use the spotlight module that aims to treat each modality as the primary modality while calculating the final representation. We observe an increase of $\sim 2\%$ across all evaluation metrics as shown in Table 5.6. These results directly support our hypothesis of the effect of multimodality in sarcasm analysis.

Ablation Study

Model	**R1**	**R2**	**RL**	**B1**	**B2**	**B3**	**B4**	**M**
BART	36.88	11.91	33.49	27.44	12.23	5.96	2.89	26.65
+concat	17.22	1.7	14.12	13.11	2.11	0.0	0.0	9.34
+DPA	36.43	13.04	33.75	28.73	14.02	8.0	4.89	25.6
+MCA2	36.37	13.85	34.92	28.49	14.34	9.0	6.16	25.75
+ GIF	39.69	17.1	37.37	33.2	18.69	12.37	8.58	30.4
+ PE	40.88	18.33	38.38	33.27	18.87	12.6	8.8	31.41
+ MS (MOSES)	**42.17**	**20.38**	**39.66**	**34.95**	**21.47**	**15.47**	**11.45**	**32.37**

Table 5.7: Ablation results on MOSES (Abbreviation: DPA: Dot Product Attention).

To highlight the importance of all modules in consideration, we perform extensive ablation studies on the WITS dataset. Table 5.7 shows the results when we add the different proposed modules to our system sequentially. The first row highlights the BART model's results for the text modality which results in a ROUGE-2 of 11.91%. As illustrated, the use of naive trimodal concatenation ($T \oplus A \oplus V$) of text, audio, and video representations produces a noisy fusion resulting in decreased performance (-10.2% ROUGE-2). Next, we try with the standard dot-product attention, which, being a comparatively smarter way of multimodal fusion, results in a slightly improved performance over the text-only modality ($+2\%$ ROUGE-2). Further, adding the multimodal context-aware attention module (MCA2) and replacing standard dot-product attention, produces a further performance boost by $\sim 1\%$ across all metrics, signifying the importance of the intelligent fusion that the MCA2 module provides us. The performance is increased even more when the GIF module is introduced to compute the final multimodal vector representation ($+4\%$ ROUGE-2), signifying the positive effect gated fusion has on efficient multimodal representations. Next, we incorporate pronunciation embeddings (PE) into the model and observe another performance boost across majority metrics ($\sim 1\%$), suggesting that we can obtain better code-mixed representations by reducing the spelling ambiguities. Finally, our entire model with modality spotlight included produces the best performance, verifying the necessary use of each module discussed.

5.5.5 Result Analysis

	mBART	**BART**	**MAF**	**MOSES**
Source	75	77.23	**91.07**	90.17
Target	45.33	52.67	46.42	**56.69**

Table 5.8: Accuracy for the sarcasm source and target for BART-based systems.

Quantitative Analysis We evaluate the MAF and MOSES on their ability to capture sarcasm source and target in the generated explanations. We compare them with mBART and BART. Table 5.8 shows that BART performs better than mBART for both source and target detection. The inclusion of multimodal signals, even without pronunciation embeddings and modality spotlight, improves the source identification performance by $\sim 14\%$. MOSES is able to detect the sarcasm source most efficiently, resulting in an improvement of $\sim 4\%$ over the next best result. Consequently, we can relate the presence of multimodal capabilities to capture speaker-specific peculiarities more efficiently, resulting in better source/target identification.

Qualitative Analysis We sample a few dialogues from the test set of WITS and show their generated explanations by MAF and MOSES along with the ground-truth explanations in Table 5.9. We show one of the many instances where our model generates the correct explanation for the given sarcastic instance in the first row. The last row, highlights a case where the generated explanation is not syntactically similar to the ground-truth explanation but resembles it semantically. To evaluate the semantic similarity properly, we perform a human evaluation.

Dialogue	**Ground Truth**	**MAF**	**MOSES**
KISMI: Bas na Sahil bhai, meri firki kheech rahe ho na!? *(Enough brother Sahil, are you teasing me?!)* SAHIL: Nahi, nahi, kya hai ki, mere CD ki collection mein na, ye train ke awaaj vali CD nahi hai... *(No no, see I don't have train's sound in my CD collection...)*	Sahil Kismi ko taunt maarta hai kyuki use rail gaadi ki awaaj sunni hai. *(Sahil taunts Kismi that she wants to hear the sound of a train)*	Sahil Kismi ko taunt maarta hai ki use pasand nahi. *(Sahil taunts Kismi that he doesn't like)*	Sahil Kismi ko taunt maarta hai kyuki use rail gaadi ki awaaj sunni hai. *(Sahil taunts Kismi that she wants to hear the sound of a train)*
MONISHA: Say hello to Tommy the dog. *(Say hello to Tommy the dog.)* MAYA: Tumne iss kutte ka naam Tommy the dog rakha? *(Did you name your dog Tommy the dog?)*	Maya monisha ko tana marti hai kyunki usne apne kutte ka naam tommy the dog rakha hai. *(Maya taunts Monisha on naming her dog Tommy the dog.)*	Maya kehti hai ki uske kutte ka naam tommy the dog rakha hai. *(Maya says that her dog's name is Tommy the dog.)*	Maya taunts monisha kyunki usne apne kutte ka naam tommy the dog rakha hai. *(Maya taunts Monisha that she has named her dog Tommy the dog.)*

Table 5.9: Generated samples from test set. The last utterance is the sarcastic utterance for each dialogue.

Human Evaluation. We sample a total of 25 random instances from the test set and ask 20 human evaluators (the evaluators are fluent in English and their age ranges in 25-30 years) to evaluate the generated explanations (on a scale of 1 to 5) on the following basis:

- **Coherence:** Checks the generated explanation for correct structure and grammar.
- **On topic:** Measures the extent to which the generated explanation revolves around the dialogue topic.
- **Capturing sarcasm:** Estimates the level of emitted sarcasm being captured in the generated output.

	Coherency	**On topic**	**Capturing sarcasm**
mBART	2.57	2.66	2.15
BART	2.73	2.56	2.18
MAF	3.03	3.11	2.77
MOSES	**3.96**	**3.27**	**3.10**

Table 5.10: Human evaluation for the SED task comparing MAF and MOSES model with BART based systems.

We show the average score for the human evaluation parameters in Table 5.10. As illustrated, the proposed MOSES model exhibits more coherent, on topic, and sarcasm related explanations. However, there is still a scope for improvement, which can be taken up as future work.

5.5.6 Results for Monolingual English

Textual: Table 5.11 shows the results on MUStARD obtained when textual systems are used to obtain the generated explanations. Just as we noticed with WITS, PGN delivers us with the least performance across most metrics while BART-based representations outperform the rest by providing the best performance across the majority of the metrics.

Multimodality: We experimented with MAF-TAV and MOSES to estimate the effect of multimodality. Table 5.11 exhibits that, unline WITS, the addition of acoustic signals results in performance gain, while adding visual cues boosts the performance further across all metrics. Finally, when we add all multimodal signals together and observe the best performance for both, MAF and MOSES.

Modality Spotlight: As hypothesised, we obtain the best performance for sarcasm understanding when all the three modalities are used in tandem. We argue that especially in the case of sarcasm, multimodal signals such as audio and video might play the principal role in many instances. To comprehend this rotating importance of modalities, we use the spotlight module that aims to treat each modality as the primary modality while calculating the final representation. Similar to WITS, we observe an increase of $\sim 2\%$ across all evaluation metrics for MUStARD as well, as shown in Table 5.11. These results directly support our hypothesis of the effect of multimodality in sarcasm analysis. Moreover, we believe that this increase comes primarily from the modality spotlight and little from pronunciation embedding as there is hardly any spelling variation in English texts.

	Model	R1	R2	RL	B1	B2	B3	B4	M
Textual	**RNN**	24.71	7.58	21.93	24.08	9.87	1.61	0.63	17.98
	Transformers	26.62	12.41	22.95	26.02	11.76	3.86	0.67	19.74
	PGN	23.06	9.84	19.64	22.10	8.61	0.16	0.86	16.22
	mBART	27.13	13.71	24.53	26.17	12.23	3.90	2.82	20.61
	BART	28.94	13.78	26.90	28.44	12.85	7.84	4.17	21.29
Multimodal	**MAF-TA**	31.01	14.75	28.74	30.61	14.39	9.32	6.82	23.74
	MAF-TV	31.89	14.70	28.86	31.67	15.69	10.16	6.80	24.59
	MAF-TAV	32.95	15.17	30.92	32.80	16.84	10.46	7.53	26.46
	MOSES-TA	32.07	17.05	28.74	31.80	15.51	11.94	7.43	24.67
	MOSES-TV	34.11	16.31	30.58	**33.65**	17.67	12.27	9.89	27.19
	MOSES-TAV	34.86	**21.50**	31.12	34.42	17.90	11.67	**9.92**	28.14
	MOSES	**36.73**	18.21	**33.16**	32.84	**19.21**	**12.54**	9.12	**28.38**

Table 5.11: Experimental results on MUStARD (Abbreviation: R1/2/L: ROUGE1/2/L; B1/2/3/4: BLEU1/2/3/4; M: METEOR; PGN: Pointer Generator Network). Final row denotes MOSES including the pronunciation and spotlight modules.

5.5.7 Understanding Affects with Explanation

We study three understanding tasks in dialogues – sarcasm detection, humour identification, and emotion recognition using sWITS, hWITS, and eWITS, respectively. A trained SED system is used to obtain the explanations for all the instances present in these datasets. To verify our hypothesis that sarcasm explanation helps affect understanding, we perform experiments with and without explanations, as explained in the subsequent sections.

Sarcasm Detection. We take a base RoBERTa model (120) and perform the task of sarcasm detection over sWITS. The experimentation is performed using three setups as described below:

1. When we do not provide any utterance explanation to the input dialogue.
2. When we provide utterances appended with their generated explanation at the training time. Plain dialogues are given at the testing time in this case.
3. When dialogue instances are appended with their corresponding explanations during train and test time.

Table 5.12 illustrates the results we obtain for all the settings for MAF and MOSES. As can be seen, RoBERTa obtains 62% F1 score when we do not use any explanations. However, with the use of the generated explanations by MOSES during the train time, we obtain an improvement of 6% F1-score. On the other hand, the best performance is achieved by the last case, where the input instances are appended with their corresponding explanations both at the train and test time, with an increase of 8% F1-score. Consistent to the results obtained by MOSES's generation, MAF also reports an improved performance over no explanation model. However, the improvement shown by MAF is not at par with the improvement obtained by MOSES. These results directly support our hypothesis that utterance explanations can assist an efficient detection of sarcasm in the input instances.

Model	Use of Expl		Sarcasm				Humour				Emotion		
	Train	**Test**	**P**	**R**	**F1**	**Acc**	**P**	**R**	**F1**	**Acc**	**P**	**R**	**F1**
None	**0**	**0**	0.57	0.68	0.62	0.57	0.69	0.78	0.73	0.87	0.8	0.78	0.78
MAF	**1**	**0**	0.58	0.73	0.65	0.6	0.57	**0.87**	0.69	0.81	0.78	0.78	0.78
	1	**1**	0.66	0.77	0.71	0.68	0.73	0.71	0.72	0.87	0.78	**0.81**	0.79
MOSES	**1**	**0**	0.65	0.71	0.68	0.66	**0.84**	0.63	0.72	**0.89**	0.79	0.78	0.78
	1	**1**	**0.70**	**0.83**	**0.76**	**0.73**	0.72	0.77	**0.75**	0.88	**0.81**	0.80	**0.80**

Table 5.12: Experimental results on RoBERTa base when explanations generated by MOSES and MAF are used for completing the respective tasks. The first row indicates the performance without explanation.

Humour Identification. Another RoBERTa base is used to perform humour identification on hWITS. As for sarcasm detection, humour identification is also evaluated for the three setups described in the previous section. Table 5.12 illustrates the results obtained for the described setups. When no explanations are used during the training or testing time, we get an F1-score of 73%. This score is comparable to the performance we get when input instances are appended with their corresponding explanations generated by MOSES at the training time. This performance is boosted by 3% when the explanations are provided at the train/test time. However, it is important to note that the explanations generated by the MAF model resulted in a slightly decreased performance indicating the superiority of MOSES.

Emotion Recognition. Table 5.12 illustrates the results obtained for the task of emotion recognition on eWITS. We see the same value for the weighted F1 when we add explanations during the training phase of the system for both MAFand MOSES. However, when explanations assist both the training and testing phase, we observe an increase of 2% in the weighted F1 score for MOSES and 1% increase for MAF, indicating the positive effect explanations deliver for emotion recognition.

Error Analysis

Quantitative To capture the improvement exhibited by explanations in affect understanding, we show the confusion matrices emitted by the understanding models with and without using explanations. Table 5.13 illustrates these matrices – and as can be seen, the methods with explanation obtains higher true positive rate with a decreased false positive and false negative rates for majority of the classes among sarcasm, humour, and emotion labels.

	NS	**S**
NS	137/100	81/117
S	39/70	185/153

(a) Sarcasm detection on sWITS.

	NH	**H**
NH	335/330	32/37
H	24/23	82/83

(b) Humour identification on hWITS.

	Neutral	**Sadness**	**Joy**	**Anger**
Neutral	148/137	13/23	18/19	16/16
Sadness	5/2	62/66	3/2	0/0
Joy	7/5	10/9	120/124	4/3
Anger	0/9	0/1	8/9	50/48

(c) Emotion recognition on eWITS.

Table 5.13: Confusion matrix of the systems with and without (with/without) explanations.

Qualitative. While quantitative results confirm that explanations assist in identifying affects efficiently, qualitative analysis can further corroborate this hypothesis. Table 5.14 shows one instance from the test set where the presence of explanation helps for all affective tasks in question. More such examples can be found in the supplementary.

Dialogue	**MAYA:** And this time I thought lets have a theme party! *(And this time I thought lets have a theme party!)* **MONISHA:** Animals! Hum log sab animals banenge! *(Animals! Let us all be animals this time!)* **MAYA:** Mai hiran, Sahil horse, and Monisha chhipakalee! *(I'll be a deer, Sahil a horse, and Monisha can be a lizard!)*		
Exp	Maya Monisha ko animal keh ke taunt maarti hai. *(Maya taunts Monisha by calling her an animal)*		
	Sarcasm	**Humour**	**Emotion**
GT	1	0	Anger
w/o Exp	0	1	Neutral
w Exp	1	0	Anger

Table 5.14: True and predicted labels for the three affect tasks with and without using MOSES's explanation.

5.5.8 Evaluating LLMs

We experiment with models such as BART and T5 in the main results of this chapter. However, with the rise in popularity and the ease of availability of larger language models such as Llama, it becomes imperative for us to compare our models' performance with these LLMs. For comparison, we consider Llama, and perform the task of sarcasm explanation for monolingual English and Hindi-English code-mixed conversations. Table 5.15 illustrates this comparison. Due to the size of Llama, it outperforms the tested BART and T5 in all of the observed metrics.

	Model	R1	R2	RL	B1	B2	B3	B4	M
Code-mixed	**MAF**	39.69	17.10	37.37	33.20	18.69	12.37	8.58	30.40
	MOSES	42.17	20.38	39.66	34.95	21.47	15.47	11.45	32.37
	Llama	**45.61**	**23.16**	**40.34**	**35.98**	**23.14**	**17.27**	**12.54**	**34.41**
English	**MAF**	32.95	15.17	30.92	32.80	16.84	10.46	7.53	26.46
	MOSES	36.73	18.21	33.16	32.84	19.21	12.54	9.12	28.38
	Llama	**38.64**	**19.73**	**35.24**	**33.71**	**21.84**	**14.76**	**10.36**	**30.74**

Table 5.15: Performance of Llama when compared with our proposed methodologies for sarcasm explanation.

5.6 Conclusion

In summary, this chapter explored the complex domain of conversational affective understanding, focusing on discerning nuanced qualities like sarcasm, humour, and emotions. While sarcasm identification has made strides, understanding the underlying sarcastic nuances remains a challenge. To tackle this, we introduced the Sarcasm Explanation in Dialogue (SED) task within a multimodal and multilingual framework. Anchored by the WITS dataset, meticulously crafted to support SED, we proposed two deep neural networks, MAF and MOSES. These models process multimodal sarcastic dialogue instances to generate coherent natural language explanations. Our evaluation showcased the superiority of MAF and MOSES over state-of-the-art systems for SED across various metrics like ROUGE, BLEU, and BERTScore. Moreover, leveraging these explanations led to significant improvements in downstream tasks such as sarcasm detection, humour identification, and emotion recognition, demonstrating an average F1-score enhancement of approximately 14% in sarcasm detection and around 2% in humour identification and emotion recognition. Detailed analyses underscored the robustness of our findings, highlighting the potential of Sarcasm Explanation in Dialogue as a crucial step towards more nuanced and context-aware conversational systems.

Part III

Personalising Dialogues

6. Speaker Profiling

In the context of conversational dynamics, individuals manifest idiosyncratic behaviors, rendering a uniform, *one-size-fits-all* strategy inadequate for the generation of responses by dialogue agents. While previous studies have endeavored to craft personalized dialogue agents harnessing speaker persona data, their reliance on the presupposition that the speaker's persona is readily available is not universally applicable. To this end, we aim to bridge this existing gap by embarking on an exploration of the Speaker Profiling in Conversations (SPC) task. We tackle this task using two distinct approaches for two different language settings. For monolingual English, we consider the fundamental aim of SPC to be the creation of a concise synthesis of persona attributes for each individual participant within a dialogue. This version of SPC can be dissected into three core subtasks: persona discovery, persona-type classification, and persona-value extraction. Within a given dialogue, the initial subtask is centered on the identification of all utterances containing persona-related information. Subsequently, the second subtask scrutinizes these utterances, elucidating the specific category of persona details they encapsulate. Finally, the third subtask unearths the precise persona values corresponding to each ascertained category. To tackle the multifaceted challenge of SPC, we have meticulously assembled a novel dataset, SPICE, annotated with explicit profiling labels. This dataset serves as the foundation for our careful evaluation of diverse baseline models. Furthermore, we benchmark these results against a novel neural model, SPOT. Additionally, we present an exhaustive analysis of SPOT, encompassing a nuanced assessment of both quantitative and qualitative advantages and limitations exhibited by its constituent modules. For the other variant of SPC, we consider Hindi-English code-mixed conversations, and propose an unsupervised method to extract speaker profiles on the go while trying to improve response generation. We introduce a novel approach centered on harnessing the Big Five personality traits acquired in an unsupervised manner from the conversations to bolster the performance of response generation. These inferred personality attributes are seamlessly woven into the fabric of the dialogue context, using a novel fusion mechanism, PA3. It uses an effective two-step attention formulation to fuse the dialogue and personality information. This fusion not only enhances the contextual relevance of generated responses but also elevates the overall performance of the model. Our experimental results, grounded in a dataset comprising of multi-party Hindi-English code-mix conversations, highlight the substantial advantages offered by personality-infused models over their conventional counterparts. This is evident in the increase observed in ROUGE and BLUE scores for the response generation task when the identified personality is seamlessly integrated into the dialogue context. Qualitative assessment for personality identification and response generation aligns well with our quantitative results.

6.1 Introduction

Understanding natural language inputs is crucial for effective processing, as evidenced by a substantial body of work dedicated to the analysis of standalone textual content (240; 241; 242; 243). However,

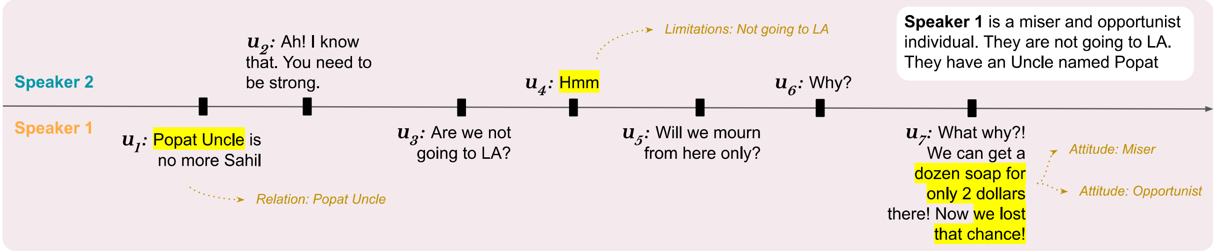

Speaker 2
Speaker 1
u_1: Popat Uncle is no more Sahil
Relation: Popat Uncle
u_2: Ah! I know that. You need to be strong.
u_3: Are we not going to LA?
u_4: Hmm
Limitations: Not going to LA
u_5: Will we mourn from here only?
u_6: Why?
u_7: What why?! We can get a dozen soap for only 2 dollars there! Now we lost that chance!
Attitude: Miser
Attitude: Opportunist
Speaker 1 is a miser and opportunist individual. They are not going to LA. They have an Uncle named Popat

Figure 6.1: Example of speaker profiling in conversation.

recent research has shifted towards contextual conversational data, emphasizing the need for mutual understanding among speakers and leading to extensive investigations in emotional analysis (244; 245; 246), intent discernment (247; 248), and dialogue act detection (249; 250). This change is driven by the growing prevalence of dialogue agents, necessitating contextually appropriate response generation. In this context, research has explored the engagement of participants, including empathetic (251; 252; 253) and stylistic dialogue generation (254; 255; 256). While such agents enhance system appeal, there is a need to address personalized dialogue generation, incorporating users' personas as essential inputs (70; 63; 64; 65; 66). Although persona details improve response intuitiveness and engagement (70; 63; 64; 65), the studies in this domain assume prior persona provision, a rarity in practical applications.

To tackle the challenge of persona information unavailability within chatbots, we embark on the task of **Speaker Profiling in Conversations** *aka* **SPC** as shown in Figure 6.1. We explore the task of SPC for monolingual English and Hindi-English code-mixed dialogues. For monolingual English, we consider SPC as a task geared towards the creation of comprehensive profiles for all participants engaged in a conversation, encompassing various speaker-centric attributes, including traits, likes, and occupation. On the other hand, we focus on the Big-five personality traits as possible personalities for Hindi-English code-mixed conversations. We also focus on improving response generation in this setting. We highlight each of these settings below.

Speaker Profiling for Monolingual English. The intricate task of SPC for English language unfolds into a triad of subtasks: *persona discovery*, *persona-type identification*, and *persona-value extraction*. In the first subtask, the objective is to discern which utterances within the conversation harbor persona-related insights. Subsequently, the second subtask entails the discernment of the specific persona information type within each identified utterance. Finally, the last subtask involves the meticulous extraction of precise values associated with each recognized persona type. To bolster research efforts in this domain, we present SPICE[1], a novel dataset teeming with multi-party conversations, thoughtfully adorned with annotated labels for all three subtasks. Complementing this, we introduce SPOT[2], a neural methodology that amalgamates RoBERTa (257), Transformer (150), and attention based methods, adept at capturing both the minutiae of dialogue-level context and the nuances of speaker-specific context for persona discovery. In our rigorous evaluation, SPOT outshines four baseline approaches, both in standalone and pipeline configurations, excelling in both subtasks. To gain deeper insights into its efficacy, we conduct a comprehensive analysis of the discrete components of SPOT, thereby affording a more nuanced understanding of its strengths and limitations.

Speaker Profiling for Hindi-English Code-mixed Conversations. Personality traits, by their very nature, span a vast spectrum and thus possess the potential for infinite possibilities (258). Numerous studies have been conducted to quantify these traits (259; 260; 261), with the Big Five personality traits (262) emerging as the prominent framework in this context. This theory distils human personality into five distinctive

[1] SPICE: **S**peaker **P**rofiling **I**n **C**onv**E**rsation
[2] SPOT: **S**peaker **P**r**O**filing using **T**ransformers

dimensions: Openness (OPN), Conscientiousness (CON), Extraversion (EXT), Agreeableness (AGR), and Neuroticism (NEU), in which each dimension encapsulates a pivotal facet of an individual's character. For instance, elevated levels of openness may signify a predisposition towards imagination. Here, we adopt this widely accepted model as the foundation for characterizing a speaker's personality. Our central hypothesis contends that incorporating personality indicators within the response generation process plays a pivotal role in generating contextually appropriate responses to given queries. Given the intricate and non-generalizable nature of manually annotating personality traits, we propose an unsupervised learning approach to acquire these traits, which, in turn, enhances response generation capabilities. We introduce PA3, a neural architecture to improve response generation while identifying appropriate personalities on the fly. Our evaluation shows that PA3 delivers superior performance for the task of response generation, when compared with comparable systems.

In a nutshell, our contributions are enlisted below, for both the settings:

- We explore the problem of Speaker Profiling in Conversations from two angles, where given a dialogue as input, the task is to extract the speaker-centric personality information of all speakers present in the dialogue.
- For English SPC, we curate SPICE, a multi-party conversation dataset with human annotated SPC labels.
- We benchmark SPICE with a RoBERTa and attention based novel model, SPOT for the SPC task.
- We explore the task of personality-aided code-mixed response generation.
- For code-mixed SPC, we propose an unsupervised mechanism to identify speakers' personality traits and leverage them for better response generation.
- We propose a novel method, PA3[3], which combines the identified traits with dialogue context to generate responses.
- Our quantitative and qualitative analyses show the benefits of identifying personality traits and including them in code-mixed response generation.
- We perform a comparative analysis of our models with several baselines and establish the superiority of our systems.

6.2 Related Work

Personalised Dialogue Systems. Several studies have shown that incorporating customization in dialogue systems improves their performance (63; 64; 65; 66; 67; 68). In the case of goal-oriented dialogue systems, various studies have been conditioned upon the user's profile to adjust responses (67; 68); however, recently, the focus has shifted to chit-chat setting. Personalization in vector form was used in some early projects. For example, a study (69) captured Twitter user's persona by learning distributed embeddings for each user to incorporate individual characteristics such as writing style or past experience. Later, (70) released a dataset, called PersonaChat which contains dyadic dialogues where the speakers are assigned fictitious persona, and are required to converse accordingly. Following this, a few studies showed the benefit of the PersonaChat dataset for personalization while generating replies to the user (63; 64; 65). Although leveraging persona information for dialogue generation results in better responses, it is vital not to presume that this information is freely available.

Persona Identification in Dialogues. An intitial study (263) investigated the extraction of speaker qualities from conversations. They proposed the MovieChAtt dataset and completed their objective using additional datasets such as PersonaChat and Reddit. However, the authors only evaluated *profession*, *gender*, and *family status* as possible speaker attributes. Another work (264) used a two-stage attribute

[3] Personality-Aware Axial Attention

extractor to extract user attributes as triplets of (subject, predicate, object). The dataset and model they utilised, however, were designed for dyadic interactions and cannot be expanded to a multi-party scenario. Recently, (265) used the PersonaChat dataset and offered the task of speaker persona detection, which classifies each speaker into one of the predefined personas.

Conversation and Code-mixing. Dialogues represent a well-established domain in NLP, having undergone extensive exploration (266; 267). However, the bulk of this research has predominantly revolved around monolingual text, despite the enduring prevalence of code-mixing, a timeworn linguistic phenomenon (17). Consequently, recent years have witnessed a surge in studies dedicated to unravelling the intricacies of code-mixing within dialogues (268). These investigations have honed in on exploring various nuances of code-mixed dialogues, delving into attributes such as intents (269; 270), the presence of hate speech (271; 106), humour (191; 272), and sarcasm (272; 103). Yet, the landscape for the generative dimension of code-mixing remains relatively uncharted, with limited concerted efforts in this direction.

Response Generation. For dialogue agents, it is of paramount importance to keep the conversation engaging (273). Consequently, generating apt responses becomes a primary field of research in terms of dialogue analysis. Many studies have been conducted to generate the right responses for monolingual English dialogues (274; 275; 276). However, response generation in the code-mixed setting remains a comparatively unexplored topic with only a handful of existing studies (277; 278). (279) illustrated that multilingual speakers prefer chatbots that can code-mix, thus making code-mixed response generation crucial.

Big Five Personality Traits. In pursuit of a deeper understanding of the user's personality, a range of studies have delved into the realm of the Big Five personality (280; 281). Numerous studies endeavored to categorize individuals into one of these personality archetypes based on their salient attributes (282; 283; 284; 285). A few studies have also attempted to use different personality theories other than the Big Five personality traits such as MBTI (259; 286).

6.3 Dataset

In this chapter, we handle the task of SPC in two different ways and linguistic settings. Consequently, we utilise two distinct datasets for benchmarking, which we describe in detail below.

6.3.1 Dataset for SPC in Monolingual English Dialogues

Set	#Dlg	#Utt	#Sp/Dlg	#P Utt	#P Utt/Dlg
Train	1039	9989	2.70	1005	0.96
Dev	114	1109	3.01	109	0.96
Test	280	1983	2.66	305	1.09

Set	#Persona Slot				
	Trait	Likes	Relation	Occ	Misc
Train	389	244	107	89	179
Dev	32	36	10	10	24
Test	120	88	28	18	53

Table 6.1: Statistics of SPICE. (Dlg: Dialogue; Utt: Utterance; Sp: Speaker; P Utt: Persona Utterance)

We introduce a new dataset, SPICE, tailored for speaker profiling in multi-party dialogues. Leveraging conversations extracted from the MELD dataset (71), we meticulously annotate each utterance for our designated task. Following MELD's original train-dev-test distribution, we undertake three subtasks for annotation.

- *Persona discovery*: We identify the presence of persona information in each utterance of the dialogue by marking it as 'yes' in this subtask.

- *Persona-type identification*: We associate a type of persona with each utterance marked as 'yes' in the previous phase, within this subtask. Following a comprehensive analysis of each conversation in the dataset, we define five persona types - *trait*, *likes*, *relation*, *occupation*, and *misc* - to encapsulate various personality characteristics of the speakers.
- *Persona-value extraction*: In this subtask, we extract persona values from the given instance for each identified persona type. These values may include a span from the input (e.g., for *occupation*), reference to another speaker present in the conversation (e.g., for *relationship*), or something inferred from the context (e.g., for *trait*).

Three annotators[4] were engaged in annotating SPICE. The initial two annotators assigned relevant persona labels to dialogue utterances, with any discrepancies resolved by the third annotator. Inter-annotator agreement was assessed using Krippendorff's Alpha (108). For persona discovery, an inter-annotator agreement score of 0.83 was attained, while persona-type identification achieved an agreement score of 0.71. Refer to Table 6.1 for dataset statistics, including the persona type distribution within SPICE.

Annotation guidelines for SPICE

We employ three annotators, all aged between 20-35 years of age and are fluent in English and Hindi, to annotate the proposed dataset. The annotations are performed in three stages. In the first stage, the annotators are asked to identify all the utterances in each dialogue that contain any persona related information by marking them as 'yes' and all others as 'no'. In the second stage, we ask the annotators to consider only the utterances marked as 'yes' in the previous stage and identify the type of persona present in it (out of the five possible persona-types defined earlier). Finally, for each persona type, the annotators are asked to identify the value of the corresponding persona slot. These values can either be present as a span in the input or can be a result of an inference made by the annotator.

Each instance for the second stage is made up of a sequence of utterances $\{u_1, u_2, \cdots, u_i\}$, where the last utterance is the one marked as containing persona information in the first stage. The annotators are given the following guidelines to decide the persona-type for the utterance u_i:

- ***Trait:*** Mark the utterance u_i as containing trait persona information if the information gathered from utterances $\{u_1, u_2, \cdots, u_i\}$ indicates towards the speaker having any type of a distinguishing quality.
- ***Likes:*** The utterance u_i contains persona information about likes if the utterances from u_1 to u_i indicate the concerned speaker finds someone/something pleasant or enjoyable.
- ***Occupation:*** If until utterance u_i we have enough evidence to conclude the profession of the speaker, we mark that utterance as containing information about the persona type occupation.
- ***Relation:*** Mark an utterance u_i as having the persona information of the speaker's relationship if enough evidence has been found from context of them being related to someone either biologically (mother, father) or non-biologically (friend, spouse).
- ***Miscellaneous:*** If the annotators find evidence of some important perpetual information related to the speaker (like education, and important dates) present in utterance u_i and if this information does not fall under any other identified persona types, then they mark it as miscellaneous.

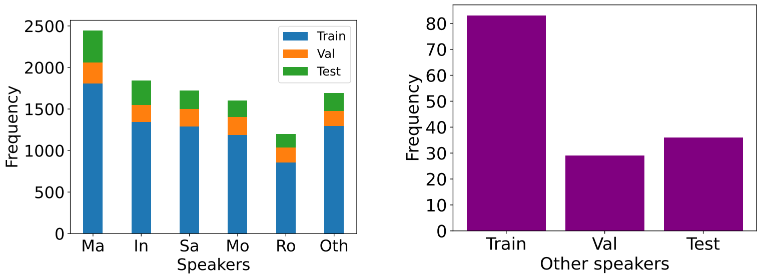
6.3.2 Dataset for SPC in Hindi-English Dialogues

Datasets for code-mixed conversations are inadequate, especially for multi-turn, multi-party conversations. In this study, we consider the MaSaC dataset We extract the conversations from this dataset and construct our response generation instances. We highlight the critical statistics of the dataset in Table 6.2. The

[4] They were NLP researchers or linguistics by profession; and their age ranges between $20 - 45$ years.

Set	#Dlgs	#Utts	Avg sp/dlg	Utt len		Vocab len	
				Avg	Max	English	Hindi
Train	8506	8506	3.60	10.82	113	3157	14803
Val	45	1354	4.13	10.12	218		
Test	56	1580	4.32	10.61	84		
Total	8607	11440	12.05	31.55	415		

Table 6.2: Statistics of MaSaC.



(a) Speaker distribution in the MaSaC dataset.

(b) Number of speakers other than the five primary speakers.

Figure 6.2: Dataset description of MaSaC (Abbreviation: Dlgs: Dialogues, Utts: Utterances, sp: speakers, Ma: Maya, In: Indravardhan, Sa: Sahil, Mo: Monisha, Ro: Rosesh, Oth: Others).

speaker distribution in Figure 6.2a and Figure 6.2b shows that there are five primary speakers in the dataset, each with varying personalities. Thus, aiding response generation with speaker personalities can improve its performance.

6.4 Methodology

Since we deal with two distinct views for the task of SPC using two distinct datasets, we propose two different novel approaches to accomplish our end goal.

6.4.1 SPOT: Solving SPC for Monolingual English

In this section, we illustrate SPOT, our proposed method to benchmark the task on SPICE. SPOT constitutes three subtasks – persona discovery, persona-type identification, and persona-value extraction.

Persona Discovery

In this stage, we employ a RoBERTa encoder (257) to capture dialogue-level contextual information, as illustrated in Figure 6.3. The model input comprises a sequence of utterances forming the dialogue, denoted as $D = \{u_1, u_2, ..., u_m\}$. Subsequently, the dialogue-level representations are fed into fully-connected layers for classification. It's noteworthy that SPICE, resembling a real world scenario, exhibits an inherent skew towards utterances lacking persona information. To address this imbalance, we apply the SMOTE upsampling technique (287) to boost the representation of persona-related utterances.

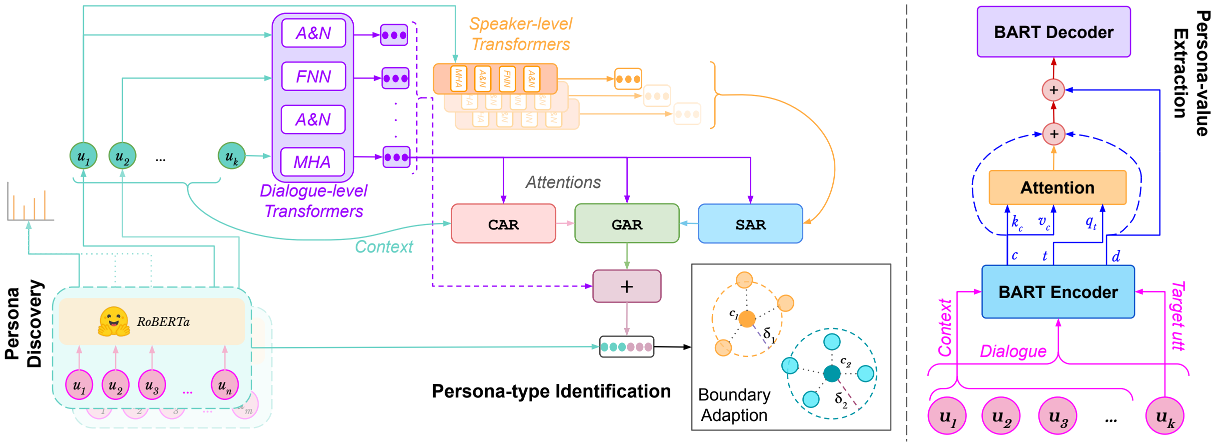


Figure 6.3: *Persona discovery*: The utterance representations obtained from dialogue-level RoBERTa are used for classification. *Persona-type identification*: Utterance representations are obtained from Dialogue-level Transformers and the speaker-specific Transformers. After receiving the representation from context, speaker, and global attention mechanism, the final representation is used to obtain adaptive decision boundary. We initialize the centroids $\{c_i\}_{i=1}^{K}$ and the radius of decision boundaries $\{\delta\}_{i=1}^{K}$ for each persona type and use the boundary loss for optimisation. *Persona-value extraction*: The context, target utterance, and the complete dialogue is transformed using a BART encoder following which attention is applied to get target attended vectors. Finally a concatenated vector is sent to the BART decoder for output generation.

Persona-type Identification

With a clear identification of persona-bearing utterances within the dialogue, the focus shifts to the subsequent subtask. This phase processes a sequence of utterances denoted as $I = \{u_1, u_2, ..., u_k\}$, where u_k signifies the target utterance, i.e., the one containing persona information, while $u_1, u_2, ..., u_{k-1}$ encompass the contextual utterances. For this task, our proposition integrates a fusion of RoBERTa and Transformer (150) elements, as depicted in Figure 6.3.

Dialogue representation. Every utterance u_j within the input sequence I undergoes processing via a Transformer layer to yield its representation.

Speaker-specific representation. In our quest to proficiently encapsulate the speaker sequence within a dialogue, we implement distinct Transformer encoder layers, one for each participating speaker in the discourse, thus yielding contextually tailored speaker-specific representations. Each of these speaker-specific encoders receives input from the utterance representations that are specific to the respective speaker i.

Attention. The creation of a speaker-aware representation, denoted as H_{SAR}, hinges on the judicious application of attention mechanisms (288) that facilitate the integration of speaker-specific representations with the target representation. Specifically, $H_{SAR} = Softmax(\frac{h_s h_t^T}{\sqrt{d_k}})h_s$. Similarly, H_{CAR} is computed by calculating the attention between the target representation, h_t, and context obtained from the RoBERTa model, h_c, i.e., $H_{CAR} = Softmax(\frac{h_c h_t^T}{\sqrt{d_k}})h_c$. Building upon this foundation, we embark on the synthesis of a global attention representation, denoted as H_{GAR}, serving as the cohesive fusion point for the speaker-aware and context-aware representations.

Adaptive decision boundary. Additionally, to capture the dialogue-level context effectively, we forward the RoBERTa embeddings to the model as a skip connection. Finally, we acquire the adaptive decision boundary for the persona classes through a k-means (289; 290) methodology. Moreover, for the optimization of decision boundaries specific to each class and to achieve maximal class separation, we employ the innovative boundary loss (291). The loss is computed using the following equation:

$$
\begin{aligned}
L_b = \frac{1}{N}\sum_{i=1}^{N}[\Delta_i(||z_i - c_{yi}||_2 - \delta_{yi})+ \\
(1-\Delta_i)(\delta_{yi} - ||z_i - c_{yi}||_2)]
\end{aligned}
$$

where N is the total number of samples in our set, z_i is the representation of the i^{th} instance, c_{yi} is the centroid for class yi, and δ_{yi} is the radius for class yi. Here,

$$
\Delta_i = \begin{cases} 1, if||z_i - c_{yi}||_2 > \delta_{yi} \\ 0, if||z_i - c_{yi}||_2 \leq \delta_{yi} \end{cases}
$$

Persona-value Extraction

In the culmination of shaping a speaker profile, the extraction of persona values for designated persona types within a dialogue emerges as the ultimate stride. Notably, these persona values can be conjectured from the input, often lacking specific constraints. Consequently, we adopt an encoder-decoder framework aligned with a generative objective to undertake this task effectively. This endeavor hinges upon adeptly encompassing the entirety of the conversation to grasp its essence, the contextual utterances to assimilate contextual knowledge, and the focal utterance, as it constitutes the primary wellspring for persona attributes.

In our approach, we employ a BART encoder (235) to meticulously encode the context, target, and dialogue utterances, resulting in c, t, and d, respectively. These representations undergo a pivotal phase where an attention mechanism amalgamates the key k_c and value v_c extracted from the context representation with the query q_t derived from the target utterance. This sophisticated interplay encapsulates the dynamic interaction between the target utterance and contextual utterances, thereby adeptly capturing the context-driven persona nuances embedded in the target utterance. The resultant representation is seamlessly integrated with the dialogue representation, and subsequently channeled into the BART decoder for the generation of output.

6.4.2 PA3: Solving SPC for Hindi-English Code-mixing

In this section, we discuss PA3, our proposed methodology for benchmarking SPC in code-mixed setting. Here, we focus on enhancing response generation with the foremost objective being the effective identification of personality attributes from the dialogue context. To achieve this, we propose an unsupervised technique that leverages response generation performance to improve personality identification. Subsequently, we fuse the personality attributes into the dialogue context to generate responses influenced by individual traits. We propose the incorporation of an intermediary module within the core encoder. This module leverages a straightforward yet effective two-step attention mechanism, facilitating the fusion of personality attributes with the representation of the dialogue. Broadly, we employ context-aware attention (114), which is instrumental in infusing personality characteristics into the key and value vectors of the dialogue. Subsequently, we employ Axial attention (292) to yield a refined, conclusive representation,

Trait	Templatic Definition
Openness	The speaker has high openness trait. They embrace new ideas, are curious about the world, and are often drawn to creative and unconventional pursuits.
Conscientious	The speaker has conscientiousness trait. They are reliable, organized, and detail-oriented, demonstrating a strong work ethic and a commitment to achieving their goals.
Extraversion	The speaker has extraversion trait. They thrive in social settings, energized by interactions with others, and enjoy being at the center of activities.
Agreebleness	The speaker has agreeableness trait. They prioritize cooperation, are empathetic, and often go out of their way to maintain harmonious relationships and help others.
Neuroticism	The speaker has high neuroticism trait. They have a greater tendency for emotional instability, anxiety, and a propensity to experience negative emotions such as fear, sadness, and anger.

Table 6.3: Personality traits in the Big Five personality model along with their templatic definitions.

which ultimately feeds into the decoder. Figure 6.5 provides a schematic diagram of our model. In the following subsections, we offer a comprehensive overview of individual modules.

Personality Identification

In this section, we describe our methodology for discerning the personality traits of each speaker and subsequently mapping them to their corresponding trait definitions. Although multiple theories quantify a speaker's personality traits (259; 260; 261), existing NLP applications widely use the Big Five Personality theory (262). Consequently, we select this model for our study, encompassing five distinct personality dimensions as shown in Table 6.3, where one of these dimensions is presumed to be dominant. To find the most suitable personality trait for a speaker in a dialogue, we employ an approach similar to Word2Vec (293), where a *'pseudo'* task is implemented to facilitate the acquisition of word embeddings. In the context of personality identification, our *'pseudo'* task takes the form of response generation, where we seek to enhance the generated response based on the intermediary step of personality identification. Figure 6.4 gives an overview of our mechanism for personality identification. We employ RoBERTa base (257) to classify personalities attributed to the target speaker, using the input dialogue as the primary data source. Once the personality is identified, it is subsequently linked to its templatic definition — a descriptive representation of the speaker's character, as outlined in Table 6.3. This personality definition is presented alongside the input dialogue to an encoder for further steps in the proposed pipeline.

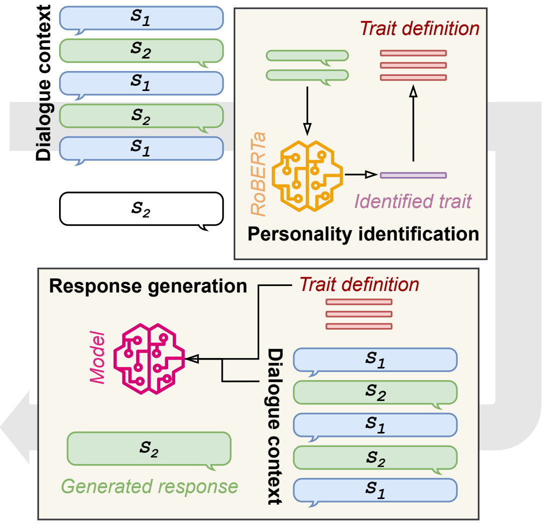


Figure 6.4: Outline of learning personality traits using the *'pseudo'* task of response generation.

Personality-Aware Attention (PAA). With the personality definition and the input dialogue at our disposal, our next step is to seamlessly integrate the personality information with the dialogue information to craft a suitable response. Conventional attention-based fusion mechanisms often facilitate a direct interplay between the input representations, in which one representation functions as the query while the

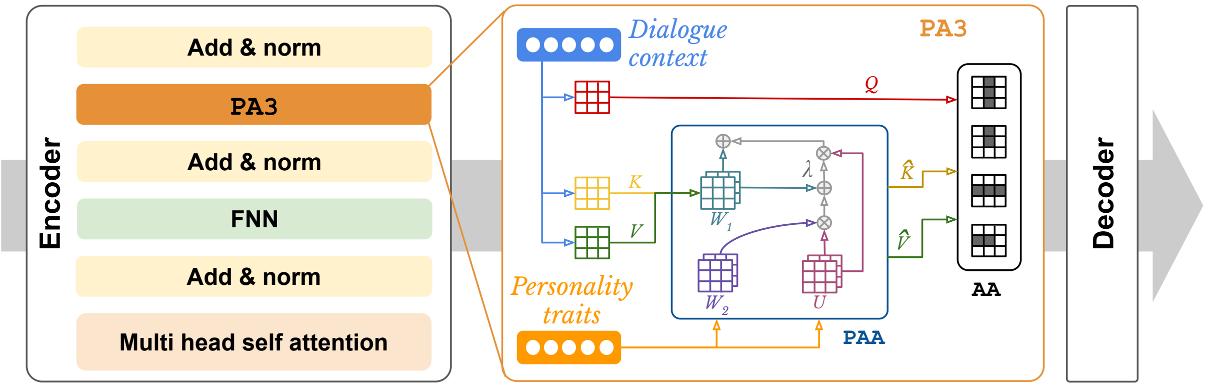


Figure 6.5: Model architecture to fuse personality values with dialogue context. The PA3 module can be injected into any encoder-decoder architecture, and it takes as inputs the dialogue representation along with the personality trait definition representation. First, context-aware attention is used to learn personality-infused key and value pairs and axial attention is then used to combine query, key, and value vectors into one final representation.

others assume the roles of key and value. However, as each representation captures distinct attributes, straightforward fusion may not preserve the optimal contextual information and could introduce significant noise into the final representations. Consequently, we introduce personality-aware attention (PAA) fusion employing context-aware attention (114). Our method entails the initial generation of personality-conditioned key and value vectors, followed by applying axial attention (292) to obtain the final fused values. We explain the process in detail below.

For an encoder model, we have the intermediate representation H at a specific layer to compute the query, key, and value vectors denoted as Q, K, and V respectively, in $\mathbb{R}^{n\times d}$ as outlined in Equation 6.1. $W_Q, W_K, and W_V$ are model parameters each with dimensions of $\mathbb{R}^{d\times d}$. In this context, n signifies the maximum sequence length of the text, while d represents the dimensionality of the dialogue vector.

$$[QKV] = H\,[W_Q W_K W_V] \tag{6.1}$$

The vector P in $\mathbb{R}^{n\times d_p}$, the encoded personality vector is used to create personality-influenced key and value vectors, $\hat{K}$ and $\hat{V}$ respectively, based on the method outlined by (114). For balancing of information from the personality source and information retention from the dialogue, we train a matrix λ in $\mathbb{R}^{n\times 1}$ (Equation 6.3). U_k and U_v in $\mathbb{R}^{d_p\times d}$ are matrices that can be learned.

$$\begin{bmatrix}\hat{K}\\ \hat{V}\end{bmatrix} = (1 - \begin{bmatrix}\lambda_k\\ \lambda_v\end{bmatrix}) \begin{bmatrix}K\\ V\end{bmatrix} + \begin{bmatrix}\lambda_k\\ \lambda_v\end{bmatrix} (P \begin{bmatrix}U_k\\ U_v\end{bmatrix}) \tag{6.2}$$

Rather than setting λ_k and λ_v as hyperparameters, we allow the model to autonomously determine their values through a gating mechanism, as defined in Equation 6.3. Additionally, the matrices $W_{k_1}, W_{k_2}, W_{v_1}$, and W_{v_2}, each with dimensions $\mathbb{R}^{d\times 1}$, are trained in conjunction with the model.

$$\begin{bmatrix}\lambda_k\\ \lambda_v\end{bmatrix} = \sigma(\begin{bmatrix}K\\ V\end{bmatrix} \begin{bmatrix}W_{k_1}\\ W_{v_1}\end{bmatrix} + P \begin{bmatrix}U_k\\ U_v\end{bmatrix} \begin{bmatrix}W_{k_2}\\ W_{v_2}\end{bmatrix}) \tag{6.3}$$

Once we obtain the personality-infused key and value vectors, we use the Axial attention mechanism.

Axial Attention

Axial attention (292) finds its primary application in computer vision, where its utility extends to managing multidimensional tensors. The fundamental aim is to approach each axis independently, thereby comprehensively exploring relationships between the various dimensions. The proposed approach preserves the original shape of the multidimensional tensor, performing either masked or unmasked attention along a single axis at any given time. This specific operation, referred to as axial attention and denoted as $\text{Attention}_k(x)$, is responsible for directing attention over axis k within the tensor x. In doing so, it blends information across axis k while maintaining the independence of information along the remaining axes. Implementing axial attention for a given axis k involves a series of steps, such as transposing all axes except k to the batch axis, invoking standard attention as a subroutine, and reverting the transpose operation. Within our network architecture, we leverage two axial attention layers, culminating in the derivation of the ultimate dialogue representation denoted as $\hat{H}$, signifying the personality-infused representation of the dialogue, which is then passed on to the next encoder/decoder layer. For our input two dimensional arrays of $\hat{K}$, $\hat{V}$, and Q:

$$\hat{H} = \text{Attention}_k(\hat{K}, \hat{V}, Q) \tag{6.4}$$

6.5 Experiments and Results

6.5.1 Evaluating SPOT

We perform experiments for all three subtasks in two settings – standalone and pipeline. Following sections present both settings.

Standalone Evaluation. In this configuration, the distinct phases' models are individually trained and assessed. To elucidate, in the persona discovery phase, all dialogues undergo processing by SPOT, meticulously scrutinizing each utterance for persona-related cues. Transitioning to persona-type identification, we strictly adhere to the ground-truth, selectively forwarding solely the persona-laden utterances, alongside their contextual information, to the model, as visually depicted in Figure 6.3. Ultimately, when it comes to persona-value extraction, we furnish SPOT with the ground truth persona categories, together with the persona-imbued utterances and their contextual backdrop, thereby enabling the precise extraction of persona values.

Pipeline evaluation. In this context, the persona discovery process aligns with the standalone setup. Yet, in the persona-type identification phase, we exclusively supply the model with the utterances pinpointed as persona-bearing in the preceding subtask, without adhering to the ground truth. Subsequently, the results yielded from the second phase serve as input for the ultimate persona-value extraction task, devoid of any ground-truth reference.

Baseline methods. Given that persona discovery and persona-type identification are classification-oriented tasks, we have leveraged four classification baselines that are originally designed for akin tasks like emotion detection and dialogue-act identification.

- **BERT:** BERT (Bidirectional Encoder Representations from Transformers) (110) is encoder stack of transformer architecture (150). We use pre-trained BERT base and fine-tune it for our tasks.
- **RoBERTa:** RoBERTa (Robust BERT) (257) extends upon BERT by adjusting critical hyperparameters, eliminating the next-sentence pretraining task, and utilizing significantly larger mini-batches and learning rates during training.
- **DialogXL:** Shen et al. (246) modified XLNet by changing the segment-level recurrence mechanism to an utterance-level recurrence mechanism so that XLNet could be mapped to a dialogue

setting. They also incorporated dialogue-aware self-attention to capture the intra- and inter-speaker dependencies in a conversation.

- **Co-GAT:** Qin et al. (249) proposed a co-interactive graph interaction layer with cross-utterance and cross-tasks connections.
- **AGHMN:** Jiao et al. (245) used an attention-based GRU to monitor the flow of information through a hierarchical memory network. The attention weights are calculated over the contextual utterances in the conversation and combined for the final classification.

For the third subtask of persona-value extraction, we consider sequence-to-sequence models for comparison.

- **RNN:** OpenNMT5 provides with an implementation of the RNN seq-to-seq architecture which we use in our study.
- **Transformers:** We use the standard encoder and decoder stack to generate the output (150).
- **Pointer Generator Network (PGN):** The standard seq-to-seq architecture supporting both generation of new words as well as copying words from input (234).
- **BART:** BART (235) contains a bidirectional encoder and an auto-regressive decoder to create a denoising auto-encoder model.
- **T5:** T5 (294) is a seq-to-seq model trained on a mixture of unsupervised and supervised tasks.

Evaluation metrics. Since the first two subtasks are multi-class classification problems, we use F1 score as our choice of evaluation metric. We consider F1 score of the positive class for the task of *persona discovery*, while weighted F1 score is used for *persona type identification*. On the other hand, since the task of *persona-value extraction* follows a generative objective, we use the ROUGE (228) and the BLEU (229) scores to gauge the performance of the systems.

Results

Standalone evaluation. We evaluate SPOT for all subtasks of SPC separately and show the results in Table 6.4 and Table 6.6.

- *Persona discovery*: We train SPOT as a binary classifier using cross-entropy loss. We obtain 52% F1 score, which is $\sim 21\%$ better than the best baseline, DialogXL, as can be seen in Table 6.4. The gain in performance can be attributed to the efficient way we use different modules in our architecture to capture different essence of a conversation. It is interesting to observe that Co-GAT produce the best performance in terms of recall scores while SPOT holds a balance between precision and recall to obtain the highest F1 score, which is our metric of choice due to the class imbalance present in our data.
- *Persona-type identification*: We use boundary loss (291) to train SPOT for this task. Table 6.4 shows that SPOT yields a weighted average of 56% F1-score with the maximum score for persona-type *likes*. AGHMN, the best baseline, results in a weighted average of 48% F1-score, which is $\sim 8\%$ less than SPOT. We observe that SPOT achieves the best F1-score for all persona slots showcasing a global dominance of our system. It is interesting to observe that our model performs quite well across the persona slots that are dominantly present in our data and consistently decreases for the slots based on their availability in the data.
- *Persona-value extraction*: Using a generative objective, we obtain the results by SPOT for this subtask. Table 6.6 demonstrates that SPOT outperforms the baselines by around 1% for all metrics except BLEU 1 and BLEU 3.

Pipeline evaluation. Tables 6.5 and 6.6 show the performance obtained by our model along with the baseline scores. For the task of persona discovery, we obtain the same results as standalone due to the

Systems	*Persona Discovery*			*Persona-type Identification*					
	P	**R**	**F1**	**Trait**	**Likes**	**Rel**	**Misc**	**Occ**	**Wtd**
BERT	0.17	0.72	0.27	0.48	0.0	0.09	0.24	0.05	0.24
RoBERTa	0.20	0.56	0.29	0.51	0.17	0.11	0.26	0.04	0.27
DialogXL	0.45	0.23	0.31	0.52	0.0	0.0	0.0	0.0	0.18
Co-GAT	0.15	**0.94**	0.27	0.50	0.35	0.06	0.14	0.05	0.33
AGHMN	0.43	0.14	0.21	0.56	0.58	0.38	0.26	0.25	0.48
SPOT	**0.47**	0.58	**0.52**	**0.61**	**0.63**	**0.49**	**0.35**	**0.31**	**0.56**

Table 6.4: Comparative results for standalone evaluation. (P: Precison; R: Recall; Rel: Relationship; Occ: Occupation; Wtd: Weighted F1 score.)

Systems	*Persona Discovery*			*Persona-type Identification*					
	P	**R**	**F1**	**Trait**	**Likes**	**Rel**	**Misc**	**Occ**	**Wtd**
BERT	0.17	0.72	0.27	0.12	0.0	0.05	0.07	0.0	0.07
RoBERTa	0.20	0.56	0.29	0.14	0.02	0.07	0.09	0.02	0.09
DialogXL	0.45	0.23	0.31	0.33	0.0	0.0	0.0	0.0	0.07
Co-GAT	0.15	**0.94**	0.27	0.31	0.26	0.0	0.07	0.0	0.15
AGHMN	0.43	0.14	0.21	0.48	0.43	**0.41**	0.21	0.16	0.40
SPOT	**0.47**	0.58	**0.52**	**0.57**	**0.52**	0.38	**0.28**	**0.24**	**0.46**

Table 6.5: Comparative results for pipeline evaluation. (P: Precison; R: Recall; Rel: Relationship; Occ: Occupation; Wtd: Weighted F1 score.)

same type of input and evaluation strategies. However, we observe a performance drop of $\sim 10\%$ for the persona-type identification task and a drop of at most $\sim 6\%$ for the persona-value extraction task when compared to the standalone results. This is expected as the erroneous predictions from the previous stage may propagate to the next stage. Nevertheless, when compared to the baseline systems, our proposed mechanism gives the best score, with an increase of $\sim 6\%$ over the best baseline and $\sim 39\%$ over the worst performing baseline for the former task (c.f. Table 6.5). Apart from *relation*, SPOT performs the best for all persona slots. While for the last subtask, we obtain an improvement of $\sim 1\%$ over the baselines. Consequently, we establish that SPOT is able to capture the essence of persona more clearly when compared with the baseline systems.

Ablation Study

SPOT captures two primary aspects of a dialogue – the dialogue context and the speaker semantics. To capture the dialogue-level context, we use SPOT$_{Base}$, containing the dialogue-level Transformer followed by context H_{CAR}, and global attention representation H_{GAR}. That is, no speaker level information in captured in this variant. We reinforce the dialogue context by using the RoBERTa representation as a skip connection in this architecture, SPOT$_{Base+RoBERTa}$. Speaker semantics are captured by the speaker-specific Transformers and attention representation H_{SAR}. We add these modules in our final model, SPOT. We observe that the addition of the RoBERTa representation improves the performance of our model considerably (21%) while the addition of speaker module improves it further (7%) verifying the use of each module.

Error Analysis

In this section, we present a detailed analysis of the results obtained for SPOT. We first show the quantitative analysis by analysing the confusion

True \ **Predicted**	*No*	*Yes*
No	1191	487
Yes	121	**184**

(a) SPOT

True \ **Predicted**	*No*	*Yes*
No	1594	84
Yes	235	**70**

(b) DialogXL

Table 6.8: Confusion matrices for SPOT and DialogXL (best baseline) for the persona discovery task.

Models	Standalone					Pipeline				
	R1	**R2**	**B1**	**B2**	**B3**	**R1**	**R2**	**B1**	**B2**	**B3**
RNN	26.85	2.28	24.78	1.48	0.36	19.64	0.37	18.98	0.36	1.12
Transformer	26.02	2.37	23.93	1.54	0.58	19.48	0.70	18.82	0.69	2.05
PGN	24.40	1.59	23.12	1.08	0.36	17.11	0.49	16.13	0.33	**9.28**
BART	28.93	2.16	**27.23**	1.51	0.36	22.41	**1.01**	21.37	0.80	0.08
T5	15.07	0.0	14.90	2.25	**1.20**	11.62	0.37	11.51	0.36	1.12
SPOT	**29.51**	**2.97**	27.16	**2.27**	0.60	**23.40**	0.60	**22.12**	**1.12**	0.08

Table 6.6: Comparative results for standalone and pipeline evaluation for *persona-value extraction*. (R1/2: ROUGE1/2; B1/2/3: BLEU1/2/3)

Systems	Trait	Likes	Relation	Misc	Occ	Weighted
SPOT$_{Base}$	0.36	0.27	0.22	0.23	0.07	0.28
SPOT$_{Base+RoBERTA}$	0.56	0.53	0.45	0.40	0.26	0.49
SPOT	**0.61**	**0.63**	**0.49**	**0.35**	**0.31**	**0.56**

Table 6.7: Ablation results for *persona-type identification*. (Misc: Miscellaneous; Occ: Occupation)

matrices obtained. After this, we show a qualitative analysis by observing a few test samples and their predicted persona slots and values. We also pick some predicted examples to illustrate the shortcomings of our approach and give a possible direction for future research.

Quantitative Analysis We compare the confusion matrices obtained by SPOT and the best baseline for this subtask, AGHMN in Table 6.9. Both the models produce a comparable performance with maximum accurate predictions for the persona-type *trait*. Moreover, we observe that the classes *likes* and *trait* are most confused, followed by *likes*, *relation*, *misc* and *occupation* for SPOT as well as for AGHMN, while *occupation* and *relation* are least confused among all classes.

		Predicted				
		Trait	Occ	Misc	Likes	Relation
True	Trait	72/65	8/11	10/10	22/23	6/9
	Occ	4/5	8/6	3/3	2/3	1/1
	Misc	16/16	7/8	17/12	5/7	8/10
	Likes	24/25	2/5	3/4	55/51	4/3
	Relation	3/3	0/0	7/8	3/5	17/12

Table 6.9: Confusion matrices for the *persona-type identification* task. Each cell represents value like {SPOT/AGHM}.

Table 6.8 presents the confusion matrices for SPOT and the best performing baseline, DialogXL for the persona discovery task. SPOT correctly predicts 184 out of 305 positive instances (60.3%) while DialogXL is only able to predict 70 (22.9%). Although DialogXL performs poorly while identifying the true positives, it does a better job in identifying the true negatives. It is able to correctly classify 1594 instances as negative (94.9%), whereas SPOT predicts only 1191 true negative instances (70.9%).

Common Errors by SPOT: *False positives.* While attaining a decent value for true positives, SPOT obtains a significant value of false positives (487) for *persona discovery* (c.f. Table 6.8). We analyse the type of misclassified instances and observe that SPOT often identifies utterances containing questions as having persona information. For example, the utterance *'We're in a relationship?'* is marked as a positive instance for persona discovery when in true sense, its answer was the one carrying persona. Table 6.10 presents similar examples. This phenomenon can be attributed to the presence of words such as 'relationship' (instance 1), 'like' (instance 2), or 'father' (instance 4) in the utterances as these words may hint towards the presence of explicit persona information in a statement. In addition, SPOT frequently predicts the utterances expressing a temporary/trivial state for the speaker as containing persona information. For instance, the utterance *'Oh my god, I am losing my mind.'* is marked as the one containing persona information. We show more such instances in Table 6.10. Future work could be done to handle such cases of false positives.

#	Speaker	Utterance	PD	
			True	Pred
1	Chandler	We're in a relationship?	0	1
2	Danny	So you like the short hair better?	0	1
3	Rachel	Yeah. Oh! Was how you invented the cotton gin?!	0	1
4	Phoebe	Well, so, umm, anyway umm, I've been, I've been looking for my Father, and umm, have you heard from him, or seen him?	0	1
5	Janice	So, I hear, you hate me?	0	1

(a) Persona does not lie in questions.

#	Speaker	Utterance	PD	
			True	Pred
1	Monica	Oh my god, I am losing my mind.	0	1
2	Phoebe	Because we're girls.	0	1
3	Monica	No, Phoebe, I'll tell you what, if you get ready now I'll let you play it at the wedding.	0	1
4	Leslie	My best shoes, so good to me.	0	1
5	Chandler	Okay uh, for now, temporarily, you can call me, Clint.	0	1

(b) Persona is not temporary/trivial attributes.

Table 6.10: Examples of false positives by SPOT for the Persona Discovery (PD) task.

Common Errors by SPOT: *False negatives.* In addition to falsely identifying utterances containing no persona information as positive instances, SPOT identifies 121 true positive instances as belonging to the negative class. We analyse the misclassified positive instances and identify two situations where such misclassifications happen. When the persona information is present in the answer to a question, it is often misclassified by SPOT. For example, in the dialogue *'Ross: Okay! All right! Now, Chandler you-you wanna live with Monica, right? Chandler: Yeah, I do.'*, Chandler's utterance contains information about his persona (*relationship* with Monica), but SPOT is unable to identify this instance correctly. Table 6.11 highlights similar examples from SPICE. Furthermore, SPOT often misclassifies instances where the persona information is implicit in nature. For instance, the utterance *'Ya see, it's just, see I was a regular on a soap opera y'know?'* contains persona information (that the speaker's *occupation* is an actor); however, SPOT is not able to relate the phrase 'soap opera' to *occupation* and thus does not mark the instance as having persona information. Supporting examples are shown in Table 6.11.

#	#	Speaker	Utterance	PD	
				True	Pred
1	u_1	Ross	Okay! All right! Now, Chandler you-you wanna live with Monica, right?	0	1
	u_2	Chandler	Yeah, I do.	1	0
2	u_1	Judge	So based on your petition you are seeking an annulment on the grounds that Mr. Geller is mentally unstable?	1	1
	u_2	Ross	Fine, I'm mentally unstable.	1	0
3	u_1	Ross	Are you intrigued?	0	0
	u_2	Chandler	You're flingin'-flangin' right I am!	1	0
4	u_1	Rachel	Why, does she have a bad personality?	1	1
	u_2	Phoebe	Oh no, Bonnie's the best!	1	0
5	u_1	Chandler	Soo, ah, Eric, what kind of photography do ya do?	0	0
	u_2	Eric	Oh, mostly fashion, so there may be models here from time to time, I hope that's cool.	1	0

(a) Persona lies in answer to a question.

#	Speaker	Utterance	PD	
			True	Pred
1	Joey	Ya see, it's just, see I was a regular on a soap opera y'know?	1	0
2	Joey	Awww, one of my students got an audition. I'm so proud.	1	0
3	Joey	Yeah but we won't be able to like get up in the middle of the night and have those long talks about our feelings and the future.	1	0
4	Janice	Oh, Chandler, look. You and Monica are meant to have children. I am sure it's gonna be just fine.	1	0
5	Steve	Umm, see, I was thinking maybe you two could switch apartments because Phoebe's more our kind of people.	1	0

(b) Persona is implicit.

Table 6.11: Examples of false negatives by SPOT for the Persona Discovery (PD) task.

Qualitative Analysis This section presents a subjective analysis of the quality of predictions made by SPOT and the best baselines, based on a sample dialogue from the test set. The dialogue contains six utterances, where utterances u_1, u_3, and u_5 are identified as having the persona type *relationship*, as shown in Table 6.12. In the first subtask of persona discovery, SPOT correctly identifies two positive instances out of the total three, while the best baseline, DialogXL, only identifies one such instance. However, both SPOT and DialogXL misclassify one utterance as false negative. Moving on to the second subtask of persona-type identification, SPOT correctly classifies two instances of persona-type *relationship* but misclassifies one instance as *trait*. On the other hand, the best baseline, AGHMN, only predicts one correct class and misclassifies the others as *likes*.

#	Speaker	Utterance	*Persona Discovery*			*Persona Type Identification*		
			True	**Predicted**		**True**	**Predicted**	
				SPOT	**DialogXL**		**SPOT**	**AGHMN**
u_1	Rachel	Everybody, this is Paolo, Paolo, I want you to meet my friends. This is Monica	Yes	Yes	No	relationship	relationship	likes
u_2	Monica	Hi!	No	No	No	-	-	
u_3	Rachel	And Joey...	Yes	No	Yes	relationship	relationship	relationship
u_4	Monica	Hi!	No	No	No	-	-	
u_5	Rachel	And Ross...	Yes	Yes	Yes	relationship	trait	likes
u_6	Monica	Hi!	No	No	Yes	-	-	

Table 6.12: Actual and predicted labels for the *persona discovery* and *persona type identification* tasks. DialogXL and AGHM are the best performing baseline for the respective tasks.

6.5.2 Evaluating PA3

Sp	GT	Opn	Con	Ext	Ext	Neu
Ma	Con	14%	**54%**	8%	13%	11%
In	Agr	6%	18%	8%	**65%**	3%
Sa	Con	14%	**52%**	4%	16%	14%
Mo	Opn	**58%**	11%	21%	8%	2%
Ro	Ext	16%	14%	**51%**	15%	4%

Table 6.13: Percentage of times a personality trait is assigned to a speaker. (Abbr - Sp: Speakers, GT: Ground Truth, Ma: Maya, In: Indravardhan, Sa: Sahil, Mo: Monisha, Ro: Rosesh, Oth: Others)

Evaluation Metrics. Given the absence of ground-truth labels for evaluating personality detection, we resort to a manual assessment process, meticulously scrutinizing the outputs for the primary speakers to derive meaningful insights into the system's performance in this regard. To assess the response generation proficiency, we employ established evaluation metrics, specifically ROUGE (228) and BLEU (229) scores. These metrics are adept at quantifying the syntactic competence of the system in question. Additionally, we incorporate BERTScore (295), which serves to gauge the semantic aptitude of the system, and human evaluation provides a more comprehensive evaluation.

In this section, we present a comprehensive overview of the quantitative and qualitative results achieved by personality identification and response generation. Additionally, we offer a closer look at our ablation results, shedding light on the significance of each submodule within our proposed architectural framework for response generation. Further, human evaluation highlights the pros and cons of the generated responses and personalities.

Personality Identification

As shown in Figure 6.4, our initial step predicts the most suitable personality from the Big Five personality traits for the target speaker. To gauge the efficacy of our predicted personalities, we focus on the five primary speakers featured in the MaSaC dataset. Figure 6.2a shows the distribution of the speakers where it can be observed that the speakers — Maya, Indravardhan, Sahil, Monisha, and Rosesh, are the most frequently occurring speakers. We perform a manual evaluation of the personality predictions. Using information from Wikipedia[5], we procure character descriptions for each of the five prominent speakers which were given to five expert annotators.

The annotators then categorize each speaker within the Big Five personality framework. This annotator-driven classification enables the construction of a definitive ground-truth for evaluation encompassing the five speakers, each associated with an assigned personality trait value as shown in Table 6.13. We compare the obtained ground-truth personalities with the ones predicted by the RoBERTa model, an outcome of the *'pseudo'* task centred around response generation. The ensuing distribution of these predictions is laid out for scrutiny in both Table 6.13 and Figure 6.6. We can see that the personalities found most suitable by the human annotators are the ones preferred by the RoBERTa model, too, validating the performance of our system.

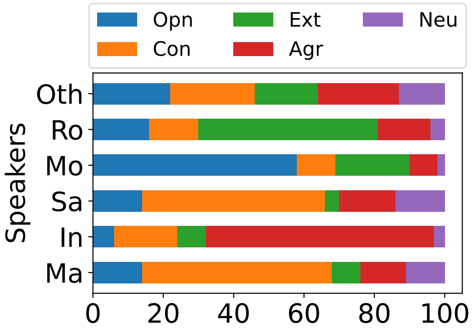


Figure 6.6: Distribution of the predicted personality traits assigned to different speakers (Abbr - Ma: Maya, In: Indravardhan, Sa: Sahil, Mo: Monisha, Ro: Rosesh, Oth: Others).

6.5.3 Response Generation

Here, we discuss the effect of adding personality information to the dialogue context quantitatively.

Comparative Systems

To attain the most promising textual representations for discourse, we employ a range of well-established encoder-decoder-based sequence-to-sequence (seq2seq) models.

- **RNN**: We leverage the RNN seq2seq architecture, implemented through openNMT4[6].
- **Pointer Generator Network (PGN)** (234): In this seq2seq architecture, a fusion of generative and copy mechanisms is harnessed, offering a versatile approach to content generation.
- **Transformer** (150): Responses are generated using the conventional Transformer encoder-decoder model.
- **T5** (296): We deploy the base version of the text-to-text-transfer-transformer (T5), which excels in framing multiple NLP tasks as text-to-text challenges, facilitating a unified and efficient approach to tasks such as translation, summarization, and question answering.
- **BART** (235): We utilize the basic denoising autoencoder model with a bidirectional encoder and a left-to-right auto-regressive decoder.
- **mBART** (236): mBART[7], trained on multiple extensive monolingual datasets, shares the same objective and architectural structure as BART.

	Model	R1	R2	RL	B1	B2	B3	B4	BS
w/o personality	**RNN**	8.17	0.02	8.09	5.11	0.01	0.11	0	54.16
	PGN	7.06	0	7.01	4.31	0	0.08	0	53.12
	Transformers	10.64	0.83	10.35	7.22	0.92	0.13	0.01	58.94
	mBART	11.36	1.23	10.9	7.91	1.01	0.21	0	61.02
	T5	11.87	1.01	11.43	8.41	1.02	0.17	0.02	61.98
	BART	12.94	1.66	12.34	9.66	1.64	0.43	0.07	63.12
w personality	$\textbf{RNN}_{\textbf{PA3}}$	9.96 (↑1.79)	0.08 (↑0.06)	10.71 (↑2.62)	6.87 (↑1.76)	1.04 (↑1.03)	0.43 (↑0.32)	0.22 (↑0.22)	56.24 (↑2.08)
	$\textbf{PGN}_{\textbf{PA3}}$	8.45 (↑1.39)	1.11 (↑1.11)	9.41 (↑2.40)	5.95 (↑1.64)	1.03 (↑1.03)	0.37 (↑0.29)	0.21 (↑0.21)	55.87 (↑2.75)
	$\textbf{Transformers}_{\textbf{PA3}}$	12.76 (↑2.12)	1.75 (↑0.92)	12.14 (↑1.79)	8.46 (↑1.24)	2.02 (↑1.10)	0.45 (↑0.32)	0.24 (↑0.23)	61.06 (↑2.12)
	$\textbf{mBART}_{\textbf{PA3}}$	13.43 (↑2.07)	2.36 (↑1.13)	12.15 (↑1.25)	8.89 (↑0.98)	**2.61** (↑1.60)	0.56 (↑0.35)	0.18 (↑0.18)	63.42 (↑2.40)
	$\textbf{T5}_{\textbf{SC}}$	12.02 (↑0.15)	1.51 (↑0.50)	11.98 (↑0.55)	8.52 (↑0.11)	1.51 (↑0.49)	0.39 (↑0.22)	0.11 (↑0.09)	62.05 (↑0.07)
	$\textbf{T5}_{\textbf{DPA}}$	12.04 (↑0.17)	1.56 (↑0.55)	12.01 (↑0.58)	8.58 (↑0.17)	1.58 (↑0.56)	0.41 (↑0.24)	0.14 (↑0.12)	62.35 (↑0.37)
	$\textbf{T5}_{\textbf{PA3}-\textbf{Axial}}$	12.79 (↑0.92)	1.64 (↑0.63)	12.53 (↑1.10)	9.04 (↑0.63)	1.96 (↑0.94)	0.46 (↑0.29)	0.18 (↑0.16)	62.99 (↑1.01)
	$\textbf{T5}_{\textbf{OT}}$	13.48 (↑1.61)	1.97 (↑0.96)	12.89 (↑1.46)	9.21 (↑0.80)	2.23 (↑1.21)	0.52 (↑0.35)	0.21 (↑0.19)	63.14 (↑1.16)
	$\textbf{T5}_{\textbf{PA3}}$	13.61 (↑1.74)	2.03 (↑1.02)	13.92 (↑2.49)	9.78 (↑1.37)	2.62 (↑1.60)	0.51 (↑0.34)	0.26 (↑0.24)	63.87 (↑1.89)
	$\textbf{BART}_{\textbf{SC}}$	13.05 (↑0.11)	1.89 (↑0.23)	12.64 (↑0.30)	9.84 (↑0.18)	1.82 (↑0.18)	0.52 (↑0.09)	0.12 (↑0.05)	63.48 (↑0.36)
	$\textbf{BART}_{\textbf{DPA}}$	13.12 (↑0.18)	1.98 (↑0.32)	12.81 (↑0.47)	9.96 (↑0.30)	1.94 (↑0.30)	0.54 (↑0.11)	0.15 (↑0.08)	63.82 (↑0.70)
	$\textbf{BART}_{\textbf{PA3}-\textbf{Axial}}$	13.97 (↑1.03)	2.21 (↑0.55)	13.05 (↑0.71)	10.16 (↑0.50)	2.07 (↑0.43)	0.61 (↑0.18)	0.18 (↑0.11)	64.34 (↑1.22)
	$\textbf{BART}_{\textbf{OT}}$	14.29 (↑1.35)	2.54 (↑0.88)	13.72 (↑1.38)	10.59 (↑0.93)	2.16 (↑0.52)	0.73 (↑0.30)	0.22 (↑0.15)	65.05 (↑1.93)
	$\textbf{BART}_{\textbf{PA3}}$	**14.67** (↑1.73)	**2.77** (↑1.11)	**14.11** (↑1.77)	**10.92** (↑1.26)	2.51 (↑0.87)	**0.73** (↑0.30)	**0.27** (↑0.20)	**65.93** (↑2.81)

Table 6.14: Experimental and ablation results for the response generation task with and without fusing personalities. (Abbr: R1/2/L: ROUGE-1/2/L, B1/2/3/4: BLEU-1/2/3/4, BS: BERTScore (F1 score), SC: Simple Concat, DPA: Dot Product Attention, OT: Only Traits, PA3: Personality-Aware Axial Attention).

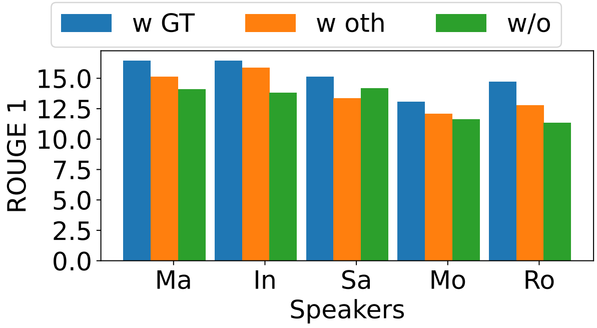


Figure 6.7: R1 scores for the responses generated by the most frequent five speakers in the dataset when the GT personality, other personalities sans GT, and no personalities are used for response generation.

Quantitative Results

Table 6.14 presents the scores achieved across the evaluation metrics for the MaSaC dataset. Apparently, the inclusion of personality information elevates the performance of our comparative systems across all metrics. Notably, BART outperforms the competition, whether with or without personality information, across majority of the metrics. We observe increased ROUGE-1 scores for all models, typically ranging from +13% to +21%. BLEU-1 also increases simultaneously from +12% to +38%. The consistent improvement in BERTScore (+3% to +5%) also underscores that the fusion of personality information into the dialogue context results in responses marked by enhanced coherence.

[5] `https://en.wikipedia.org/wiki/Sarabhai_vs_Sarabhai`
[6] `https://github.com/OpenNMT/OpenNMT-py`
[7] `https://huggingface.co/facebook/mbart-large-50-many-to-many-mmt`

Dialogue Context	**Monisha:** Mummy ji, use apni pehli anniversary yaad hai, ye sunkar aap chaunki kyun? *(Mother-in-law, why did you seem surprised when he remembered his first anniversary?)* **Maya:** Kyunki mardon ko, cricket match kab hai, WWF kab dikhane wale hain, inn do baaton ko chhod bahut jyada yaad ni rehta. And that's fine. *(Because men tend to remember when the next cricket match or WWE show is rather than things like anniversaries. And that's fine.)* **Monisha:** Kya fine? Mujhe to har important din yaad rehta hai. *(What do you mean, fine? I remember every important day.)* **Maya:** Haan, kyunki mard aur aurat alag-alag hain. Unke dimag alag tarah se kaam krte hain. *(Yes, because men and women are different. Their minds work in different ways.)*
GT Response	**Monisha:** Mai nhi manti. pichhle saal raat ko barah baje daddy ji ne aapko anniversary wish kiya tha na? Yaad krk? *(I don't believe that. Didn't Daddy wish you a happy anniversary last year at midnight? Remember?)*
BART	**Monisha:** daddy ji ka dimag alag tarah se kaam krte hain. *(Daddy's mind work in different way.)*
BART$_{\text{PA3}}$ (OPN)	**Monisha:** Mai nhi manti. daddy ji ko apni pehli anniversary yaad hai. *(I don't believe that. Daddy remembers his first anniversary.)*

Table 6.15: Responses generated for a sample dialogue from the test set of MaSaC by different model architectures.

Effect of Personality

We monitor ROUGE scores for responses from the top five most frequent speakers, as shown in Figure 6.2a. Comparing the responses generated by the BART model with ground-truth (GT) personalities (as listed in Table 6.13), we also assess results without personality fusion. The findings, presented as ROUGE-1 scores in Figure 6.7, consistently demonstrate improved performance after personality fusion. Notably, except for Sahil, every speaker exhibits enhanced performance when infused with the GT personality within the dialogue context.

Ablation Study

It is essential to recognize that integrating personality information into the dialogue context can be achieved through various techniques, each varying in complexity. In our study, we have delved into several fusion methodologies, encompassing straightforward concatenation, conventional dot-product attention, and personality-aware attention, both with and without the inclusion of Axial attention. We provide results for both BART and T5 since they exhibit comparable capabilities in Table 6.14. Evidently, the fusion of personality information contributes to better responses. Nevertheless, our findings emphasize that simple concatenation falls short in efficiency, yielding only marginal performance gains. In contrast, introducing attention mechanisms elevates performance, with our proposed approach of personality-aware fusion, coupled with Axial attention, being the most effective strategy. Additionally, we investigate the potential impact of fusing solely the identified personality trait without the intermediary step of mapping it into a trait definition. Our observations underscore the advantages of incorporating the complete trait definition rather than merely the isolated trait string within the response generation pipeline.

Qualitative Analysis

We select a sample dialogue from the test set and present the predicted responses generated by the conventional BART model alongside those generated after the integration of personality factors using PA3. These responses are compared with the ground-truth responses, comprehensively detailed in Table 6.15.

We observe that utilising personality information (OPN for the speaker in this case) aligns the response closer to the ground truth when compared with the standard BART model.

Human Evaluation

For generative tasks such as response generation, simple reliance on quantitative results proves insufficient, primarily due to the tendency of such metrics, like ROUGE and BLEU scores, to prioritize syntactic similarity over semantic equivalence. Therefore, we perform human evaluation. We conduct a comparative analysis of predictions derived from BART and T5 with and without the incorporation of personality information using PA3. We engage 25 human evaluators[8] who are tasked with assessing a randomly selected set of 50 responses generated by these methods. They assign each response a rating within the range of 1 to 5, considering common human evaluation metrics, including fluency, relevance, coherence, and personality orientation. To monitor the validity of the human evaluations, we calculate Cohen's Kappa (297) to quantify the inter-annotator agreement between the annotators. The average Kappa score for fluency, coherence, relevancy and personality oriented came out to be 0.83, 0.79, 0.68, and 0.71, respectively. The consolidated results of our human evaluation, shown in Table 6.16, reflect the averaged ratings across all obtained responses. Evidently, BART, when equipped with personality information using PA3, emerges as the top performer across all metrics.

Model	Fluency	Coherence	Relevancy	Personality oriented
T5	2.13	2.07	1.64	2.01
BART	2.17	2.03	1.79	2.04
T5_{PA3}	3.07	2.84	2.26	3.11
BART_{PA3}	**3.14**	**3.09**	**2.98**	**3.23**

Table 6.16: Results of human evaluation for the response generation task.

6.5.4 Evaluating LLMs

Similar to the rest of the chapters, we compare the performance of our best performing systems with Llama, a large language model. Table 6.17 highlights the results obtained for the SPICE data for the tasks of persona discovery, persona-type identification, and persona value extraction. It can be observed that, although Llama performs better than SPOT in almost all tasks, it performs comparably to SPOT. This minor increase in performance signifies the importance of data availability over parameter increase. For the task of personality-assisted response generation in the code-mixed setting, we show the results in Table 6.18. Here also, we see a minor increase in the response generation performance when Llama is used in place of our top performing system. PA3.

	Persona Discovery			**Persona-Type Identification**	**Persona Value Extraction**			
	Prec	**Rec**	**F1**	**Weighted F1 Score**	**R1**	**R2**	**B1**	**B2**
SPOT	**0.47**	0.58	0.52	0.56	29.51	2.97	27.16	2.27
Llama	**0.47**	**0.60**	**0.53**	**0.58**	**31.02**	**3.54**	**28.90**	**4.01**

Table 6.17: Performance of Llama when compared with our proposed methodology for the three tasks of speaker profiling.

[8] The evaluators are linguists fluent in English and Hindi with a good grasp of personalized dialogues, aged between 25-30.

	ROUGE 1	**ROUGE 2**	**ROUGE L**	**BLEU 1**	**BLEU 2**	**BLEU 3**	**BLEU 4**	**BERTScore**
PA3	14.67	2.77	14.11	10.92	2.51	0.73	0.27	65.93
Llama	**15.71**	**3.06**	**14.98**	**11.46**	**3.74**	**1.08**	**1.01**	**67.64**

Table 6.18: Performance of Llama when compared with our proposed methodology for personality-assisted response generation.

6.6 Conclusion

In conclusion, this chapter addressed the critical issue of personalization in dialogue agents, recognising the inadequacy of *one-size-fits-all* approaches in the dynamic landscape of conversational interactions. The exploration of the Speaker Profiling in Conversations task, undertaken through two distinct approaches for monolingual English and Hindi-English code-mixed conversations, contributed significantly to the advancement of personalised dialogue generation. For monolingual English, our focus on persona discovery, persona-type classification, and persona-value extraction within dialogues led to the development of a novel dataset, SPICE, annotated with explicit profiling labels. The meticulous evaluation of diverse baseline models against this dataset, along with the introduction of the SPOT, provided a comprehensive understanding of the advantages and limitations exhibited by different constituent modules. In the realm of Hindi-English code-mixed conversations, our proposed unsupervised method for speaker profile extraction introduced a unique approach leveraging the Big Five personality traits. The fusion of these inferred personality attributes with the dialogue context, facilitated by the PA3, demonstrated substantial improvements in response generation. The observed enhancements in ROUGE and BLUE scores underscored the effectiveness of integrating personality information seamlessly into the dialogue context, offering a more contextually relevant and personalised dialogue generation experience.

7. Conclusion and Future Work

With this chapter, we now summarize the contributions made by this thesis and a discussion of possible directions in which future research could be extended.

This thesis centrally explored the integration of affective cues into chatbots, with the aim of endowing them with an acute awareness of the nuanced dimensions of human conversation. Our investigation delved into four pivotal affective elements: emotion, humour, sarcasm, and speaker profile. In addition to comprehending these attributes, we embarked on explaining their underlying mechanisms. The impetus for delving into this research domain stemmed from a dedication to advancing the ambitious objective of Artificial General Intelligence. The underlying premise was that an exemplary conversational agent should not merely engage in dialogue but should also adeptly navigate the intricate landscape of emotions, humour, sarcasm, and other affective facets. The aspiration was to foster an affective understanding in conversational agents so that ultimately they become capable of emulating human-like conversational prowess. Below we provide a conclusion for each of the explored affect in the thesis.

Emotions is the primary affect that influence a conversation. Consequently, emotion analysis becomes of paramount importance in the conversational AI domain. To this end, we handled the task of Emotion Recognition in Conversation and Emotion Flip Reasoning in monolingual English and Hindi-English code-mixed conversations. While the former task deals with recognising the emotions of a dialogue, the latter focuses on giving an explanation behind any emotion shift present in a dialogue. We proposed multiple models for the different settings of the ERC and EFR tasks based on the target language. We also curated datasets to perform the EFR and the code-mixed ERC task. We explored standalone as well as external knowledge infused mechanisms and found them to perform well when compared with comparative systems.

Apart from emotions, humour and sarcasm also directly effect the speaker dynamics in a conversation. To this end, we explore the analysis of humour and sarcasm to obtain the underlying meaning of a statement in a dialogue. We focus on identifying utterances in a dialogue that display the presence of amusement and/or irony in them. Further, we also proposed a new task – Sarcasm Explanation in Dialogues, which aimed at generating natural language explanations for the sarcasm present in a dialogue. We posit that sarcasm needs an external explanation for the dialogue agent as otherwise, it becomes difficult to capture the true connotation behind an uttered remark. We explore the classification and explanation tasks for monolingual English and Hindi-English code-mixed conversations and consequently curated datasets for all these tasks. We experimented with multimodality using Transformer based systems assisted with novel attention mechanisms. Thorough analysis shows considerable improvement and importance of our results.

Once we obtain a clear understanding of emotions, humour, and sarcasm, we focus on another aspect of human behaviour, that is, speaker profile. An individual experiences life in a distinct manner which results in a unique set of characteristics and demeanour, which in turn affect the style of their responses in a dialogue. In other words, different speakers respond differently in the exact same situation based on their inherent perception and bias. Consequently, we explore the task of speaker profiling in monolingual English and Hindi-English code-mixed to classify or generate a speaker's profile so that the profile can be used to generate appropriate responses. We curated a dataset for generating speaker profile for English dialogues. Whereas, we present the task of speaker profiling as an unsupervised task in the code-mixed setting. Experimentation with Transformer based architectures and attention based fusion mechanisms, we establish the superiority of our proposed systems over baseline methods.

Throughout this thesis, deep learning methods such as Transformers and RNNs are employed extensively.

While these techniques excel in handling large datasets and complex conversational scenarios, obtaining sufficient computational resources for larger datasets may pose challenges. Longer training times and extended hardware utilization are typically necessary for such models. Therefore, achieving an optimal balance between compute resource allocation and model performance is crucial for scalability considerations. Regarding generalizability, our methods have been rigorously tested across a diverse array of datasets, encompassing multilingual and multimodal dialogues. This extensive evaluation highlights the robustness and transferability of our systems across various data domains. However, when addressing affective traits, it is imperative to navigate a myriad of ethical considerations. Prior to real-world deployment, comprehensive studies must be conducted to ensure that these models uphold principles of inclusivity, fairness, and sensitivity toward diverse user demographics.

In summary, we have explored emotion analysis, humour and sarcasm analysis, and speaker profiling for monolingual English and Hindi-English code-mixed conversations. With this understanding, we have set the stage for further research in the direction of affect understanding, explanation, and generation for multilingual dialogues. Below, we highlight a few possible research direction for the future.

Intensity aware emotion analysis: In this thesis, we have exclusively focused on Ekman's emotion labels. However, many studies have also focused on other types of emotion sets, such as Piccard's emotions. Future studies, that aims to perform ERC and EFR tasks for dialogue systems can consider these other emotion sets, some of which also contain an emotional intensity indicator with them. Emotional intensity indicates the extent to which a particular emotion is being experienced. For example, being denied your favourite food can make you sad, but a demise of a pet will make you more sad, thus increasing the intensity of the sadness emotion. Consequently, monitoring the change in emotional intensity and reasoning it can also be a possible direction for future research.

Multimodal triggers for emotion flips: We use the textual modality only to monitor the emotional dynamics of speakers in a dialogue in this thesis. We focus on the textual utterances as triggers in our setting. However, there can be cases where a person's emotion shift from one to another due to some other modality. For instance, suppose a person sees an overflowing trashcan and their emotion shifts from joy to disgust, in this case the trigger modality is visual and it becomes impossible to identify the appropriate emotion flip trigger from text. Thus, multimodal trigger identification for emotion flips can be an interesting research avenue.

Cognitive analysis of affects: The ability to gauge affects lies in the cognitive behaviour of the listener. From a computational standpoint, we have explored affect comprehension using deep neural networks, which tries to mimic the cognitive behaviour of the interlocutors. However, the correlation between the biological neural network and the computational neural network for understanding the various affects is not yet fully understood. Moreover, the use of cognitive and physiological signals such as EEG and gaze, which is implicitly a part of biological neural networks, is not yet entirely explored from a computational viewpoint. To this end, an interesting future direction could be to utilise an individual's cognitive signals to explain various affects in a conversation.

Affect aware response generation: Once we obtain the values and explanation for distinct affects of an individual, the direct implication of them would be to generate apt responses. This thesis focuses on the understanding aspect and introduces various new datasets and methods to give a head-start for the research in this area. One of the future work can be to use these affects for generating responses which are more human in nature.

This thesis has illuminated novel perspectives for research in affects for English and code-mixed dialogues, shedding light on previously unexplored facets and contributing valuable insights to the existing body of knowledge. We also underscore the ongoing need for continued exploration, encouraging future researchers to build upon this foundation and explore new avenues for further innovation and discovery.

Links to codes and datasets

- https://github.com/LCS2-IIITD/Emotion-Flip-Reasoning.git
- https://github.com/LCS2-IIITD/EMNLP-COFFEE.git
- https://github.com/LCS2-IIITD/EFR-Instigators.git
- https://github.com/LCS2-IIITD/MSH-COMICS.git
- https://github.com/LCS2-IIITD/MAF.git
- https://github.com/LCS2-IIITD/MOSES.git
- https://github.com/LCS2-IIITD/SPOT.git
- https://github.com/LCS2-IIITD/PA3.git

REFERENCES

[1] C. Busso, M. Bulut, C.-C. Lee, A. Kazemzadeh, E. Mower, S. Kim, J. N. Chang, S. Lee, and S. S. Narayanan, “Iemocap: Interactive emotional dyadic motion capture database,” *Language resources and evaluation*, vol. 42, no. 4, pp. 335–359, 2008.

[2] S. Castro, D. Hazarika, V. Pérez-Rosas, R. Zimmermann, R. Mihalcea, and S. Poria, “Towards multimodal sarcasm detection (an _Obviously_ perfect paper),” in *ACL*, 2019, pp. 4619–4629.

[3] A. Ghosh and T. Veale, “Magnets for sarcasm: Making sarcasm detection timely, contextual and very personal,” in *EMNLP*, 2017, pp. 482–491.

[4] N. Majumder, S. Poria, D. Hazarika, R. Mihalcea, A. F. Gelbukh, and E. Cambria, “DialogueRNN: An Attentive RNN for Emotion Detection in Conversations,” in *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI*, Honolulu, Hawaii, USA, 2019, pp. 6818–6825.

[5] M. Dingemanse and S. Floyd, “Conversation across cultures,” in *The Cambridge handbook of linguistic anthropology*. Cambridge University Press, 2014, pp. 447–480.

[6] G. Kasper and J. Wagner, “Conversation analysis in applied linguistics,” *Annual Review of Applied Linguistics*, vol. 34, pp. 171–212, 2014.

[7] W. Turnbull, *Language in action: Psychological models of conversation*. Psychology Press, 2003.

[8] D. Traum, “A computational theory of grounding in natural language conversation,” 1994.

[9] E. Horvitz and T. Paek, “A computational architecture for conversation,” in *UM99 User Modeling: Proceedings of the Seventh International Conference*. Springer, 1999, pp. 201–210.

[10] J. Dalton, S. Fischer, P. Owoicho, F. Radlinski, F. Rossetto, J. R. Trippas, and H. Zamani, “Conversational information seeking: Theory and application,” ser. SIGIR ’22, 2022, p. 3455–3458.

[11] J. Lester, K. Branting, and B. Mott, “Conversational agents,” *The practical handbook of internet computing*, pp. 220–240, 2004.

[12] J. Welser, J. W. Pitera, and C. Goldberg, “Future computing hardware for ai,” in *2018 IEEE International Electron Devices Meeting (IEDM)*, 2018, pp. 1.3.1–1.3.6.

[13] A. Gottardi, O. Ipek, G. Castellucci, S. Hu, L. Vaz, Y. Lu, A. Khatri, A. Chadha, D. Zhang, S. Sahai, P. Dwivedi, H. Shi, L. Hu, A. Huang, L. Dai, B. Yang, V. Somani, P. Rajan, R. Rezac, M. Johnston, S. Stiff, L. Ball, D. Carmel, Y. Liu, D. Hakkani-Tur, O. Rokhlenko, K. Bland, E. Agichtein, R. Ghanadan, and Y. Maarek, “Alexa, let’s work together: Introducing the first alexa prize taskbot challenge on conversational task assistance,” 2022.

[14] J. Ruusuvuori, “Emotion, affect and conversation,” *The handbook of conversation analysis*, pp. 330–349, 2012.

[15] L. Clark, N. Pantidi, O. Cooney, P. Doyle, D. Garaialde, J. Edwards, B. Spillane, E. Gilmartin, C. Murad, C. Munteanu, V. Wade, and B. R. Cowan, “What makes a good conversation? challenges in designing truly conversational agents,” in *Proceedings of the 2019 CHI Conference on Human*

Factors in Computing Systems, ser. CHI '19. New York, NY, USA: Association for Computing Machinery, 2019, p. 1–12. [Online]. Available: https://doi.org/10.1145/3290605.3300705

[16] D. Menaker, *A good talk: The story and skill of conversation.* Hachette UK, 2010.

[17] M. W. Tay, "Code switching and code mixing as a communicative strategy in multilingual discourse," *World Englishes*, vol. 8, no. 3, pp. 407–417, 1989.

[18] N. Tarihoran and I. R. Sumirat, "The impact of social media on the use of code mixing by generation z," *International Journal of Interactive Mobile Technologies (iJIM)*, vol. 16, no. 7, pp. 54–69, 2022.

[19] H. Dahl and V. Teller, "What ai needs is a theory of the functions of emotions," AAAI Technical Report FS-98-O3, Tech. Rep., 1998.

[20] D. Hazarika, S. Poria, A. Zadeh, E. Cambria, L.-P. Morency, and R. Zimmermann, "Conversational memory network for emotion recognition in dyadic dialogue videos," in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, New Orleans, Louisiana, 2018, pp. 2122–2132.

[21] S. Poria, N. Majumder, D. Hazarika, D. Ghosal, R. Bhardwaj, S. Y. B. Jian, P. Hong, R. Ghosh, A. Roy, N. Chhaya *et al.*, "Recognizing emotion cause in conversations," *Cognitive Computation*, vol. 13, pp. 1317–1332, 2021.

[22] M. Dynel, "Beyond a joke: Types of conversational humour," *Language and Linguistics Compass*, vol. 3, no. 5, pp. 1284–1299, 2009. [Online]. Available: https://compass.onlinelibrary.wiley.com/doi/abs/10.1111/j.1749-818X.2009.00152.x

[23] A. Joshi, P. Bhattacharyya, and M. J. Carman, "Automatic sarcasm detection: A survey," *ACM Comput. Surv.*, vol. 50, no. 5, sep 2017. [Online]. Available: https://doi.org/10.1145/3124420

[24] M. P. Mulder and A. Nijholt, *Humour research: State of the art.* Centre for Telematics and Information Technology, University of Twente, 2002.

[25] D. Ghosh, A. R. Fabbri, and S. Muresan, "Sarcasm analysis using conversation context," *Computational Linguistics*, vol. 44, no. 4, pp. 755–792, 2018.

[26] G. Chen, Y. Zheng, and Y. Du, "Listener's social identity matters in personalised response generation," in *Proceedings of the 13th International Conference on Natural Language Generation*. Dublin, Ireland: Association for Computational Linguistics, Dec. 2020, pp. 205–215. [Online]. Available: https://aclanthology.org/2020.inlg-1.26

[27] S. Zhang, E. Dinan, J. Urbanek, A. Szlam, D. Kiela, and J. Weston, "Personalizing dialogue agents: I have a dog, do you have pets too?" in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Melbourne, Australia: Association for Computational Linguistics, Jul. 2018, pp. 2204–2213. [Online]. Available: https://aclanthology.org/P18-1205

[28] X. Yang, M. Aurisicchio, and W. Baxter, "Understanding affective experiences with conversational agents," in *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, ser. CHI '19. New York, NY, USA: Association for Computing Machinery, 2019, p. 1–12. [Online]. Available: https://doi.org/10.1145/3290605.3300772

[29] P. Ekman, "An argument for basic emotions," *Cognition & emotion*, vol. 6, no. 3-4, pp. 169–200, 1992.

[30] R. W. Picard, *Affective Computing*. Cambridge, MA, USA: MIT Press, 1997.

[31] L. Dini and A. Bittar, "Emotion analysis on twitter: the hidden challenge," in *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, 2016, pp. 3953–3958.

[32] T. Tomihira, A. Otsuka, A. Yamashita, and T. Satoh, "What does your tweet emotion mean? neural emoji prediction for sentiment analysis," in *proceedings of the 20th international conference on information integration and web-based applications & services*, 2018, pp. 289–296.

[33] H. Li, H. Liu, and Z. Zhang, "Online persuasion of review emotional intensity: A text mining analysis of restaurant reviews," *International Journal of Hospitality Management*, vol. 89, p. 102558, 2020.

[34] A. Zablocki, K. Makri, and M. J. Houston, "Emotions within online reviews and their influence on product attitudes in austria, usa and thailand," *Journal of interactive marketing*, vol. 46, pp. 20–39, 2019.

[35] Y. Rao, J. Lei, L. Wenyin, Q. Li, and M. Chen, "Building emotional dictionary for sentiment analysis of online news," *World Wide Web*, vol. 17, pp. 723–742, 2014.

[36] B. Ghanem, P. Rosso, and F. Rangel, "An emotional analysis of false information in social media and news articles," *ACM Transactions on Internet Technology (TOIT)*, vol. 20, no. 2, pp. 1–18, 2020.

[37] R. E. Banchs, "On the construction of more human-like chatbots: Affect and emotion analysis of movie dialogue data," in *2017 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*. IEEE, 2017, pp. 1364–1367.

[38] K.-J. Oh, D. Lee, B. Ko, and H.-J. Choi, "A chatbot for psychiatric counseling in mental healthcare service based on emotional dialogue analysis and sentence generation," in *2017 18th IEEE international conference on mobile data management (MDM)*. IEEE, 2017, pp. 371–375.

[39] A. S. Gabriel, A. Cheshin, C. M. Moran, and G. A. Van Kleef, "Enhancing emotional performance and customer service through human resources practices: A systems perspective," *Human Resource Management Review*, vol. 26, no. 1, pp. 14–24, 2016.

[40] S. Fitrianie, P. Wiggers, and L. J. Rothkrantz, "A multi-modal eliza using natural language processing and emotion recognition," in *Text, Speech and Dialogue: 6th International Conference, TSD 2003, České Budějovice, Czech Republic, September 8-12, 2003. Proceedings 6*. Springer, 2003, pp. 394–399.

[41] Z.-J. Chuang and C.-H. Wu, "Multi-modal emotion recognition from speech and text," in *International Journal of Computational Linguistics & Chinese Language Processing, Volume 9, Number 2, August 2004: Special Issue on New Trends of Speech and Language Processing*, 2004, pp. 45–62.

[42] H. Li, N. Pang, S. Guo, and H. Wang, "Research on textual emotion recognition incorporating personality factor," in *2007 IEEE International Conference on Robotics and Biomimetics (ROBIO)*. IEEE, 2007, pp. 2222–2227.

[43] D. Hazarika, S. Poria, R. Mihalcea, E. Cambria, and R. Zimmermann, "Icon: interactive conversational memory network for multimodal emotion detection," in *EMNLP*, 2018, pp. 2594–2604.

[44] P. Zhong, D. Wang, and C. Miao, "Knowledge-enriched transformer for emotion detection in textual conversations," in *EMNLP-IJCNLP*. Hong Kong, China: Association for Computational Linguistics, Nov. 2019, pp. 165–176.

[45] W. Li, W. Shao, S. Ji, and E. Cambria, "Bieru: bidirectional emotional recurrent unit for conversational sentiment analysis," *arXiv preprint arXiv:2006.00492*, 2020.

[46] D. Ghosal, N. Majumder, S. Poria, N. Chhaya, and A. Gelbukh, "Dialoguegcn: A graph convolutional neural network for emotion recognition in conversation," *arXiv preprint arXiv:1908.11540*, 2019.

[47] W. Jiao, M. Lyu, and I. King, "Real-time emotion recognition via attention gated hierarchical memory network," in *AAAI*, vol. 34, no. 05, 2020, pp. 8002–8009.

[48] W. Jiao, H. Yang, I. King, and M. R. Lyu, "Higru: Hierarchical gated recurrent units for utterance-level emotion recognition," *arXiv preprint arXiv:1904.04446*, 2019.

[49] D. Hazarika, S. Poria, R. Zimmermann, and R. Mihalcea, "Conversational transfer learning for emotion recognition," *Information Fusion*, vol. 65, pp. 1–12, 2021.

[50] W. Shen, J. Chen, X. Quan, and Z. Xie, "Dialogxl: All-in-one xlnet for multi-party conversation emotion recognition," *arXiv preprint arXiv:2012.08695*, 2020.

[51] L. Yang, Y. Shen, Y. Mao, and L. Cai, "Hybrid curriculum learning for emotion recognition in conversation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, no. 10, 2022, pp. 11 595–11 603.

[52] H. Ma, J. Wang, H. Lin, X. Pan, Y. Zhang, and Z. Yang, "A multi-view network for real-time emotion recognition in conversations," *Knowledge-Based Systems*, vol. 236, p. 107751, 2022.

[53] R. Xia and Z. Ding, "Emotion-cause pair extraction: A new task to emotion analysis in texts," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence, Italy: Association for Computational Linguistics, Jul. 2019, pp. 1003–1012. [Online]. Available: https://aclanthology.org/P19-1096

[54] R. Mihalcea and C. Strapparava, "Making computers laugh: Investigations in automatic humor recognition," in *EMNLP*, 2005, pp. 531–538.

[55] Z. Yang, B. Hu, and J. Hirschberg, "Predicting Humor by Learning from Time-Aligned Comments," in *Interspeech*, 2019, pp. 496–500.

[56] I. Annamoradnejad, "Colbert: Using bert sentence embedding for humor detection," *ArXiv*, vol. abs/2004.12765, 2020.

[57] M. K. Hasan, W. Rahman, A. Bagher Zadeh, J. Zhong, M. I. Tanveer, L.-P. Morency, and M. E. Hoque, "UR-FUNNY: A multimodal language dataset for understanding humor," in *EMNLP-IJCNLP*, 2019, pp. 2046–2056.

[58] R. Kreuz and G. Caucci, "Lexical influences on the perception of sarcasm," in *Proceedings of the Workshop on Computational Approaches to Figurative Language*, 2007, pp. 1–4.

[59] O. Tsur, D. Davidov, and A. Rappoport, "ICWSM - A Great Catchy Name: Semi-Supervised Recognition of Sarcastic Sentences in Online Product Reviews," in *ICWSM*, W. W. Cohen and S. Gosling, Eds., 2010.

[60] Y. Tay, A. T. Luu, S. C. Hui, and J. Su, “Reasoning with sarcasm by reading in-between,” in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Melbourne, Australia: Association for Computational Linguistics, Jul. 2018, pp. 1010–1020. [Online]. Available: https://aclanthology.org/P18-1093

[61] T. Xiong, P. Zhang, H. Zhu, and Y. Yang, “Sarcasm detection with self-matching networks and low-rank bilinear pooling,” in *The World Wide Web Conference*, ser. WWW ’19. New York, NY, USA: Association for Computing Machinery, 2019, p. 2115–2124. [Online]. Available: https://doi.org/10.1145/3308558.3313735

[62] H. Srivastava, V. Varshney, S. Kumari, and S. Srivastava, “A novel hierarchical BERT architecture for sarcasm detection,” in *Proceedings of the Second Workshop on Figurative Language Processing*. Online: Association for Computational Linguistics, Jul. 2020, pp. 93–97. [Online]. Available: https://aclanthology.org/2020.figlang-1.14

[63] J. Weston, E. Dinan, and A. H. Miller, “Retrieve and refine: Improved sequence generation models for dialogue,” *arXiv preprint arXiv:1808.04776*, 2018.

[64] S. Roller, E. Dinan, N. Goyal, D. Ju, M. Williamson, Y. Liu, J. Xu, M. Ott, K. Shuster, E. M. Smith *et al.*, “Recipes for building an open-domain chatbot,” *arXiv preprint arXiv:2004.13637*, 2020.

[65] E. Dinan, S. Roller, K. Shuster, A. Fan, M. Auli, and J. Weston, “Wizard of wikipedia: Knowledge-powered conversational agents,” in *International Conference on Learning Representations*, 2019. [Online]. Available: https://openreview.net/forum?id=r1l73iRqKm

[66] G. Chen, Y. Zheng, and Y. Du, “Listener’s social identity matters in personalised response generation,” *arXiv preprint arXiv:2010.14342*, 2020.

[67] J. Lucas, F. Fernández, J. Salazar, J. Ferreiros, and R. San Segundo, “Managing speaker identity and user profiles in a spoken dialogue system,” *Procesamiento del lenguaje natural*, no. 43, pp. 77–84, 2009.

[68] C. K. Joshi, F. Mi, and B. Faltings, “Personalization in goal-oriented dialog,” *arXiv preprint arXiv:1706.07503*, 2017.

[69] J. Li, M. Galley, C. Brockett, G. P. Spithourakis, J. Gao, and B. Dolan, “A persona-based neural conversation model,” *arXiv preprint arXiv:1603.06155*, 2016.

[70] S. Zhang, E. Dinan, J. Urbanek, A. Szlam, D. Kiela, and J. Weston, “Personalizing dialogue agents: I have a dog, do you have pets too?” *arXiv preprint arXiv:1801.07243*, 2018.

[71] S. Poria, D. Hazarika, N. Majumder, G. Naik, E. Cambria, and R. Mihalcea, “Meld: A multimodal multi-party dataset for emotion recognition in conversations,” *arXiv preprint arXiv:1810.02508*, 2018.

[72] P. Ekman, “An argument for basic emotions,” *Cognition and Emotion*, pp. 169–200, 1992.

[73] M. Abdul-Mageed and L. Ungar, “EmoNet: Fine-grained emotion detection with gated recurrent neural networks,” in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Vancouver, Canada, 2017, pp. 718–728.

[74] A. Chatterjee, K. N. Narahari, M. Joshi, and P. Agrawal, “SemEval-2019 task 3: EmoContext contextual emotion detection in text,” in *Proceedings of the 13th International Workshop on Semantic Evaluation*, Minneapolis, Minnesota, USA, 2019, pp. 39–48.

[75] M. S. Akhtar, D. Ghosal, A. Ekbal, P. Bhattacharyya, and S. Kurohashi, "All-in-one: Emotion, sentiment and intensity prediction using a multi-task ensemble framework," *IEEE Transactions on Affective Computing*, pp. 1–1, 2019.

[76] N. Gupta, M. Gilbert, and G. Di Fabbrizio, "Emotion detection in email customer care," in *Proceedings of the NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*, Los Angeles, CA, 2010, pp. 10–16.

[77] L. Dini and A. Bittar, "Emotion analysis on twitter: The hidden challenge," in *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, Portorož, Slovenia, 2016, pp. 3953–3958.

[78] M. S. Akhtar, A. Ekbal, and E. Cambria, "How intense are you? predicting intensities of emotions and sentiments using stacked ensemble [application notes]," *IEEE Computational Intelligence Magazine*, vol. 15, no. 1, pp. 64–75, 2020.

[79] H. Khanpour and C. Caragea, "Fine-grained emotion detection in health-related online posts," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, Brussels, Belgium, 2018, pp. 1160–1166.

[80] G. Tu, J. Wen, C. Liu, D. Jiang, and E. Cambria, "Context- and sentiment-aware networks for emotion recognition in conversation," *IEEE Transactions on Artificial Intelligence*, vol. 3, no. 5, pp. 699–708, 2022.

[81] A. Hakulinen, "Conversation types," *The pragmatics of interaction*, pp. 55–65, 2009.

[82] L. Caires and H. T. Vieira, "Conversation types," *Theoretical Computer Science*, vol. 411, no. 51, pp. 4399–4440, 2010, european Symposium on Programming 2009. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0304397510004895

[83] E. Weigand, "Language as dialogue," vol. 7, no. 3, pp. 505–515, 2010. [Online]. Available: https://doi.org/10.1515/iprg.2010.022

[84] G. Kasper and J. Wagner, "Conversation analysis in applied linguistics," *Annual Review of Applied Linguistics*, vol. 34, p. 171–212, 2014.

[85] S. Poria, D. Hazarika, N. Majumder, G. Naik, E. Cambria, and R. Mihalcea, "MELD: A multimodal multi-party dataset for emotion recognition in conversations," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence, Italy: Association for Computational Linguistics, Jul. 2019, pp. 527–536. [Online]. Available: https://aclanthology.org/P19-1050

[86] S. Sukhbaatar, A. Sszlam, J. Weston, and R. Fergus, "End-to-end memory networks," in *Proceedings of the Advances in Neural Information Processing Systems*, 2015, pp. 2440–2448.

[87] D. Ghosal, N. Majumder, A. Gelbukh, R. Mihalcea, and S. Poria, "Cosmic: Commonsense knowledge for emotion identification in conversations," *arXiv preprint arXiv:2010.02795*, 2020.

[88] R. Speer, J. Chin, and C. Havasi, "Conceptnet 5.5: An open multilingual graph of general knowledge," in *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, ser. AAAI'17. AAAI Press, 2017, p. 4444–4451.

[89] A. Bosselut, H. Rashkin, M. Sap, C. Malaviya, A. Celikyilmaz, and Y. Choi, "COMET: Commonsense transformers for automatic knowledge graph construction," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence, Italy: Association for Computational Linguistics, Jul. 2019, pp. 4762–4779. [Online]. Available: https://aclanthology.org/P19-1470

[90] P. Zhong, D. Wang, P. Li, C. Zhang, H. Wang, and C. Miao, "Care: Commonsense-aware emotional response generation with latent concepts," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 16, pp. 14 577–14 585, May 2021.

[91] D. Ghosal, N. Majumder, A. Gelbukh, R. Mihalcea, and S. Poria, "COSMIC: COmmonSense knowledge for eMotion identification in conversations," in *Findings of the Association for Computational Linguistics: EMNLP 2020*. Online: Association for Computational Linguistics, Nov. 2020, pp. 2470–2481. [Online]. Available: https://aclanthology.org/2020.findings-emnlp.224

[92] S. Poria, E. Cambria, D. Hazarika, N. Majumder, A. Zadeh, and L.-P. Morency, "Context-dependent sentiment analysis in user-generated videos," in *ACL*, 2017, pp. 873–883.

[93] A. Balahur, J. M. Hermida, A. Montoyo, and R. Munoz, "Emotinet: A knowledge base for emotion detection in text built on the appraisal theories," in *Natural Language Processing and Information Systems: 16th International Conference on Applications of Natural Language to Information Systems, NLDB 2011, Alicante, Spain, June 28-30, 2011. Proceedings 16*. Springer, 2011, pp. 27–39.

[94] J. Chen, T. Yang, Z. Huang, K. Wang, M. Liu, and C. Lyu, "Incorporating structured emotion commonsense knowledge and interpersonal relation into context-aware emotion recognition," *Applied Intelligence*, vol. 53, no. 4, p. 4201–4217, jun 2022. [Online]. Available: https://doi.org/10.1007/s10489-022-03729-4

[95] M. Sap, R. Le Bras, E. Allaway, C. Bhagavatula, N. Lourie, H. Rashkin, B. Roof, N. A. Smith, and Y. Choi, "Atomic: An atlas of machine commonsense for if-then reasoning," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 01, pp. 3027–3035, Jul. 2019.

[96] W. Nie, Y. Bao, Y. Zhao, and A. Liu, "Long dialogue emotion detection based on commonsense knowledge graph guidance," *IEEE Transactions on Multimedia*, pp. 1–15, 2023.

[97] D. Li, X. Zhu, Y. Li, S. Wang, D. Li, J. Liao, and J. Zheng, "Enhancing emotion inference in conversations with commonsense knowledge," *Knowledge-Based Systems*, vol. 232, p. 107449, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0950705121007115

[98] T. T. Sasidhar, P. B, and S. K. P, "Emotion detection in hinglish(hindi+english) code-mixed social media text," *Procedia Computer Science*, vol. 171, pp. 1346–1352, 2020, third International Conference on Computing and Network Communications (CoCoNet'19). [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1877050920311236

[99] A. Ilyas, K. Shahzad, and M. Kamran Malik, "Emotion detection in code-mixed roman urdu - english text," *ACM Trans. Asian Low-Resour. Lang. Inf. Process.*, vol. 22, no. 2, mar 2023. [Online]. Available: https://doi.org/10.1145/3552515

[100] A. Wadhawan and A. Aggarwal, "Towards emotion recognition in hindi-english code-mixed data: A transformer based approach," 2021.

[101] A. Suciati and I. Budi, “Aspect-based sentiment analysis and emotion detection for code-mixed review,” *International Journal of Advanced Computer Science and Applications*, vol. 11, no. 9, 2020. [Online]. Available: http://dx.doi.org/10.14569/IJACSA.2020.0110921

[102] X. Zhu, Y. Lou, H. Deng, and D. Ji, “Leveraging bilingual-view parallel translation for code-switched emotion detection with adversarial dual-channel encoder,” *Knowledge-Based Systems*, vol. 235, p. 107436, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0950705121006985

[103] S. Kumar, A. Kulkarni, M. S. Akhtar, and T. Chakraborty, “When did you become so smart, oh wise one?! sarcasm explanation in multi-modal multi-party dialogues,” in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Dublin, Ireland: Association for Computational Linguistics, May 2022, pp. 5956–5968. [Online]. Available: https://aclanthology.org/2022.acl-long.411

[104] S. Kumar, I. Mondal, M. S. Akhtar, and T. Chakraborty, “Explaining (sarcastic) utterances to enhance affect understanding in multimodal dialogues,” 2022.

[105] M. Bedi, S. Kumar, M. S. Akhtar, and T. Chakraborty, “Multi-modal sarcasm detection and humor classification in code-mixed conversations,” *IEEE Transactions on Affective Computing*, vol. 14, no. 2, pp. 1363–1375, 2023.

[106] H. Madhu, S. Satapara, S. Modha, T. Mandl, and P. Majumder, “Detecting offensive speech in conversational code-mixed dialogue on social media: A contextual dataset and benchmark experiments,” *Expert Systems with Applications*, vol. 215, p. 119342, 2023.

[107] C.-C. Hsu, S.-Y. Chen, C.-C. Kuo, T.-H. Huang, and L.-W. Ku, “EmotionLines: An emotion corpus of multi-party conversations,” in *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, N. Calzolari, K. Choukri, C. Cieri, T. Declerck, S. Goggi, K. Hasida, H. Isahara, B. Maegaard, J. Mariani, H. Mazo, A. Moreno, J. Odijk, S. Piperidis, and T. Tokunaga, Eds. Miyazaki, Japan: European Language Resources Association (ELRA), May 2018. [Online]. Available: https://aclanthology.org/L18-1252

[108] K. Krippendorff, “Computing krippendorff’s alpha-reliability,” 2011.

[109] D. Ma, S. Li, X. Zhang, and H. Wang, “Interactive attention networks for aspect-level sentiment classification,” in *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017*, Melbourne, Australia, 2017, pp. 4068–4074.

[110] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “Bert: Pre-training of deep bidirectional transformers for language understanding,” *arXiv preprint arXiv:1810.04805*, 2018.

[111] S. Poria, E. Cambria, R. Bajpai, and A. Hussain, “A review of affective computing: From unimodal analysis to multimodal fusion,” *Information fusion*, vol. 37, pp. 98–125, 2017.

[112] R. Bose, S. Pande, and B. Banerjee, “Two headed dragons: multimodal fusion and cross modal transactions,” in *2021 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2021, pp. 2893–2897.

[113] Y. Ma and B. Ma, “Multimodal sentiment analysis on unaligned sequences via holographic embedding,” in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022, pp. 8547–8551.

[114] B. Yang, J. Li, D. F. Wong, L. S. Chao, X. Wang, and Z. Tu, “Context-aware self-attention networks,” *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 01, pp. 387–394, Jul. 2019. [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/3809

[115] D. Hazarika, S. Poria, R. Mihalcea, E. Cambria, and R. Zimmermann, “ICON: Interactive conversational memory network for multimodal emotion detection,” in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, Brussels, Belgium, 2018, pp. 2594–2604.

[116] D. Ghosal, N. Majumder, S. Poria, N. Chhaya, and A. Gelbukh, “DialogueGCN: A graph convolutional neural network for emotion recognition in conversation,” in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Hong Kong, China: Association for Computational Linguistics, Nov. 2019, pp. 154–164. [Online]. Available: https://aclanthology.org/D19-1015

[117] W. Jiao, M. R. Lyu, and I. King, “Real-time emotion recognition via attention gated hierarchical memory network,” *arXiv preprint arXiv:1911.09075*, 2019.

[118] O. Vinyals, M. Fortunato, and N. Jaitly, “Pointer networks,” *arXiv preprint arXiv:1506.03134*, 2015.

[119] C. Busso, M. Bulut, C.-C. Lee, A. Kazemzadeh, E. Mower, S. Kim, J. N. Chang, S. Lee, and S. S. Narayanan, “Iemocap: Interactive emotional dyadic motion capture database,” *Language resources and evaluation*, vol. 42, no. 4, pp. 335–359, 2008.

[120] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, “Roberta: A robustly optimized bert pretraining approach,” *arXiv preprint arXiv:1907.11692*, 2019.

[121] S. Khanuja, D. Bansal, S. Mehtani, S. Khosla, A. Dey, B. Gopalan, D. K. Margam, P. Aggarwal, R. T. Nagipogu, S. Dave, S. Gupta, S. C. B. Gali, V. Subramanian, and P. P. Talukdar, “Muril: Multilingual representations for indian languages,” *CoRR*, vol. abs/2103.10730, 2021. [Online]. Available: https://arxiv.org/abs/2103.10730

[122] J. Lee and W. Lee, “CoMPM: Context modeling with speaker’s pre-trained memory tracking for emotion recognition in conversation,” in *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Seattle, United States: Association for Computational Linguistics, Jul. 2022, pp. 5669–5679. [Online]. Available: https://aclanthology.org/2022.naacl-main.416

[123] P. Zhong, D. Wang, and C. Miao, “Knowledge-enriched transformer for emotion detection in textual conversations,” in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, Hong Kong, China, 2019, pp. 165–176.

[124] S. Mohammad, “Obtaining reliable human ratings of valence, arousal, and dominance for 20,000 English words,” in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Melbourne, Australia: Association for Computational Linguistics, Jul. 2018, pp. 174–184. [Online]. Available: https://aclanthology.org/P18-1017

[125] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, A. Rodriguez, A. Joulin, E. Grave, and G. Lample, “Llama: Open and efficient foundation language models,” 2023.

[126] S. Poria, N. Majumder, R. Mihalcea, and E. Hovy, “Emotion recognition in conversation: Research challenges, datasets, and recent advances,” *IEEE Access*, vol. 7, pp. 100 943–100 953, 2019.

[127] K. Sailunaz and R. Alhajj, “Emotion and sentiment analysis from twitter text,” *Journal of Computational Science*, vol. 36, p. 101003, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1877750318311037

[128] L. Dini and A. Bittar, “Emotion analysis on twitter: The hidden challenge,” in *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC’16)*, 2016, pp. 3953–3958.

[129] M. S. Akhtar, A. Ekbal, and E. Cambria, “How intense are you? predicting intensities of emotions and sentiments using stacked ensemble [application notes],” *IEEE Computational Intelligence Magazine*, vol. 15, no. 1, pp. 64–75, 2020.

[130] N. Gupta, M. Gilbert, and G. D. Fabbrizio, “Emotion detection in email customer care,” *Computational Intelligence*, vol. 29, no. 3, pp. 489–505, 2013.

[131] D. Hazarika, S. Poria, A. Zadeh, E. Cambria, L.-P. Morency, and R. Zimmermann, “Conversational memory network for emotion recognition in dyadic dialogue videos,” in *NAACL*, vol. 2018. NIH Public Access, 2018, p. 2122.

[132] J. Mooren and I. Van Krogten, “Contributions to the history of psychology: Cxii. magda b. arnold revisited: 1991,” *Psychological reports*, vol. 72, no. 1, pp. 67–84, 1993.

[133] R. S. Lazarus and S. Folkman, *Stress, appraisal, and coping*. Springer publishing company, 1984.

[134] R. W. Picard, A. Wexelblat, and C. I. N. I. Clifford I. Nass, “Future interfaces: social and emotional,” in *CHI’02 Extended Abstracts*, 2002, pp. 698–699.

[135] A. S. Cowen and D. Keltner, “Self-report captures 27 distinct categories of emotion bridged by continuous gradients,” *PNAS*, vol. 114, no. 38, pp. E7900–E7909, 2017.

[136] A. Mencattini, E. Martinelli, G. Costantini, M. Todisco, B. Basile, M. Bozzali, and C. Di Natale, “Speech emotion recognition using amplitude modulation parameters and a combined feature selection procedure,” *Knowledge-Based Systems*, vol. 63, pp. 68–81, 2014.

[137] L. Zhang, K. Mistry, S. C. Neoh, and C. P. Lim, “Intelligent facial emotion recognition using moth-firefly optimization,” *Knowledge-Based Systems*, vol. 111, pp. 248–267, 2016.

[138] H. Cui, A. Liu, X. Zhang, X. Chen, K. Wang, and X. Chen, “Eeg-based emotion recognition using an end-to-end regional-asymmetric convolutional neural network,” *Knowledge-Based Systems*, vol. 205, p. 106243, 2020.

[139] G. Assunção, B. Patrão, M. Castelo-Branco, and P. Menezes, “An overview of emotion in artificial intelligence,” *IEEE Transactions on Artificial Intelligence*, vol. 3, no. 6, pp. 867–886, 2022.

[140] Z. Lian, B. Liu, and J. Tao, “Smin: Semi-supervised multi-modal interaction network for conversational emotion recognition,” *IEEE Transactions on Affective Computing*, 2022.

[141] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. R. Salakhutdinov, and Q. V. Le, “Xlnet: Generalized autoregressive pretraining for language understanding,” in *Advances in Neural Information Processing Systems*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, Eds., vol. 32. Curran Associates, Inc., 2019.

[142] Z. Lian, B. Liu, and J. Tao, "Decn: Dialogical emotion correction network for conversational emotion recognition," *Neurocomputing*, pp. 483–495, 2021.

[143] Y. Shou, T. Meng, W. Ai, S. Yang, and K. Li, "Conversational emotion recognition studies based on graph convolutional neural networks and a dependent syntactic analysis," *Neurocomputing*, pp. 629–639, 2022.

[144] S. Y. M. Lee, Y. Chen, and C.-R. Huang, "A text-driven rule-based system for emotion cause detection," in *Proceedings of the NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*, 2010, pp. 45–53.

[145] L. Gui, R. Xu, D. Wu, Q. Lu, and Y. Zhou, "Event-driven emotion cause extraction with corpus construction," in *Social Media Content Analysis: Natural Language Processing and Beyond*. World Scientific, 2018, pp. 145–160.

[146] R. Xia and Z. Ding, "Emotion-cause pair extraction: A new task to emotion analysis in texts," 2019.

[147] R. Xia, M. Zhang, and Z. Ding, "Rthn: A rnn-transformer hierarchical network for emotion cause extraction," *arXiv preprint arXiv:1906.01236*, 2019.

[148] S. Poria, N. Majumder, D. Hazarika, D. Ghosal, R. Bhardwaj, S. Y. B. Jian, R. Ghosh, N. Chhaya, A. Gelbukh, and R. Mihalcea, "Recognizing emotion cause in conversations," *arXiv preprint arXiv:2012.11820*, 2020.

[149] R. S. Lazarus and S. Folkman, *Stress, appraisal, and coping*, 1984.

[150] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, 2017, pp. 5998–6008.

[151] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., vol. 30. Curran Associates, Inc., 2017.

[152] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *arXiv preprint arXiv:1412.3555*, 2014.

[153] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal loss for dense object detection," in *ICCV*, Venice, Italy, Oct 2017.

[154] P. D. Turney, "Thumbs up or thumbs down?: Semantic orientation applied to unsupervised classification of reviews," in *ACL*, 2002, pp. 417–424.

[155] B. Pang, , and L. Lee, "Seeing Stars: Exploiting Class Relationships for Sentiment Categorization with respect to Rating Scales," in *ACL*, 2005, pp. 115–124.

[156] 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.

[157] Y. Ma, H. Peng, T. Khan, E. Cambria, and A. Hussain, "Sentic LSTM: a Hybrid Network for Targeted Aspect-Based Sentiment Analysis," *Cognitive Computation*, vol. 10, pp. 639–650, 2018.

[158] P. Ekman, *Basic Emotions*. The handbook of cognition and emotion., 1999, pp. 45–60.

[159] B. Zhang, E. M. Provost, and G. Essl, “Cross-corpus acoustic emotion recognition with multi-task learning: Seeking common ground while preserving differences,” *IEEE Transactions on Affective Computing*, vol. 10, no. 1, pp. 85–99, 2019.

[160] M. S. Akhtar, A. Ekbal, and E. Cambria, “How Intense Are You? Predicting Intensities of Emotions and Sentiments using Stacked Ensemble,” *IEEE CIM*, vol. 15, pp. 64–75, 2020.

[161] S. Kumar, A. Shrimal, M. S. Akhtar, and T. Chakraborty, “Discovering emotion and reasoning its flip in multi-party conversations using masked memory network and transformer,” *arXiv preprint arXiv:2103.12360*, 2021.

[162] Y. Cai, H. Cai, and X. Wan, “Multi-modal sarcasm detection in twitter with hierarchical fusion model,” in *ACL*, 2019, pp. 2506–2515.

[163] S. Sangwan, M. S. Akhtar, P. Behera, and A. Ekbal, “I didn’t mean what I wrote! Exploring Multimodality for Sarcasm Detection,” in *IJCNN*, 2020.

[164] D. Bertero and P. Fung, “Deep learning of audio and language features for humor prediction,” in *LREC*, 2016, pp. 496–501.

[165] S. Poria, E. Cambria, D. Hazarika, N. Majumder, A. Zadeh, and L.-P. Morency, “Context-dependent sentiment analysis in user-generated videos,” in *ACL*, vol. 1, 2017, pp. 873–883.

[166] D. Ghosal, M. S. Akhtar, D. Chauhan, S. Poria, A. Ekbal, and P. Bhattacharyya, “Contextual Inter-modal Attention for Multi-modal Sentiment Analysis,” in *EMNLP*, 2018, pp. 3454–3466.

[167] A. Zadeh, P. P. Liang, S. Poria, E. Cambria, and L.-P. Morency, “Multimodal language analysis in the wild: Cmu-mosei dataset and interpretable dynamic fusion graph,” in *ACL*, 2018, pp. 2236–2246.

[168] M. S. Akhtar, D. Chauhan, D. Ghosal, S. Poria, A. Ekbal, and P. Bhattacharyya, “Multi-task Learning for Multi-modal Emotion Recognition and Sentiment Analysis,” in *NAACL-HLT*, 2019, pp. 370–379.

[169] M. S. Akhtar, D. S. Chauhan, and A. Ekbal, “A Deep Multi-Task Contextual Attention Framework for Multi-modal Affect Analysis,” *TKDD*, vol. 14, pp. 32:1–32:27, 2020.

[170] F. Noroozi, M. Marjanovic, A. Njegus, S. Escalera, and G. Anbarjafari, “Audio-visual emotion recognition in video clips,” *IEEE Transactions on Affective Computing*, vol. 10, no. 1, pp. 60–75, 2019.

[171] A. Conneau, G. Lample, M. Ranzato, L. Denoyer, and H. Jégou, “Word translation without parallel data,” *arXiv preprint arXiv:1710.04087*, 2017.

[172] G. Lample, A. Conneau, L. Denoyer, and M. Ranzato, “Unsupervised machine translation using monolingual corpora only,” *arXiv preprint arXiv:1711.00043*, 2017.

[173] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural computation*, vol. 9, pp. 1735–1780, 1997.

[174] S. Castro, D. Hazarika, V. Pérez-Rosas, R. Zimmermann, R. Mihalcea, and S. Poria, “Towards multimodal sarcasm detection (an _Obviously_ perfect paper),” in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence, Italy: Association for Computational Linguistics, Jul. 2019, pp. 4619–4629. [Online]. Available: https://aclanthology.org/P19-1455

[175] J. Wu, H. Lin, L. Yang, and B. Xu, “Mumor: A multimodal dataset for humor detection in conversations,” in *Natural Language Processing and Chinese Computing*, L. Wang, Y. Feng, Y. Hong, and R. He, Eds. Cham: Springer International Publishing, 2021, pp. 619–627.

[176] D. Davidov, O. Tsur, and A. Rappoport, “Semi-supervised recognition of sarcastic sentences in twitter and amazon,” in *CoNLL*, 2010, p. 107–116.

[177] A. Joshi, V. Sharma, and P. Bhattacharyya, “Harnessing context incongruity for sarcasm detection,” in *ACL*, 2015, pp. 757–762.

[178] M. Bouazizi and T. Otsuki Ohtsuki, “A pattern-based approach for sarcasm detection on twitter,” *IEEE Access*, vol. 4, pp. 5477–5488, 2016.

[179] A. Joshi, P. Bhattacharyya, and M. J. Carman, “Automatic sarcasm detection: A survey,” *ACM Comput. Surv.*, vol. 50, 2017.

[180] L. H. Son, A. Kumar, S. R. Sangwan, A. Arora, A. Nayyar, and M. Abdel-Basset, “Sarcasm detection using soft attention-based bidirectional long short-term memory model with convolution network,” *IEEE Access*, vol. 7, pp. 23 319–23 328, 2019.

[181] A. Khattri, A. Joshi, P. Bhattacharyya, and M. Carman, “Your sentiment precedes you: Using an author’s historical tweets to predict sarcasm,” in *Proceedings of the 6th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, 2015, pp. 25–30.

[182] D. Hazarika, S. Poria, S. Gorantla, E. Cambria, R. Zimmermann, and R. Mihalcea, “CASCADE: Contextual sarcasm detection in online discussion forums,” in *COLING*, 2018, pp. 1837–1848.

[183] D. Ghosh, A. Richard Fabbri, and S. Muresan, “The role of conversation context for sarcasm detection in online interactions,” in *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue*, 2017, pp. 186–196.

[184] S. Swami, A. Khandelwal, V. Singh, S. S. Akhtar, and M. Shrivastava, “A corpus of english-hindi code-mixed tweets for sarcasm detection,” *CoRR*, vol. abs/1805.11869, 2018.

[185] S. K. Bharti, K. Sathya Babu, and S. K. Jena, “Harnessing online news for sarcasm detection in hindi tweets,” in *Pattern Recognition and Machine Intelligence*, 2017, pp. 679–686.

[186] P. Potash, A. Romanov, and A. Rumshisky, “SemEval-2017 task 6: #HashtagWars: Learning a sense of humor,” in *SemEval*, 2017, pp. 49–57.

[187] A. Purandare and D. Litman, “Humor: Prosody analysis and automatic recognition for f*r*i*e*n*d*s*.” 01 2006, pp. 208–215.

[188] D. Bertero and P. Fung, “Predicting humor response in dialogues from tv sitcoms,” in *ICASSP*, 2016, pp. 5780–5784.

[189] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, “Attention is All you Need,” in *NIPS*, 2017, pp. 5998–6008.

[190] A. Zadeh, P. P. Liang, N. Mazumder, S. Poria, E. Cambria, and L. Morency, “Memory fusion network for multi-view sequential learning,” in *AAAI*, 2018, pp. 5634–5641.

[191] A. Khandelwal, S. Swami, S. S. Akhtar, and M. Shrivastava, “Humor detection in English-Hindi code-mixed social media content : Corpus and baseline system,” in *LREC*, 2018.

[192] S. R. Sane, S. Tripathi, K. R. Sane, and R. Mamidi, "Deep learning techniques for humor detection in Hindi-English code-mixed tweets," in *Proceedings of the Tenth Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, 2019, pp. 57–61.

[193] Brian McFee, Colin Raffel, Dawen Liang, Daniel P.W. Ellis, Matt McVicar, Eric Battenberg, and Oriol Nieto, "librosa: Audio and Music Signal Analysis in Python," in *Proceedings of the 14th Python in Science Conference*, 2015, pp. 18 – 24.

[194] S. Abdoli, P. Cardinal, and A. L. Koerich, "End-to-end environmental sound classification using a 1d convolutional neural network," *Expert Systems with Applications*, vol. 136, pp. 252–263, 2019.

[195] R. V. Sharan and T. J. Moir, "Acoustic event recognition using cochleagram image and convolutional neural networks," *Applied Acoustics*, vol. 148, pp. 62–66, 2019.

[196] A. Khamparia, D. Gupta, N. G. Nguyen, A. Khanna, B. Pandey, and P. Tiwari, "Sound classification using convolutional neural network and tensor deep stacking network," *IEEE Access*, vol. 7, pp. 7717–7727, 2019.

[197] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *CVPR*, 2016, pp. 770–778.

[198] J. Arevalo, T. Solorio, M. Montes-y Gómez, and F. A. González, "Gated multimodal units for information fusion," *arXiv preprint arXiv:1702.01992*, 2017.

[199] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A Simple Way to Prevent Neural Networks from Overfitting," *JMLR*, vol. 15, pp. 1929–1958, 2014.

[200] X. Glorot, A. Bordes, and Y. Bengio, "Deep Sparse Rectifier Neural Networks," in *AISTATS*, 2011, p. 315–323.

[201] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," *CoRR*, vol. abs/1412.6980, 2014.

[202] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *CVPR*, 2015, pp. 1–9.

[203] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, pp. 273–297, 1995.

[204] R. M. Roberts and R. J. Kreuz, "Why do people use figurative language?" *Psychological Science*, vol. 5, no. 3, pp. 159–163, 1994. [Online]. Available: https://doi.org/10.1111/j.1467-9280.1994.tb00653.x

[205] H. Olkoniemi, H. Ranta, and J. K. Kaakinen, "Individual differences in the processing of written sarcasm and metaphor: Evidence from eye movements." *Journal of Experimental Psychology: Learning, Memory, and Cognition*, vol. 42, no. 3, p. 433, 2016.

[206] H. L. Colston, "Salting a wound or sugaring a pill: The pragmatic functions of ironic criticism," *Discourse Processes*, vol. 23, no. 1, pp. 25–45, 1997. [Online]. Available: https://doi.org/10.1080/01638539709544980

[207] H. L. Colston and S. B. Keller, "You'll never believe this: Irony and hyperbole in expressing surprise," *Journal of psycholinguistic research*, vol. 27, no. 4, pp. 499–513, 1998.

[208] S. L. Ivanko and P. M. Pexman, "Context incongruity and irony processing," *Discourse Processes*, vol. 35, no. 3, pp. 241–279, 2003. [Online]. Available: https://doi.org/10.1207/S15326950DP3503_2

[209] H. M. Wellman, *Making minds: How theory of mind develops.* Oxford University Press, 2014.

[210] D. Ghosh, A. R. Fabbri, and S. Muresan, “Sarcasm analysis using conversation context,” *Computational Linguistics*, vol. 44, no. 4, pp. 755–792, Dec. 2018. [Online]. Available: https://aclanthology.org/J18-4009

[211] S. Castro, D. Hazarika, V. Pérez-Rosas, R. Zimmermann, R. Mihalcea, and S. Poria, “Towards multimodal sarcasm detection (an _Obviously_ perfect paper),” in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics.* Florence, Italy: Association for Computational Linguistics, Jul. 2019, pp. 4619–4629. [Online]. Available: https://aclanthology.org/P19-1455

[212] S. Oraby, V. Harrison, A. Misra, E. Riloff, and M. Walker, “Are you serious?: Rhetorical questions and sarcasm in social media dialog,” in *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue.* Saarbrücken, Germany: Association for Computational Linguistics, Aug. 2017, pp. 310–319. [Online]. Available: https://aclanthology.org/W17-5537

[213] A. Mishra, T. Tater, and K. Sankaranarayanan, “A modular architecture for unsupervised sarcasm generation,” in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP).* Hong Kong, China: Association for Computational Linguistics, Nov. 2019, pp. 6144–6154. [Online]. Available: https://aclanthology.org/D19-1636

[214] A. Dubey, A. Joshi, and P. Bhattacharyya, “Deep models for converting sarcastic utterances into their non sarcastic interpretation,” in *Proceedings of the ACM India Joint International Conference on Data Science and Management of Data*, ser. CoDS-COMAD ’19. New York, NY, USA: Association for Computing Machinery, 2019, p. 289–292. [Online]. Available: https://doi.org/10.1145/3297001.3297043

[215] T. Chakrabarty, D. Ghosh, S. Muresan, and N. Peng, “Rˆ3: Reverse, retrieve, and rank for sarcasm generation with commonsense knowledge,” in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics.* Online: Association for Computational Linguistics, Jul. 2020, pp. 7976–7986. [Online]. Available: https://aclanthology.org/2020.acl-main.711

[216] M. K. Hasan, S. Lee, W. Rahman, A. Zadeh, R. Mihalcea, L.-P. Morency, and E. Hoque, “Humor knowledge enriched transformer for understanding multimodal humor,” *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 14, pp. 12 972–12 980, May 2021. [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/17534

[217] D. S. Chauhan, D. S R, A. Ekbal, and P. Bhattacharyya, “Sentiment and emotion help sarcasm? a multi-task learning framework for multi-modal sarcasm, sentiment and emotion analysis,” in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics.* Online: Association for Computational Linguistics, Jul. 2020, pp. 4351–4360. [Online]. Available: https://aclanthology.org/2020.acl-main.401

[218] A. Joshi, P. Bhattacharyya, and M. J. Carman, “Automatic sarcasm detection: A survey,” *ACM Comput. Surv.*, vol. 50, no. 5, Sep. 2017. [Online]. Available: https://doi.org/10.1145/3124420

[219] L. Peled and R. Reichart, “Sarcasm SIGN: Interpreting sarcasm with sentiment based monolingual machine translation,” in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers).* Vancouver, Canada: Association for Computational Linguistics, Jul. 2017, pp. 1690–1700. [Online]. Available: https://aclanthology.org/P17-1155

[220] D. Ghosh, A. Richard Fabbri, and S. Muresan, "The role of conversation context for sarcasm detection in online interactions," in *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue*. Saarbrücken, Germany: Association for Computational Linguistics, Aug. 2017, pp. 186–196. [Online]. Available: https://aclanthology.org/W17-5523

[221] S. K. Bharti, K. Sathya Babu, and S. K. Jena, "Harnessing online news for sarcasm detection in hindi tweets," in *Pattern Recognition and Machine Intelligence*, B. U. Shankar, K. Ghosh, D. P. Mandal, S. S. Ray, D. Zhang, and S. K. Pal, Eds. Cham: Springer International Publishing, 2017, pp. 679–686.

[222] S. Swami, A. Khandelwal, V. Singh, S. S. Akhtar, and M. Shrivastava, "A corpus of english-hindi code-mixed tweets for sarcasm detection," *arXiv preprint arXiv:1805.11869*, 2018.

[223] I. Abu Farha and W. Magdy, "From Arabic sentiment analysis to sarcasm detection: The ArSarcasm dataset," in *Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, with a Shared Task on Offensive Language Detection*. Marseille, France: European Language Resource Association, May 2020, pp. 32–39. [Online]. Available: https://aclanthology.org/2020.osact-1.5

[224] R. Ortega-Bueno, F. Rangel, D. Hernández Farıas, P. Rosso, M. Montes-y Gómez, and J. E. Medina Pagola, "Overview of the task on irony detection in spanish variants," in *Proceedings of the Iberian languages evaluation forum (IberLEF 2019), co-located with 34th conference of the Spanish Society for natural language processing (SEPLN 2019). CEUR-WS. org*, vol. 2421, 2019, pp. 229–256. [Online]. Available: http://ceur-ws.org/Vol-2421/IroSvA_overview.pdf

[225] A. T. Cignarella, S. Frenda, V. Basile, C. Bosco, V. Patti, P. Rosso *et al.*, "Overview of the evalita 2018 task on irony detection in italian tweets (ironita)," in *Sixth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian (EVALITA 2018)*, vol. 2263. CEUR-WS, 2018, pp. 1–6. [Online]. Available: http://ceur-ws.org/Vol-2263/paper005.pdf

[226] N. Xu, Z. Zeng, and W. Mao, "Reasoning with multimodal sarcastic tweets via modeling cross-modality contrast and semantic association," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics, Jul. 2020, pp. 3777–3786. [Online]. Available: https://aclanthology.org/2020.acl-main.349

[227] H. Pan, Z. Lin, P. Fu, Y. Qi, and W. Wang, "Modeling intra and inter-modality incongruity for multi-modal sarcasm detection," in *Findings of the Association for Computational Linguistics: EMNLP 2020*. Online: Association for Computational Linguistics, Nov. 2020, pp. 1383–1392. [Online]. Available: https://aclanthology.org/2020.findings-emnlp.124

[228] C.-Y. Lin, "ROUGE: A package for automatic evaluation of summaries," in *Text Summarization Branches Out*. Barcelona, Spain: Association for Computational Linguistics, Jul. 2004, pp. 74–81. [Online]. Available: https://aclanthology.org/W04-1013

[229] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "Bleu: a method for automatic evaluation of machine translation," in *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*. Philadelphia, Pennsylvania, USA: Association for Computational Linguistics, Jul. 2002, pp. 311–318. [Online]. Available: https://aclanthology.org/P02-1040

[230] M. Denkowski and A. Lavie, "Meteor universal: Language specific translation evaluation for any target language," in *Proceedings of the Ninth Workshop on Statistical Machine Translation*. Baltimore, Maryland, USA: Association for Computational Linguistics, Jun. 2014, pp. 376–380. [Online]. Available: https://aclanthology.org/W14-3348

[231] F. Eyben, K. R. Scherer, B. W. Schuller, J. Sundberg, E. André, C. Busso, L. Y. Devillers, J. Epps, P. Laukka, S. S. Narayanan, and K. P. Truong, "The geneva minimalistic acoustic parameter set (gemaps) for voice research and affective computing," *IEEE Transactions on Affective Computing*, vol. 7, no. 2, pp. 190–202, 2016.

[232] K. Hara, H. Kataoka, and Y. Satoh, "Can spatiotemporal 3d cnns retrace the history of 2d cnns and imagenet?" in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. [Online]. Available: https://openaccess.thecvf.com/content_cvpr_2018/html/Hara_Can_Spatiotemporal_3D_CVPR_2018_paper.html

[233] W. Kay, J. Carreira, K. Simonyan, B. Zhang, C. Hillier, S. Vijayanarasimhan, F. Viola, T. Green, T. Back, P. Natsev, M. Suleyman, and A. Zisserman, "The kinetics human action video dataset," 2017.

[234] A. See, P. J. Liu, and C. D. Manning, "Get to the point: Summarization with pointer-generator networks," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Vancouver, Canada: Association for Computational Linguistics, Jul. 2017, pp. 1073–1083. [Online]. Available: https://aclanthology.org/P17-1099

[235] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, and L. Zettlemoyer, "BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics, Jul. 2020, pp. 7871–7880. [Online]. Available: https://aclanthology.org/2020.acl-main.703

[236] Y. Liu, J. Gu, N. Goyal, X. Li, S. Edunov, M. Ghazvininejad, M. Lewis, and L. Zettlemoyer, "Multilingual denoising pre-training for neural machine translation," *Transactions of the Association for Computational Linguistics*, vol. 8, pp. 726–742, 2020. [Online]. Available: https://aclanthology.org/2020.tacl-1.47

[237] S. Attardo, J. Eisterhold, J. Hay, and I. Poggi, "Multimodal markers of irony and sarcasm." *Humor: International Journal of Humor Research*, vol. 16, no. 2, 2003. [Online]. Available: https://psycnet.apa.org/doi/10.1515/humr.2003.012

[238] S. Tabacaru and M. Lemmens, "Raised eyebrows as gestural triggers in humour: The case of sarcasm and hyper-understanding," *The European Journal of Humour Research*, vol. 2, no. 2, p. 11–31, Oct. 2014. [Online]. Available: https://www.europeanjournalofhumour.org/ejhr/article/view/56

[239] P. Rockwell, "Vocal features of conversational sarcasm: A comparison of methods," *Journal of psycholinguistic research*, vol. 36, no. 5, pp. 361–369, 2007.

[240] R. C. Schank, "Conceptual dependency: A theory of natural language understanding," *Cognitive psychology*, vol. 3, no. 4, pp. 552–631, 1972.

[241] Y. Pruksachatkun, J. Phang, H. Liu, P. M. Htut, X. Zhang, R. Y. Pang, C. Vania, K. Kann, and S. R. Bowman, "Intermediate-task transfer learning with pretrained models for natural language understanding: When and why does it work?" *arXiv preprint arXiv:2005.00628*, 2020.

[242] X. Zhou, X. Tao, J. Yong, and Z. Yang, "Sentiment analysis on tweets for social events," in *Proceedings of the 2013 IEEE 17th international conference on computer supported cooperative work in design (CSCWD)*. IEEE, 2013, pp. 557–562.

[243] P. Badjatiya, S. Gupta, M. Gupta, and V. Varma, "Deep learning for hate speech detection in tweets," in *Proceedings of the 26th international conference on World Wide Web companion*, 2017, pp. 759–760.

[244] S. Poria, N. Majumder, R. Mihalcea, and E. Hovy, "Emotion recognition in conversation: Research challenges, datasets, and recent advances," *IEEE Access*, vol. 7, pp. 100 943–100 953, 2019.

[245] W. Jiao, M. Lyu, and I. King, "Real-time emotion recognition via attention gated hierarchical memory network," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 05, 2020, pp. 8002–8009.

[246] W. Shen, J. Chen, X. Quan, and Z. Xie, "Dialogxl: All-in-one xlnet for multi-party conversation emotion recognition," *arXiv preprint arXiv:2012.08695*, 2020.

[247] S. Larson, A. Mahendran, J. J. Peper, C. Clarke, A. Lee, P. Hill, J. K. Kummerfeld, K. Leach, M. A. Laurenzano, L. Tang *et al.*, "An evaluation dataset for intent classification and out-of-scope prediction," *arXiv preprint arXiv:1909.02027*, 2019.

[248] R. Gangadharaiah and B. Narayanaswamy, "Joint multiple intent detection and slot labeling for goal-oriented dialog," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 2019, pp. 564–569.

[249] L. Qin, Z. Li, W. Che, M. Ni, and T. Liu, "Co-gat: A co-interactive graph attention network for joint dialog act recognition and sentiment classification," *arXiv preprint arXiv:2012.13260*, 2020.

[250] Y. Liu, K. Han, Z. Tan, and Y. Lei, "Using context information for dialog act classification in dnn framework," in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 2017, pp. 2170–2178.

[251] Z. Lin, P. Xu, G. I. Winata, F. B. Siddique, Z. Liu, J. Shin, and P. Fung, "Caire: An end-to-end empathetic chatbot," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 09, 2020, pp. 13 622–13 623.

[252] J. Shin, P. Xu, A. Madotto, and P. Fung, "Happybot: Generating empathetic dialogue responses by improving user experience look-ahead," *arXiv preprint arXiv:1906.08487*, 2019.

[253] H. Rashkin, E. M. Smith, M. Li, and Y.-L. Boureau, "Towards empathetic open-domain conversation models: A new benchmark and dataset," *arXiv preprint arXiv:1811.00207*, 2018.

[254] Y. Su, D. Cai, Y. Wang, S. Baker, A. Korhonen, N. Collier, and X. Liu, "Stylistic dialogue generation via information-guided reinforcement learning strategy," *arXiv preprint arXiv:2004.02202*, 2020.

[255] R. Akama, K. Inada, N. Inoue, S. Kobayashi, and K. Inui, "Generating stylistically consistent dialog responses with transfer learning," in *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, 2017, pp. 408–412.

[256] C. Danescu-Niculescu-Mizil and L. Lee, "Chameleons in imagined conversations: A new approach to understanding coordination of linguistic style in dialogs," *arXiv preprint arXiv:1106.3077*, 2011.

[257] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "Ro{bert}a: A robustly optimized {bert} pretraining approach," 2020. [Online]. Available: https://openreview.net/forum?id=SyxS0T4tvS

[258] F. Alam and G. Riccardi, “Fusion of acoustic, linguistic and psycholinguistic features for speaker personality traits recognition,” in *2014 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2014, pp. 955–959.

[259] M. I. Briggs and P. B. Myers, *Gifts Differing : Understanding Personality Type*. Davies-Black Publishing, 1995.

[260] J. N. Butcher and C. L. Williams, “Personality assessment with the mmpi-2: Historical roots, international adaptations, and current challenges,” *Applied Psychology: Health and Well-Being*, vol. 1, no. 1, pp. 105–135, 2009. [Online]. Available: https://iaap-journals.onlinelibrary.wiley.com/doi/abs/10.1111/j.1758-0854.2008.01007.x

[261] A. Benjamin Jr, *Type A/B Personalities*, 11 2020.

[262] J. M. Digman, “Personality structure: Emergence of the five-factor model,” *Annual Review of Psychology*, vol. 41, no. 1, pp. 417–440, 1990. [Online]. Available: https://doi.org/10.1146/annurev.ps.41.020190.002221

[263] A. Tigunova, A. Yates, P. Mirza, and G. Weikum, “Listening between the lines: Learning personal attributes from conversations,” in *The World Wide Web Conference*, 2019, pp. 1818–1828.

[264] C.-S. Wu, A. Madotto, Z. Lin, P. Xu, and P. Fung, “Getting to know you: User attribute extraction from dialogues,” *arXiv preprint arXiv:1908.04621*, 2019.

[265] J.-C. Gu, Z.-H. Ling, Y. Wu, Q. Liu, Z. Chen, and X. Zhu, “Detecting speaker personas from conversational texts,” *arXiv preprint arXiv:2109.01330*, 2021.

[266] H. Chen, X. Liu, D. Yin, and J. Tang, “A survey on dialogue systems: Recent advances and new frontiers,” *SIGKDD Explor. Newsl.*, vol. 19, no. 2, p. 25–35, nov 2017. [Online]. Available: https://doi.org/10.1145/3166054.3166058

[267] S. Kumar, S. Bhatia, M. Aggarwal, and T. Chakraborty, “Dialogue agents 101: A beginner’s guide to critical ingredients for designing effective conversational systems,” 2023.

[268] E. Ahn, C. Jimenez, Y. Tsvetkov, and A. W. Black, “What code-switching strategies are effective in dialog systems?” in *Proceedings of the Society for Computation in Linguistics 2020*, 2020, pp. 254–264.

[269] Z. Liu, G. I. Winata, Z. Lin, P. Xu, and P. Fung, “Attention-informed mixed-language training for zero-shot cross-lingual task-oriented dialogue systems,” *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 05, pp. 8433–8440, Apr. 2020. [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/6362

[270] M. Firdaus, A. Ekbal, and E. Cambria, “Multitask learning for multilingual intent detection and slot filling in dialogue systems,” *Information Fusion*, vol. 91, pp. 299–315, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1566253522001671

[271] S. Modha, T. Mandl, G. K. Shahi, H. Madhu, S. Satapara, T. Ranasinghe, and M. Zampieri, “Overview of the hasoc subtrack at fire 2021: Hate speech and offensive content identification in english and indo-aryan languages and conversational hate speech,” in *Proceedings of the 13th Annual Meeting of the Forum for Information Retrieval Evaluation*, 2021, pp. 1–3.

[272] M. Bedi, S. Kumar, M. S. Akhtar, and T. Chakraborty, “Multi-modal sarcasm detection and humor classification in code-mixed conversations,” *IEEE Transactions on Affective Computing*, 2021.

[273] A. Gottardi, O. Ipek, G. Castellucci, S. Hu, L. Vaz, Y. Lu, A. Khatri, A. Chadha, D. Zhang, S. Sahai, P. Dwivedi, H. Shi, L. Hu, A. Huang, L. Dai, B. Yang, V. Somani, P. Rajan, R. Rezac, and Y. Maarek, "Alexa, let's work together: Introducing the first alexa prize taskbot challenge on conversational task assistance," 09 2022.

[274] T. Spring, J. Casas, K. Daher, E. Mugellini, and O. Abou Khaled, "Empathic response generation in chatbots," in *Proceedings of 4th Swiss Text Analytics Conference (SwissText 2019), 18-19 June 2019, Wintherthur, Switzerland.* 18-19 June 2019, 2019.

[275] Y. Fan, X. Luo, and P. Lin, "A survey of response generation of dialogue systems," *International Journal of Computer and Information Engineering*, vol. 14, no. 12, pp. 461–472, 2020.

[276] C. Dong, Y. Li, H. Gong, M. Chen, J. Li, Y. Shen, and M. Yang, "A survey of natural language generation," *ACM Comput. Surv.*, vol. 55, no. 8, dec 2022. [Online]. Available: https://doi.org/10.1145/3554727

[277] V. Agarwal, P. Rao, and D. B. Jayagopi, "Towards code-mixed Hinglish dialogue generation," in *Proceedings of the 3rd Workshop on Natural Language Processing for Conversational AI.* Online: Association for Computational Linguistics, Nov. 2021, pp. 271–280. [Online]. Available: https://aclanthology.org/2021.nlp4convai-1.26

[278] G. V. Singh, M. Firdaus, Shambhavi, S. Mishra, and A. Ekbal, "Knowing what to say: Towards knowledge grounded code-mixed response generation for open-domain conversations," *Knowledge-Based Systems*, vol. 249, p. 108900, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0950705122004300

[279] A. Bawa, P. Khadpe, P. Joshi, K. Bali, and M. Choudhury, "Do multilingual users prefer chat-bots that code-mix? let's nudge and find out!" *Proc. ACM Hum.-Comput. Interact.*, vol. 4, no. CSCW1, may 2020. [Online]. Available: https://doi.org/10.1145/3392846

[280] P. T. Costa and R. R. McCrae, "Normal personality assessment in clinical practice: The neo personality inventory." *Psychological assessment*, vol. 4, no. 1, p. 5, 1992.

[281] P. T. Costa Jr and R. R. McCrae, *The Revised Neo Personality Inventory (neo-pi-r).* Sage Publications, Inc, 2008.

[282] F. Mairesse, M. A. Walker, M. R. Mehl, and R. K. Moore, "Using linguistic cues for the automatic recognition of personality in conversation and text," *Journal of artificial intelligence research*, vol. 30, pp. 457–500, 2007.

[283] J. Golbeck, C. Robles, and K. Turner, "Predicting personality with social media," in *CHI'11 extended abstracts on human factors in computing systems*, 2011, pp. 253–262.

[284] M. Kosinski, D. Stillwell, and T. Graepel, "Private traits and attributes are predictable from digital records of human behavior," *Proceedings of the national academy of sciences*, vol. 110, no. 15, pp. 5802–5805, 2013.

[285] H. A. Schwartz, J. C. Eichstaedt, M. L. Kern, L. Dziurzynski, S. M. Ramones, M. Agrawal, A. Shah, M. Kosinski, D. Stillwell, M. E. Seligman *et al.*, "Personality, gender, and age in the language of social media: The open-vocabulary approach," *PloS one*, vol. 8, no. 9, p. e73791, 2013.

[286] F. Celli and B. Lepri, "Is big five better than mbti? a personality computing challenge using twitter data." in *CLiC-it*, 2018.

[287] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, “Smote: synthetic minority over-sampling technique,” *Journal of artificial intelligence research*, vol. 16, pp. 321–357, 2002.

[288] D. Bahdanau, K. Cho, and Y. Bengio, “Neural machine translation by jointly learning to align and translate,” *arXiv preprint arXiv:1409.0473*, 2014.

[289] S. Lloyd, “Least squares quantization in pcm,” *IEEE transactions on information theory*, vol. 28, no. 2, pp. 129–137, 1982.

[290] J. MacQueen *et al.*, “Some methods for classification and analysis of multivariate observations,” in *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, vol. 1, no. 14. Oakland, CA, USA, 1967, pp. 281–297.

[291] H. Zhang, H. Xu, and T.-E. Lin, “Deep open intent classification with adaptive decision boundary,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 16, 2021, pp. 14 374–14 382.

[292] J. Ho, N. Kalchbrenner, D. Weissenborn, and T. Salimans, “Axial attention in multidimensional transformers,” 2020. [Online]. Available: https://openreview.net/forum?id=H1e5GJBtDr

[293] T. Mikolov, K. Chen, G. Corrado, and J. Dean, “Efficient estimation of word representations in vector space,” *arXiv preprint arXiv:1301.3781*, 2013.

[294] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, P. J. Liu *et al.*, “Exploring the limits of transfer learning with a unified text-to-text transformer.” *J. Mach. Learn. Res.*, vol. 21, no. 140, pp. 1–67, 2020.

[295] T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, and Y. Artzi, “Bertscore: Evaluating text generation with bert,” in *International Conference on Learning Representations*, 2019. [Online]. Available: https://openreview.net/forum?id=SkeHuCVFDr

[296] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, “Exploring the limits of transfer learning with a unified text-to-text transformer,” *J. Mach. Learn. Res.*, vol. 21, no. 1, jan 2020.

[297] M. L. McHugh, “Interrater reliability: the kappa statistic,” *Biochemia medica*, vol. 22, no. 3, pp. 276–282, 2012.