EXPLORING EMOTIONAL FLOW IN DYADIC INTERACTIONS: A STUDY OF CONTEXTUAL AND NON-CONTEXTUAL SENTIMENT MODELS IN TURKISH COUPLE DIALOGUES

ESMA NAFIYE POLAT

ÖZYEĞİN UNIVERSITY

FEBRUARY, 2025

EXPLORING EMOTIONAL FLOW IN DYADIC INTERACTIONS: A STUDY OF CONTEXTUAL AND NON-CONTEXTUAL SENTIMENT MODELS IN TURKISH COUPLE DIALOGUES

by
Esma Nafiye Polat

Dissertation
Submitted in Partial Fulfillment of the
Requirements for the Degree of

Doctor of Philosophy

in
Computer Science

Advisor: Assoc. Prof. Cenk Demiroğlu
Coadvisor: Prof. Olcay Taner Yıldız

Graduate School of Science and Engineering
Özyeğin University
İstanbul

February, 2025

EXPLORING EMOTIONAL FLOW IN DYADIC INTERACTIONS: A STUDY OF CONTEXTUAL AND NON-CONTEXTUAL SENTIMENT MODELS IN TURKISH COUPLE DIALOGUES

Approved by:

Assoc. Prof. Cenk Demiroğlu, Advisor
Department of Electrical & Electronics Engineering
Özyeğin University

Asst. Prof. Mustafa Furkan Kıraç
Department of Computer Science
Özyeğin University

Assoc. Prof. Arzucan Özgür
Department of Computer Engineering
Boğaziçi University

Prof. Hasan Sözer
Department of Computer Science
Özyeğin University

Prof. Ümit Güz
Department of Electrical & Electronics Engineering
Işık University

Approval Date: December 26, 2024

To . . .

Those whose right to education has been stolen by war and violence, and

to my mum, who ensured her daughters could dream, learn, and grow.

DECLARATION OF ORIGINALITY

I hereby declare that I am the sole author of this thesis and that this is the true copy of my thesis, including the final revisions, approved by my thesis committee. All data and information have been obtained, produced, and presented in accordance with the rules of research ethics and principles of academic honesty. As required by these rules, to the best of my knowledge I have acknowledged ideas, thoughts, and any copyrighted material in accordance with the standard referencing rules. I certify that any part of this thesis has not been submitted for a degree or diploma in another educational institution.

Esma Nafiye Polat

ABSTRACT

This thesis introduces the "Couple Dialogue" dataset, tailored for analyzing conversational sentiment in Turkish. The dataset includes 14,294 utterances from 118 couple conversations, each meticulously annotated to reflect sentiment shifts and interpersonal interactions. The study contrasts two primary modeling approaches: the Non-Contextual Model and the Contextual Model. The Non-Contextual Model treats utterances in isolation, neglecting the intricacies of conversational dynamics and sentiment progression. This model incorporates an in-depth morphological analysis to account for the agglutinative and morphologically complex nature of the Turkish language, including diverse word forms and negation morphemes essential for sentiment analysis. Conversely, the Contextual Model leverages cutting-edge Large Language Models (LLMs) such as BERT, GPT-3.5 Turbo, Llama 2, and GPT-4, in addition to structures like DialogueRNN. This approach considers the sequential and relational dimensions of dialogues through prompt-based, fine-tuned, and embedding-based methods, with a focus on advanced embedding techniques and fine-tuning, particularly using pre-trained and fine-tuned Turkish BERT. The Contextual Model significantly outperforms the Non-Contextual Model, with a 9.98% enhancement in Weighted F1 scores, as validated by statistical tests. This research not only pioneers the application of LLMs in Turkish conversational sentiment analysis but also emphasizes the importance of contextual comprehension in discerning complex emotional nuances in couple dialogues. It establishes a strong foundation for future research in sentiment analysis within linguistically diverse contexts.

Keywords: Contextual Model, Conversational Sentiment Analysis, Couple Dialogue Dataset, DialogueRNN, Fine Tuning, GPT, Llama 2, LLMs, Morphological Analysis, Non-Contextual Model, Turkish Language

ÖZET

Bu tez, Türkçe'deki konuşma duygu analizine yönelik özel olarak hazırlanmış "Çift Diyaloğu" veri setini tanıtmaktadır. Veri seti, duygu değişimlerini ve kişiler arası etkileşimleri yansıtacak şekilde titizlikle anotlanmış 118 çiftin konuşmasından oluşan 14.294 ifadeyi içermektedir. Çalışma, iki temel modelleme yaklaşımını karşılaştırmaktadır: Bağlam Dışı Model ve Bağlamsal Model. Bağlam Dışı Model, ifadeleri bağımsız olarak ele alır ve konuşma dinamikleri ile duygu ilerlemesinin inceliklerini göz ardı eder. Bu model, Türkçe'nin eklemeli ve morfolojik olarak karmaşık yapısını hesaba katmak için çeşitli kelime formları ve duygu analizi için gerekli olan olumsuzlama eklerini içeren detaylı bir morfolojik analiz yapar. Buna karşılık, Bağlamsal Model, BERT, GPT-3.5 Turbo, Llama 2 ve GPT-4 gibi son teknoloji Büyük Dil Modellerini (LLM'ler) ve ayrıca DialogueRNN gibi yapıları kullanır. Bu yaklaşım, özellikle önceden eğitilmiş ve ince ayar yapılmış Türkçe BERT kullanarak, gelişmiş gömme teknikleri ve ince ayar ile geliştirilmiş yöntemlerle diyalogların sıralı ve ilişkisel boyutlarını dikkate alır. Bağlamsal Model, istatistiksel testlerle doğrulanan 9.98%'lik bir artışla, Bağlam Dışı Model'den önemli ölçüde daha iyi performans gösterir. Bu araştırma, LLM'lerin Türkçe konuşma duygu analizinde uygulanmasını öncü niteliğinde olup, çift diyaloglarındaki karmaşık duygusal ipuçlarının yakalanmasında bağlamsal anlayışın önemini vurgular. Çeşitli dilbilimsel bağlamlarda duygu analizine yönelik gelecekteki araştırmalar için sağlam bir temel oluşturur.

Anahtar Kelimeler: Bağlamsal Model, Konuşma Duygu Analizi, Çift Diyalog Veri Seti, DialogueRNN, İnce Ayar, GPT, Llama 2, Büyük Dil Modelleri (LLM'ler), Morfolojik Analiz, Bağlamsız Model, Türkçe

ACKNOWLEDGEMENTS

The data of this study were collected in a project funded by TUBITAK 1001 Program. (The Scientific and Research Council of Turkey; Project No: 113K538).

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ACRONYMS AND ABBREVIATIONS

LLMs	Large Language Models
NLP	Natural Language Processing
ICON	Interactive Conversational Memory Network
GRU	Gated Recurrent Unit
BoW	Bag Of Words
CNN	Convolutional Neural Network
MLP	Multilayer Perceptron
SVM	Support Vector Machine
RNN	Recurrent Neural Network
NLU	Natural Language Understanding
GPU	graphics processing unit
OpenAI	American artificial intelligence (AI) research organization
BERT	Bidirectional Encoder Representations from Transformers
ISO	International Organization for Standardization
MBERT	multilingual BERT
NLG	Natural Language Generation
HTML	Hypertext Markup Language

BART	Bidirectional and Auto-Regressive Transformer
EMNLP	Empirical Methods in Natural Language Processing
SIGIR	Special Interest Group on Information Retrieval
ICECCO	International Conference on Electronics Computer and Computation
RLHF	Reinforcement Learning from Human Feedback
XML	Extensible Markup Language
RAM	random-access memory
TF	Term Frequency
IDF	Inverse Document Frequency.
IEMOCAP	The Interactive Emotional Dyadic Motion Capture
ICWSM	International AAAI Conference on Weblogs and Social Media
SemEval	Semantic Evaluation

1. INTRODUCTION

1.1 Introducing the conversational sentiment analysis problem

Conversational sentiment analysis aims to capture the sentiment expressed throughout interactions between individuals. This involves developing models that can identify nuances like sarcasm, irony, and other complexities that are not apparent when analyzing single utterances in isolation. Additionally, it requires monitoring sentiment changes as conversations progress. However, accurately modeling conversational flow and sentiment dynamics is a complex challenge. Overcoming issues related to word meanings that change based on context, as well as accurately identifying ambiguity and sarcasm, is crucial in this task.

Conversations inherently involve self-dependencies and interpersonal dependencies [1]. Self-dependency refers to the relationship within an individual's own utterances, while interpersonal dependency concerns the interactions between different participants. Understanding self-dependency involves tracking an individual's previous statements and how their thoughts evolve during the conversation. Interpersonal dependency requires recognizing the unique speakers, their sentiments, and how one speaker's utterances influence the sentiments of others. To model these dependencies, techniques like long short-term memory networks [2], Recurrent Neural Network (RNN)s, transformers, and attention mechanisms [3] are used, all of which can maintain contextual information from prior utterances.

Table 4.1 provides an example of a spouse interaction. In this dialogue, the female exhibits accusatory behavior in response to her partner's perceived carelessness, resulting in rising aggression. Initially, the male spouse remains calm but eventually becomes emotionally affected by his wife's behavior, responding with anger. This scenario shows the female maintaining a consistent harsh attitude, illustrating self-dependency, while the

male's reaction demonstrates interpersonal dependency as he is influenced by his wife's demeanor.

Conversational sentiment analysis plays a critical role in understanding the emotional progression, contextual transitions, and participant interactions in conversations. Grasping the sentiment within conversations is vital for applications such as customer service evaluation, social media monitoring, market research, and opinion mining to inform decision-making. Despite numerous studies examining sentiment dynamics using traditional methods like word-level [4], sentence-level [5], or aspect-based [6] analyses, modeling interaction dynamics is still an emerging research area.

1.2 Highlighting the Lack of Conversational Data

Recently, several conversational datasets have been developed to support emotional or sentimental conversational models in English, including IEMOCAP [7], DailyDialog [8], MELD [9], Persuasion for Good [10], and ScenarioSA [11]. However, to our knowledge, there is no annotated dataset specifically designed for conversational sentiment analysis in the Turkish language.

To fill this gap, we compiled a Turkish conversational dataset, annotated with sentiment labels, specifically for research on conversational sentiment analysis. This dataset, named the "Couple Dialogue" dataset, includes 118 dyadic couple conversations, totaling 14,294 individual utterances. These conversations were gathered from heterosexual couples in Turkey, with speakers identified as "F" and "M" in each dialogue. Each utterance was manually annotated for sentiment polarity: "0" for negative, "1" for neutral, and "2" for positive sentiment. Due to privacy constraints, the dataset cannot be publicly released.

The dataset is notable for several reasons. First, it represents a pioneering effort in studying couple interactions within the Turkish context. Second, it is the most extensive collection of couple-generated utterances in Turkish, annotated with meticulous detail.

Table 1.1: *Example script from the Couple Dialogue dataset illustrating interactions between spouses, where "M" denotes Male and "F" denotes Female. Sentimental changes and shifts are indicated: "2" denotes positive, "0" denotes negative, and "1" denotes neutral.*

F:	Mesela spora çok gidiyorsun, gitme spora! [0] (For example, you go to the gym a lot, don't go to the gym!)
M:	Abartıyorsun! Haftada iki gün gidiyorum. [0] (You're exaggerating! I go two days a week.)
F:	Arkadaşların çağırdığı zaman gidiyorsun, ben dediğim zaman gelmiyorsun! [0] (You go when your friends call, you don't come when I say so!)
M:	Örnek ver. [1] (Give an example.)
F:	Düşünüyorum. [1] (I am thinking.)
M:	Mesela ne aradılar da gittim? [1] (For example, why did they ask me to go?)
F:	Düşünüyorum. [1] (I am thinking.)
M:	Mesela ne zaman aradılar da gittim? [1] (For example, why did they ask me to go?)
F:	Düşünüyorum. [1] (I am thinking.)
M:	Bulamadın değil mi? (laughing) [2] (You couldn't find it, could you?)
F:	Geçenlerde X aradı, çıktın gittin! [0] (Recently, X called, you left!)
M:	Arkadaşımla görüşmeye hakkım yok mu benim? [0] (Don't I have the right to meet my friend?)
F:	[0] (Recently, X called, you left!)
F:	Benden daha çok vakit geçirmeye hakkın yok! [0] (You have no right to spend more time than me!)
M:	Belki de bu tavırlarından dolayı uzaklaşmak istiyorum! [0] (Maybe I want to move away because of this attitude!)
F:	Hemen bahaneni buldun... [0] (You immediately found your excuse...)

Third, it is the first dataset specifically collected and annotated for research in conversational sentiment analysis. Fourth, it effectively captures the interaction dynamics between couples, showing emotional changes of each speaker throughout the dialogue. The dataset was carefully annotated using a rule-based approach to minimize subjectivity. Significant effort was invested in ensuring precise labeling, and an inter-annotator agreement study with statistical analyses validated the reliability of the sentiment annotations.

1.3 Introduction to the LLMs Utilized in This Study

LLMs such as Bidirectional Encoder Representations from Transformers (BERT) [12] and GPT-2 [13] have revolutionized natural language processing by enhancing the understanding of context and semantics in text. This progress continued with models like GPT-3 [14], GPT-4 [15], and the open-source Llama 2 [16], capable of generating human-like text and performing various tasks through zero- or few-shot learning [17]. Meta GenAI's Llama 2, trained on two trillion tokens from publicly available sources, includes models ranging from 7 billion to 70 billion parameters. American artificial intelligence (AI) research organization (OpenAI)'s GPT-3 has 175 billion parameters, and GPT-4, a multimodal model, likely exceeds 175 billion parameters, trained on trillions of tokens. This extensive training enables LLMs like Llama 2, GPT-3, and GPT-4 to excel at understanding context, subtle language nuances, and mimicking human dialogue patterns. As a result, these models can detect slight variations in tone and sentiment, making them highly effective for conversational sentiment analysis [18], [19], [20], [21], [22], providing valuable insights for businesses in customer service and market analysis.

LLMs can also utilize natural language prompts [23], [24] to expand their applications beyond fine-tuning for supervised tasks in transfer learning. Instead of gathering labeled training data for fine-tuning, users can direct the model to achieve specific outcomes, such as 'Classify the sentiment in these reviews as positive, negative, or neutral."

This study explores the potential of LLMs for sentiment analysis through a prompt-based approach using GPT-4 and the chat version of the Trendyol LLM model, available on Hugging Face. The Trendyol LLM model,[1] developed by the Trendyol group, is a generative model based on the Llama 2 7B architecture, fine-tuned on 180,000 instruction sets, primarily in English and Turkish. This research employed both instruction-based and few-shot prompting with GPT-4 and Llama 2. For GPT-4, only instruction-based prompting was used, while Llama 2 employed both instruction-based and few-shot prompting. The effectiveness of these methods was assessed using F1 scores to evaluate the models' performance in sentiment analysis.

Fine-tuning (FT) is a common method for adapting pre-trained language models to specialized tasks like dialogue systems [25]. It involves additional training on smaller, task-specific datasets to boost model performance in various applications. However, fine-tuning can sometimes diminish a model's generalization capabilities [26]. Anil et al. [27] suggest that incorporating multiple in-context examples during fine-tuning can help maintain text generalization. This study fine-tuned the Couple Dialogue dataset using models like BERT, GPT-3.5 Turbo, and Llama 2, with Llama 2 also undergoing few-shot fine-tuning for a comparative analysis of different fine-tuning methodologies.

LLMs create embeddings for each token, capturing both semantic and syntactic features. The embedding layer transforms numerical identifiers into dense, fixed-size vectors, providing rich linguistic information. This allows BERT and GPT models to understand intricate language patterns, enhancing their performance in various language tasks. In this study, tokenized words were processed through BERT and GPT models, with embeddings extracted from specific layers. The 'text-embedding-3-small" model,[2] was used

[1] https://huggingface.co/Trendyol/Trendyol-LLM-7b-chat-v0.1
[2] https://openai.com/blog/new-embedding-models-and-api-updates

for GPT embeddings, while pre-trained Turkish BERT and its fine-tuned version, utilizing the Couple Dialogue dataset, provided additional embeddings. GloVe embeddings (Global Vectors for Word Representation) [28] were also used, serving as inputs to train a DialogueRNN model for conversational sentiment classification. The DialogueRNN framework [29] encodes speaker-level information, considering factors like speaker identity, context, and sentiment from previous utterances. This method captures both self-dependency and interpersonal dependency, leading to improved context representation.

1.4 The introduction of Non-Contextual and Contextual Models

This study aims to demonstrate the effectiveness of context-based approaches in conversational sentiment analysis, specifically focusing on the 'Contextual Model." Experiments were conducted with various LLMs on the Couple Dialogue dataset. Our models fall into three categories: prompt-based (using GPT-4 and Llama 2), fine-tuned (employing BERT, Llama 2, and GPT-3.5 Turbo), and embedding-based (including DialogueRNN with embeddings from pre-trained Turkish BERT, fine-tuned Turkish BERT, text-embedding-3-small, and GloVe). This Contextual Model employs methods ranging from prompt-based instructions to leveraging fine-tuned datasets and embedding-based representations for deeper analysis. To establish a baseline for comparison, we developed a Non-Contextual Model" that processes utterances independently using various embeddings like BoW, TF-IDF, Word2Vec, and pre-trained Turkish BERT through traditional machine learning and deep learning techniques. This approach contrasts non-contextual processing with the nuanced capabilities of contextual models. We hypothesize that the Contextual Model will significantly outperform the Non-Contextual Model, and we will rigorously test this hypothesis through comparisons of F1 scores, providing a robust evaluation of both methodologies.

In experiments dedicated to the Non-Contextual Model, we evaluated classifier

performance by comparing BERT's contextual embedding with non-contextual word embeddings, including TF-IDF and BoW, as well as word embeddings with less contextual information, such as Word2Vec and GloVe. This comparison aimed to assess the Non-Contextual Model's effectiveness relative to the contextual embedding, using the same Couple Dialogue data, and it revealed significant performance differences.

Given the agglutinative nature and rich morphological structure of the Turkish language, we conducted a detailed morphological analysis specifically for the Non-Contextual Model experiments. This analysis included various word forms and negation morphemes, along with other morphemes that play a crucial role in sentiment representation. We compared classifier performances based on Weighted F1 scores. Additionally, experiments involving the inclusion and exclusion of stopwords and punctuation marks provided insights into their impact on sentiment labeling.

Finally, we developed a deep neural network architecture that balances complexity and performance. This architecture is less complex than the DialogueRNN model but more sophisticated than conventional classifiers. It incorporates BERT and GloVe embedding layers, offering a middle ground between the Non-Contextual and Contextual models.

1.5 The research objectives and research questions

This study has the following research objectives:

i) To improve sentiment analysis success rates by transitioning from the Non-Contextual Model, which serves as the baseline, to the Contextual Model using the Couple Dialogue data.

ii) To enhance sentiment analysis success rates by applying additional preprocessing and morphological analysis to the Non-Contextual Model using the same dataset.

To achieve these goals, the following research questions are addressed:

1. Does the Contextual Model consistently outperform the Non-Contextual Model on the Couple Dialogue data?

2. What is the most efficient prompt structure that would be understandable for both Llama 2 and GPT-3.5 Turbo in the context of conversational sentiment analysis on the Couple Dialogue data? (This question arises because, to the best of our knowledge, a suitable prompt structure for this task has not been documented in the literature.)

3. Can our conversational sentiment analysis task, which is based on the Turkish language, be effectively achieved by Llama 2 and GPT-3.5 Turbo, both of which are primarily pre-trained in English?

4. How much improvement do fine-tuned models offer over prompt-based approaches for this conversational sentiment analysis task?

5. Do contextual embeddings outperform less sophisticated vector representations for the Couple Dialogue data?

6. What are the effects of preprocessing and morphological analysis on a language with rich morphology, like Turkish, in enhancing sentiment labeling success rates?

7. Does the use of a more complex neural network architecture in the Non-Contextual Model help close the performance gap with the Contextual model?

1.6 The contributions of the study

The key contributions of the research outlined in this paper are as follows:

• We curated a dataset of couple interactions called the 'Couple Dialogue" dataset, designed specifically for conducting conversational sentiment analysis in Turkish. Along

with this dataset, we provided statistical analysis and demonstrated the reliability of the data collection and agreement procedures.

• In this study, we developed and compared two types of frameworks: Non-Contextual, which serves as our baseline, and Contextual. We introduced a novel approach by employing contextual modeling techniques through instruction-based models, fine-tuned models, and embedding-based models, including the DialogueRNN architecture, specifically tailored for a Turkish dataset designed for conversational sentiment analysis. Our preliminary results indicate that the Contextual Model significantly outperforms the Non-Contextual Model, achieving statistically significant improvements of 9.98% in Weighted F1 scores with conventional machine learning classifiers and 6.51% with deep neural network architectures, as confirmed by paired t-tests.

• We demonstrated that the fine-tuned approach significantly outperformed the prompt-based approach, achieving a notable improvement of up to 31.12% in Weighted F1 score for our conversational sentiment analysis task. This enhancement was confirmed to be statistically significant through a paired t-test.

• Our analysis revealed that incorporating specific morphemes significantly enhances sentiment classification success in Turkish, evidenced by improvements of up to 9.89% in Weighted F1 scores. A majority of our tested pairs showed statistically significant enhancements, validating the effectiveness of these morphological features. We conducted selective paired t-tests across different classifiers and preprocessing settings, which supported the positive impact in most cases, though not universally. This underscores the overall efficacy of integrating morphemes in sentiment analysis while highlighting variability in results depending on the preprocessing context.

• In our study, we illustrated that BERT-based utterance features consistently outperform

other used vector embedding models within both Non-Contextual and Contextual frameworks. In the Contextual Model, the improvement was particularly notable, with a significant increase of 14.46% in Weighted F1 scores, as confirmed through statistical testing. In the Non-Contextual Model, BERT achieved an improvement of up to 7.42% and demonstrated statistical significance against all competing embeddings except for GloVe, where the difference did not reach statistical significance.

2. RELATED WORK

2.1 Early Studies in Sentiment Analysis

Sentiment analysis has remained a key area of focus within NLP research. Initial efforts in this domain were directed toward various levels, such as sentence-level [30], document-level [31], aspect-level [32], and phrase-level analysis [33]. In addition, several researchers explored tasks related to sentiment analysis, including multimodal sentiment analysis [34], domain adaptation [35], sarcasm detection [36], [37], and the identification of biases within sentiment analysis frameworks [38], [39]. In recent years, neural architectures introduced in studies like [40], [41], [42], and [43] have largely surpassed traditional machine learning-based algorithms.

2.2 Studies in Conversational Sentiment Analysis

The analysis of sentiment in conversational data has gained significant traction in recent years. Poria et al. [44] introduced a contextual LSTM architecture that processes a sequence of utterances from videos to determine sentiment. Hazarika et al. [45] employed Gated Recurrent Unit (GRU)s to incorporate audio, visual, and textual features, modeling conversational histories to detect utterance-level emotions in dyadic videos. However, these studies primarily focus on dyadic interactions. To address emotional dependencies both within and between speakers, Hazarika et al. [46] proposed the Interactive Conversational Memory Network (ICON), which integrates multimodal features—including language, visual, and audio data—within a memory framework. This model laid the groundwork for analyzing sentiment in multi-party conversational settings.

Majumder et al. [29] introduced the DialogueRNN framework, which is designed to scale beyond dyadic conversations to support multiple speakers. This model employs three types of GRUs: the global GRU adjusts the contextual state, the party GRU updates the state of the speaker or listener, and the emotion GRU is responsible for emotion

classification. To extract textual features, the authors utilized Convolutional Neural Network (CNN)s.

Zhong et al. [47] integrated both contextual information and external knowledge into textual conversations and proposed two architectural solutions: hierarchical self-attention and cross-attention mechanisms to identify emotions. Similarly, Ghosal et al. [48] developed the Dialogue Graph Convolutional Network (DialogueGCN), which aims to capture participant interactions within a conversation. Lian et al. [49] introduced the Conversational Transformer Network (CTNet), which combines a bi-directional GRU layer with a multi-head attention mechanism to emphasize key utterances while processing multimodal features.

Jiao et al. [50] designed a network consisting of a Hierarchical Memory Network (HMN) and an Attention GRU (AGHMN). The HMN component enhances utterance features and models interactions across previous utterances, while the Attention GRU focuses on improving memory summarization and providing a richer contextual understanding. Zhang et al. [51] proposed an interactive LSTM framework to assess understandability, credibility, and influence among speakers in conversational sentiment analysis. Additionally, they developed and released a high-quality English conversational dataset, ScenarioSA, to address the lack of publicly available conversational datasets.

Zhang et al. [52] introduced a novel quantumlike multimodal network (QMN) approach, which integrates quantum theory with long short-term memory (LSTM) networks to capture intra- and inter-utterance dynamics using both text and image features. Ghosal et al. [53] extended existing LSTM and DialogueRNN models by incorporating pre-trained embeddings, such as GloVe and Roberta [54], for five dialogue understanding tasks. In our work, we conducted experiments on the Couple Dialogue dataset using DialogueRNN with multiple embeddings, including pre-trained Turkish BERT, fine-tuned Turkish BERT, text-embedding-3-small, and GloVe embeddings, and we presented the

corresponding results.

Shou et al. [55] proposed the DSAGCN model for conversational affective analysis, which leverages dependent syntactic analysis and a graph CNN architecture. The model achieved superior accuracy and F1 scores for happiness and anger emotions on the IEMOCAP [7] and MELD [9] datasets. Similarly, Wei et al. [56] developed a model that integrates context information from the current utterance and employs a two-channel feature extractor (TFE) for emotion detection. Their efficiency analysis demonstrated that the model outperformed baselines like DialogueRNN [29] and DialogueGCN [48] in convergence speed, trainable parameter count, and training time. Tu et al. [57] introduced Sentic GAT, a context- and sentiment-aware framework that applies hierarchical multi-head attention (HMAT) to identify relationships among concepts and models dependencies between the current utterance and external contextual concepts through sentiment-aware graph attention (CSAGAT).

2.3 LLMs in General and Contextual Sentiment Analysis

Krugman et al. [18] conducted an extensive study evaluating the effectiveness of LLMs, including GPT-3.5, GPT-4, and Llama 2, in sentiment analysis—a critical aspect of marketing research aimed at interpreting consumer sentiments. The study found that, despite relying on zero-shot learning, LLMs can achieve accuracy levels comparable to or exceeding those of conventional transfer learning models. Their performance, however, is influenced by factors such as the complexity of the text and its source. Moreover, the research emphasizes the superior explainability provided by LLMs, particularly Llama 2, offering valuable guidance for model selection in the generative AI landscape. It is important to note that this study focused on binary and three-class sentiment analysis, which differs from tasks involving conversational sentiment analysis.

Buscemi et al. [19] evaluated the performance of prominent LLMs, including

ChatGPT, Gemini, and LLaMA2, in processing ambiguous and subtle texts spanning ten different languages. The results showed that models such as ChatGPT and Gemini are generally effective at interpreting ambiguous scenarios; however, they exhibit considerable biases and inconsistent outcomes that vary depending on the language and specific model. These findings, which were validated using human benchmarks, underscore the need for ongoing improvements to LLM algorithms and training datasets to enhance their accuracy, interpretability, and applicability across diverse cultural and linguistic settings. The analysis relied on responses generated through carefully crafted prompts.

Zhang et al. [20] tackled the challenges faced by LLMs in financial sentiment analysis, specifically their difficulties in handling numerical sensitivity and contextual comprehension. They proposed an instruction-tuned model named Instruct-FinGPT, which builds on the pre-trained knowledge of LLaMA and enhances its performance through instruction tuning tailored for financial data. The model achieved superior results compared to both general-purpose LLMs and advanced supervised models, even when trained on minimal instruction data. This innovative strategy underscores the importance of numerical sensitivity and contextual awareness in financial NLP, providing a promising avenue for improving sentiment interpretation within financial domains.

Carvalho et al. [21] explored sentiment analysis applied to dialogues within Portuguese customer-support interactions, assessing the impact of incorporating various levels of contextual information, including the number and speaker of preceding utterances. By experimenting with a range of classification techniques from traditional methods to advanced models like fine-tuned BERT and Few-Shot Learning with GPT-3 and OPT, their findings highlight the variable significance of context in dialogue-based sentiment analysis. Context proves particularly critical in scenarios involving human-computer interaction and brief human-human exchanges focused on the client, thereby enhancing both the performance and the relevance of the dialogue systems in practical settings. This

study represents one of the relatively rare examples of using LLMs for conversational sentiment analysis, emphasizing the importance of contextual information. In contrast to this study, our approach involves obtaining the entire context from the current conversation to provide a richer context, enabling the corresponding model to learn from previous interactions. We applied this comprehensive context logic across embedding-based, prompt-based, and fine-tuning-based approaches.

Kheiri et al. [22] performed an extensive analysis of Generative Pretrained Transformer (GPT) methods for sentiment analysis, utilizing the SemEval 2017 - Task 4 dataset. They compared three distinct techniques: prompt engineering with GPT-3.5 Turbo, fine-tuning GPT models, and a novel embedding-based classification strategy. The results demonstrated that GPT-based approaches outperformed state-of-the-art models by over 22% in F1 score, showcasing their ability to address challenging aspects of sentiment analysis, such as sarcasm detection and contextual interpretation. This study highlights the superior performance of GPT models while providing valuable insights into their application for automated sentiment assessment. In our work, we employed models similar to those used by Kheiri et al.; however, their analysis was limited to GPT-based models performing three-class sentiment classification on a Twitter dataset, whereas our focus is on conversational sentiment analysis.

2.4 Studies in other Morphologically Rich Languages

Several studies have been conducted on sentiment classification tasks in other morphologically rich languages. Abdul-Mageed, Diab et al. [58] applied inflectional and derivational suffixes to Arabic sentiment classification tasks, achieving notable improvements in F1 scores when combined with Support Vector Machine (SVM) algorithms, compared to using stems alone. Joshi, Bhattacharyya, and Balamurali [59] introduced a three-stage framework for sentiment extraction from Hindi reviews, utilizing supervised

algorithms on annotated corpora, machine translation to English in the absence of corpora, and sentiment lexicons for classification based on majority voting. Yang and Chao [60] focused on Chinese movie reviews, effectively leveraging seed words and mono-syllabic morphemes while incorporating specific suffixes as features, even in the absence of sentiment resources. Medagoda [61] explored machine learning methods for sentiment classification in Sinhala, analyzing how the selection of appropriate adjectives and adverbs influences classification outcomes and generating polarity scores using various weighting schemes. Additionally, Medagoda [62] proposed multiple approaches for creating sentiment lexicons for Sinhala, such as cross-lingual strategies, morphological feature integration, and a graph-based method. Their work combined rule-based approaches with negations and intensifiers and claimed applicability to other morphologically rich languages.

2.5 Advances in Multimodal and Neural Frameworks for Emotion and Sentiment Analysis

Zhang et al. [63] proposed a multi-task learning framework with an attention mechanism for semantic and instance segmentation in coastal urban spatial perception. This study highlights how attention mechanisms can improve model performance in complex tasks by effectively managing various subtasks simultaneously. The framework's ability to handle multiple tasks could inspire advancements in conversational sentiment analysis by integrating attention mechanisms to focus on different aspects of dialogue. In our work on conversational sentiment analysis for Couple Dialogue data, we utilized the DialogueRNN model within our embedding-based approach, which belongs to our Contextual Model framework. DialogueRNN uses an attention mechanism to dynamically focus on relevant parts of the conversation history, allowing the model to better understand the context and dependencies between utterances. This is achieved by computing

attention scores for each past utterance, normalizing them to obtain attention weights, and creating a context vector that influences sentiment or emotion prediction. By effectively managing the conversation's context, this mechanism enhances the model's ability to predict sentiments/emotions accurately.

In the realm of cross-subject emotion recognition, Xiaopeng Si and colleagues [64] developed a novel brain-computer interface using functional near-infrared spectroscopy (fNIRS) and a dual-branch joint network (DBJNet). The study introduced deep learning technology to fNIRS for the first time, demonstrating a significant improvement in emotion decoding accuracy. This work is notable for its exploration of cross-subject emotion recognition, showcasing how neural activity data can be leveraged for more accurate emotion classification, a concept potentially applicable to understanding neural correlates in conversational sentiment analysis.

Zhu's paper [65] focuses on developing an intelligent assistant system equipped with emotion recognition capabilities to enhance emotional intelligence and provide personalized services. The system uses multimodal data, including speech, text, and facial expressions, to accurately identify users' emotional states and tailor services accordingly. By analyzing user behavior and emotions, the assistant can offer more relevant and satisfying interactions, improving overall user experience. This approach emphasizes the integration of emotional intelligence in real-time sentiment analysis applications, enhancing the system's ability to respond empathetically and effectively to user needs.

Finally, the SiMaLSTM-SNP model by Gu et al. [66] combines Siamese networks and membrane computing for semantic relatedness learning. This work demonstrates that SiMaLSTM-SNP, which integrates word embeddings, multi-head self-attention, and LSTM-SNP, outperforms several classical and modern approaches in capturing the semantic distinctions between sentences. This innovative approach provides insights into advanced neural architectures that could be adapted to enhance conversational sentiment

analysis models by improving the understanding of semantic relationships within dialogues.

2.6 Studies in Turkish Sentiment Analysis

2.6.1 The Impact of Agglutinative Nature of Turkish on Sentiment Analysis

Turkish, being an agglutinative language, offers both unique opportunities and challenges for sentiment analysis due to its complex morphological structure. Recognizing these linguistic features is essential for assessing the development and effectiveness of sentiment analysis systems tailored to Turkish.

2.6.1.1 Advantages- Rich Morphological Information

The agglutinative structure of Turkish provides a wealth of morphological information, which enables a more nuanced sentiment analysis. For instance, the word "seviyorum" (I love) combines the root "sev" (love) with the suffix "-iyorum," which conveys the present continuous tense and first-person singular. As highlighted in studies like [67], this morphological richness helps sentiment analysis systems capture subtle sentiment variations, improving overall accuracy. Moreover, suffixes in Turkish can convey context, relationships, and emotional tones such as respect or sarcasm, which are vital for precise sentiment detection [68].

2.6.1.2 Challenges- Complex Morphological Parsing and Data Sparsity

On the other hand, this morphological complexity also presents challenges. The frequent use of suffixes leads to intricate word structures, which makes morphological parsing both challenging and computationally demanding [69]. Furthermore, the vast number of word forms creates data sparsity, making it harder to train models effectively and increasing the likelihood of overfitting [70]. The resulting large vocabulary size and the associated high out-of-vocabulary (OOV) rate further complicate statistical language

modeling, requiring advanced techniques to overcome these obstacles [71].

These linguistic features have been studied and utilized in various early works focused on Turkish sentiment analysis.

2.6.2 Early Studies in Turkish Sentiment Analysis

Several recent studies have explored Turkish sentiment analysis. Aydın et al. [67] performed morphological analysis and examined various word forms in the Turkish language to address the binary sentiment classification challenge. Their framework included supervised, unsupervised, and semi-supervised methods, as well as a hybrid approach combining these strategies. In addition to using the surface and root forms of words as features, they identified specific morphemes that conveyed more sentimental significance, leading to the generation of a third form known as the partial surface form. Their method was applied to datasets consisting of Turkish movie reviews and Twitter posts, achieving notable improvements over baseline models. Similarly, we constructed a partial surface form that retained the suffixes analyzed in their work whenever they appeared in a given word's surface form.

Omurca et al. [5] compiled hotel reviews from an online source and developed a semi-automatically annotated corpus at the sentence level, which could be utilized in aspect-based sentiment analysis tasks for the Turkish language. Dehkharghani et al. [68] proposed a sentiment analysis system that addressed sentence-level, aspect-level, and document-level tasks by incorporating a wide range of linguistic features. Their system achieved accuracy scores between 60% and 79%. In the work of Güler et al. [72], the researchers created eight distinct sets of word embeddings using Word2vec and FastText, specifically focusing on different morphological forms of words. The effectiveness of these embeddings was demonstrated through comparative analyses across tasks such as word analogy, sentiment analysis, text classification, and language modeling.

2.6.3 Transformer-based Studies in Turkish Sentiment Analysis

Köksal et al. [73] presented the BounTi dataset, which consists of Turkish tweets focused on various universities in Turkey. The dataset was manually annotated for the Sentiment Analysis task, and the authors evaluated the performance of Turkish transformer models such as MBERT, XLM-Roberta, and BERTurk on this dataset. In the study by Ozturk et al. [74], transformer-based models, including BERT, ConvBERT, and ELECTRA, were applied alongside traditional deep learning and machine learning methods on four different Turkish datasets, and their performance results were systematically compared.

Acikalin et al. [75] explored binary classification problems using datasets containing Turkish movie and hotel reviews. Their approach involved employing both a multilingual version of BERT and the primary BERT model after translating Turkish texts into English, enabling a comparison of the achieved results. Guven et al. [76] focused on the accuracy of pre-trained models such as BERT, ALBERT, ELECTRA, and DistilBERT for sentiment classification tasks using datasets comprising Turkish hotel reviews and movie reviews. Similarly, Mutlu et al. [77] introduced a manually labeled Turkish Twitter dataset tailored for targeted sentiment analysis tasks. They experimented with multiple BERT-based model variants and achieved significant performance improvements over baseline results.

In more recent work, Yildirim et al. [78] conducted groundbreaking research by fine-tuning the BERT-based model, BERTurk, for four essential Natural Language Understanding (NLU) tasks: Named-Entity Recognition, Sentiment Analysis, Question Answering, and Text Classification. The fine-tuned model, designed specifically for the Turkish language, achieved superior results over existing baselines when evaluated using the Turkish Benchmark dataset. Additionally, Kaya et al. [79] introduced ITUTurkBERT,

a newly developed set of BERT models fine-tuned for Turkish, which is a morphologically rich language. Their study investigated the influence of various tokenization strategies, with a particular focus on vocabulary size and morphological tagging, to determine their effects on NLP tasks such as Named Entity Recognition, Sentiment Analysis, and Question Answering. While their findings revealed that larger vocabulary sizes generally improve model performance, sentiment analysis outcomes did not align with this trend, indicating the need for further investigation in this domain.

2.6.4 Studies in Turkish Sentiment Analysis Based on Interaction Dataset

Toh et al. [80] proposed a hybrid model aimed at improving the understanding of positive and negative emotions in human-robot interaction tasks. They developed a unique dataset consisting of 13,500 question pairs, which were transformed into vector representations using pre-trained Turkish BERT, Word2vec, and TF-IDF techniques. By integrating polarity vectors, which reflect the scores of negative and positive words in the question-answer text vectors, the model achieved a significant boost in accuracy when compared to earlier models. In a separate study, Söğüt et al. [81] designed a chatbot for promoting credit card services to customers. They employed a Finite State Machine (FSM) structure tailored to the Turkish language, where the sentiment classification model's results controlled the transitions between predefined states. The dataset they compiled contained 500 utterances, which were annotated and used to compute sentiment classification results across various algorithms.

In summary, conversational sentiment analysis research is making notable progress, incorporating a variety of datasets, including both single-modal and multimodal approaches, and effectively leveraging feature engineering techniques alongside deep learning architectures. However, there remains a need for further exploration, particularly for the Turkish language. To the best of our knowledge, this study represents the first attempt to apply

sentiment analysis to conversational Turkish text using the Transformer model. Moreover, the dataset we created and annotated is the first of its kind, focusing specifically on Turkish couple interactions. No prior research has compiled or annotated dialogue utterances for conversational sentiment analysis in Turkish.

The work of Söğüt et al. [81] includes 500 Turkish utterances, where sentiment labeling was performed using utterance-based features, such as counting the number of positive and negative terms within a sentence. In another notable study, Kılıç et al. [82] created an annotated corpus of human-machine dialogues containing 426 conversations and 5020 sentences, which were used to classify dialogue acts in Turkish within the banking domain. As illustrated in Table 3.1, our study utilized a considerably larger dataset, comprising 118 dialogues and a total of 14,294 annotated utterances.

Additionally, this research marks the first effort to capture contextual dependency to achieve utterance-level comprehension for sentiment classification in Turkish text. While aspect-level sentiment analysis [6], document-level sentiment analysis [31], and sentence-level sentiment analysis [5] have been extensively explored, the development of a dialogue-based sentiment detection framework in Turkish remains unexplored until now.

2.6.5 Advancements in Turkish NLP Using LLMs

Uludogan et al. [83] introduced TURNA, which is the first unified language model for Turkish, designed to handle both NLU and Natural Language Generation (NLG) tasks. TURNA was developed using a diverse corpus comprising web data, parliamentary transcripts, and scientific articles, amounting to a total of 43 billion tokens. Utilizing an encoder-decoder architecture and boasting 1.1 billion parameters, TURNA achieved superior performance when evaluated across thirteen datasets and eight tasks. The results showed that TURNA matched or surpassed the Turkish-specific BERTurk in NLU tasks

and outperformed multilingual models like mBART and mT5 in classification tasks such as Product Reviews21, TTC490022, and Tweet Sentiments.

Turker et al. [84] introduced VBART, the first sequence-to-sequence Large Language Model (LLM) specifically for Turkish. Pre-trained on an extensive 135.7 GB corpus, VBART incorporates components from BART and mBART architectures. Available in Large and XLarge configurations, it sets new benchmarks in Turkish NLP tasks, including abstractive text summarization, text paraphrasing, title generation, question answering, and question generation, outperforming previous models significantly.

TurkishBERTweet [85] is the first large-scale pre-trained language model developed specifically for Turkish social media analysis, utilizing a corpus of nearly 900 million tweets. Optimized for shorter input lengths to improve inference speed, TurkishBERTweet outperformed both multilingual models and Turkish-specific models like BERTurk in sentiment classification and hate speech detection tasks. It also stands out as a cost-efficient and computationally effective solution for processing large-scale datasets. Additionally, the model and associated LoRA adapters have been released under the MIT License to facilitate further research and application in Turkish NLP and social media analytics.

Cam et al. [86] assessed the performance of ChatGPT and fine-tuned BERT-based models in detecting Turkish hate speech. The results showed that ChatGPT performed on par with these models, highlighting its viability for such tasks. Denghan et al. [87] further examined ChatGPT's capability for hate speech detection in Turkish tweets using the SIU2023-NST dataset. While ChatGPT achieved an accuracy of 65.81% in a few-shot setting, it lagged behind a supervised fine-tuned BERT model, which achieved 82.22%, demonstrating that BERT remains better suited for natural language classification tasks despite its smaller size.

Research specifically targeting Turkish sentiment analysis remains limited. This

study introduces key innovations, including the first exploration of a contextual framework for conversational sentiment analysis that accounts for speaker dependency, previous dialogue context, and sentiment dynamics. While Uludogan et al., Turker et al., and TurkishBERTweet focused on creating Turkish-specific LLMs, our work uniquely compares the effectiveness of fine-tuning and prompt-based approaches. To this end, we employed state-of-the-art LLM models such as GPT-4, GPT-3.5 Turbo, and Llama 2, using a Turkish dataset specifically curated for conversational sentiment analysis.

3. CORPUS DESCRIPTION

In this chapter, we introduce the Turkish conversation dataset along with sentiment labels.

3.1 Participants and Procedures

The data were collected from a sample of 103 newly married heterosexual couples in Turkey. All participants were in their first marriages and had no children. The duration of marriage ranged between one and 15 months, with an average of 6.06 months (SD = 3.43). Participants' ages spanned from 20 to 48 years, with a mean age of 27.83 (SD = 3.80). The participants' average educational attainment was 16.36 years, with individual durations ranging from 8 to 26 years.

For recruitment, flyers were distributed across various universities, municipalities, and organizations located in an urban area. Recruitment messages were also shared on social media platforms and through diverse email groups. Individuals expressing interest were contacted via phone or email to provide them with detailed information about the study. Upon agreeing to take part, participants received an email containing their unique ID number and a link to access the online survey. Before data collection began, participants were informed about the study's objectives and methods, and consent was obtained.

3.2 Audio-visual Data

The audio-visual data were gathered in the Relationship Research Laboratory at Özyegin University, located in Istanbul, Turkey. The recordings took place in rooms equipped with video cameras. The data collection procedure adhered to the protocols described in the Couples Interaction Rating System (CIRS), developed by Heavey, Gill, and Christensen [88]. CIRS is a structured observational coding system designed to analyze marital interaction dynamics. It evaluates key aspects such as communication patterns,

effectiveness of conflict resolution, and emotional expressions during discussions, providing a systematic framework for examining relational behaviors.

When couples arrived at the lab, each spouse was asked to choose a topic they considered important in their relationship where they held differing viewpoints. After both spouses independently selected their topics and mutually agreed upon each other's choices, the couple participated in two 10-minute discussions, with each session focusing on one spouse's selected topic. At the 10-minute mark, a research assistant knocked on the door to signal the couple to switch to the next topic, chosen by the other partner. Once the second 10-minute discussion was completed, the assistant knocked again to indicate the end of the session. The resulting recordings, where couples discussed their problems over two 10-minute intervals, constituted the audio-visual data used in this study.

3.3 Couple Dialogue Dataset

Our goal for this work is to conduct sentiment analysis on the collected data outlined above. Initially, we transcribed each spoken corpus into written text in the Turkish language. To ease and accelerate the annotation process, we randomly selected 118 interactions out of the possible 206 interactions (two conversations for each of the 103 couples), resulting in 118 separate conversation texts. Subsequently, we annotated the sentiment and emotional labels for each utterance within every conversation, resulting in the generation of the Couple Dialogue dataset. In contrast to the approach in [51], we chose not to annotate the entirety of the conversation in this study; our focus lies solely on the utterance-level analysis.

Since we transcribed texts from the spoken corpus, many of these conversations could be characterized as informal, resulting in a limited vocabulary of 4656 unique words. The detailed statistics are available in Table 3.1.

As seen in Figure 3.1, we observe a smaller sample size for positive samples,

Table 3.1: *Statistical Overview of the Couple Dialogue Dataset: This table summarizes key statistics from the Couple Dialogue dataset, detailing the composition and structure of conversations and utterances, including word counts and gender distribution.*

Statistics	**Value**
The total number of conversations	118
The total number of utterances	14,294
The total number of words in all corpus	142,306
The total number of unique words in all corpus	4,656
The total number of male utterances	6,798
The total number of female utterances	7,496
Average words per conversation	1,205.9
Average words per utterance	9.95

totaling 2083, rendering our Couple Dialogue Dataset imbalanced. Given that the couples engaged in discussions to resolve disagreements within their marriage, this disparity in sample sizes can be considered reasonable. In contrast, the number of negative and neutral samples is comparable. Specifically, we have 6161 negative samples, whereas the total number of neutral samples amounts to 6050.

The distribution of sentiment labels regarding gender diversity is shown in Figure 3.2 as percentages, indicating that female spouses developed a more negative attitude during conversations compared to male spouses. While the percentage of females demonstrating a negative attitude was 25.0%, this value was 18.1% for male spouses. The percentage of positive sample values remained at 6.6% for the female group, while male spouses had 8.0%. Both these statistics demonstrate that female spouses exhibited a greater tendency towards negativity in discussions aimed at resolving problems compared to their male counterparts.

3.4 Data Preprocessing in Data Collection

The speech data were directly converted into text, producing transcriptions that contained numerous illegible characters and frequent misspellings. One major challenge encountered during this process was the absence of sentence separation due to missing

Figure 3.1: *Distribution of positive, negative, and neutral sentiments in the Couple Dialogue dataset, illustrating the count of each sentiment type.*

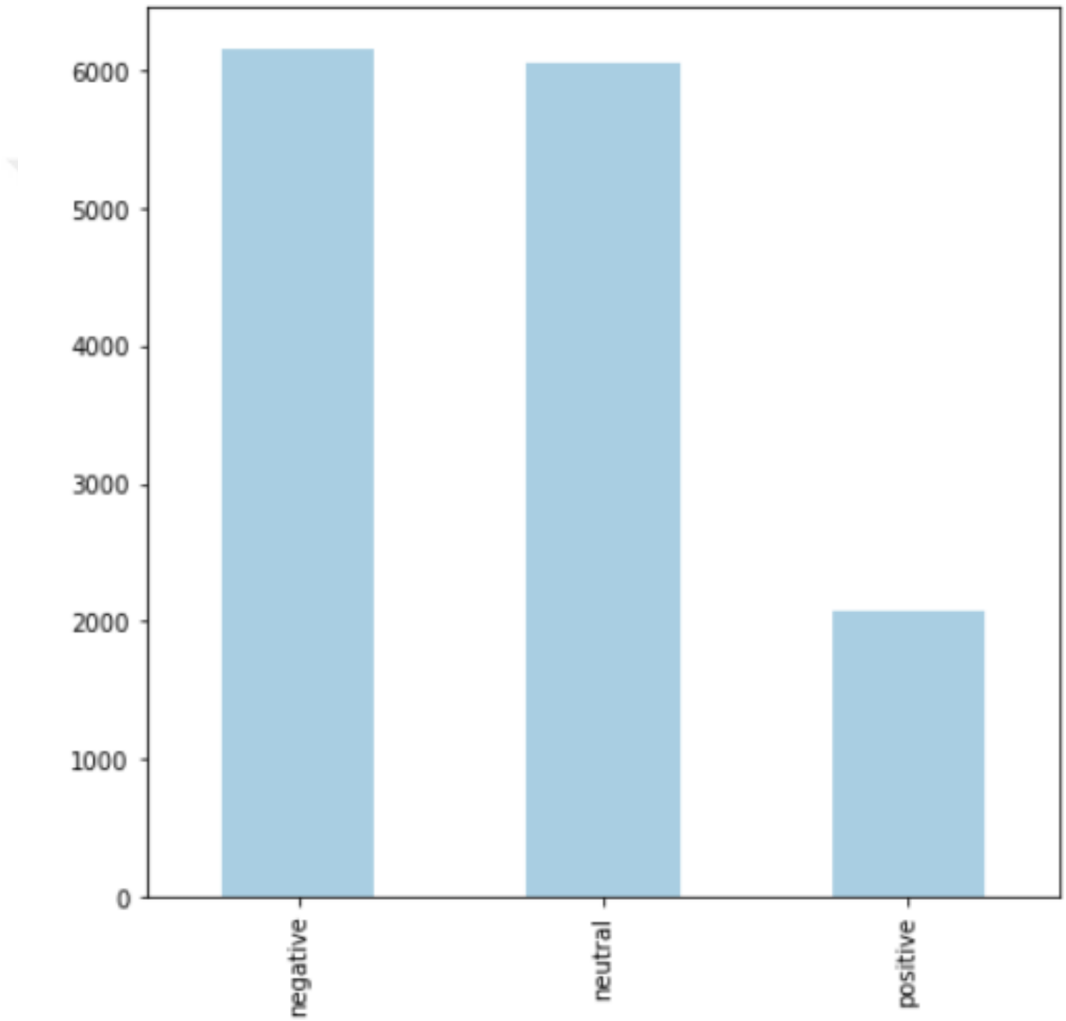
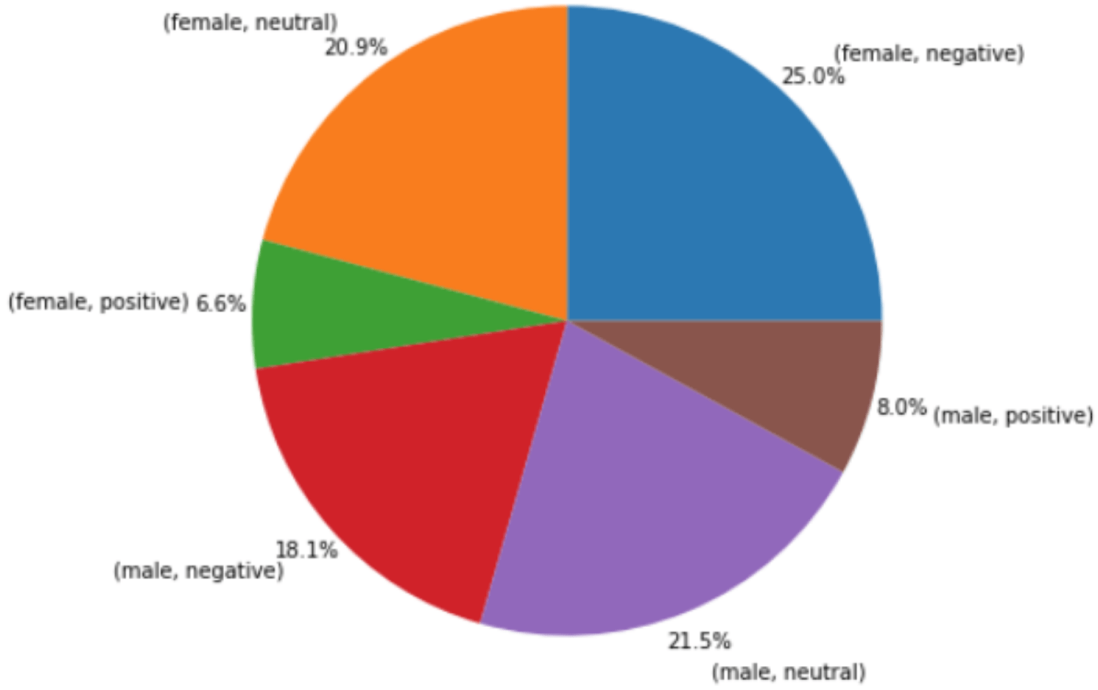


Figure 3.2: *Distribution of positive, negative, and neutral sentiment labels across male and female speakers in the Couple Dialogue dataset.*



punctuation marks in the transcribed text, which made the resulting content complex and difficult to interpret. To improve the clarity and readability of the sentences, considerable effort was invested in manually inserting appropriate punctuation marks and correcting spelling errors. Since no tool was available that could meaningfully segment the unstructured Turkish text with punctuation, this task had to be performed manually.

Beyond correcting punctuation, a custom autocorrect tool was developed to handle spelling errors in the corpus. Initially, the Microsoft Office Spell Checker was integrated to automate part of the correction process. However, this automated phase alone proved insufficient, requiring further manual inspection and correction of remaining misspellings. As a result, an exhaustive manual review was undertaken to ensure all errors were accurately rectified. This two-step correction process—starting with automated adjustments and followed by thorough manual refinements—was crucial for generating high-quality, coherent, and readable text suitable for our analysis.

3.5 Annotation Procedure

During the preprocessing stage, we manually annotated our dataset for both sentiment and emotional expressions. While the primary emphasis of this paper lies on sentiment analysis, the emotional labels played a crucial role in providing contextual understanding for each interaction, thus enabling more accurate sentiment classification. Initially, each utterance was tagged with one of seven emotional categories: 0 representing anger", 1 for no emotion", 2 for disgust", 3 for fear", 4 for happiness", 5 for sadness", and 6 for surprise." Following the assignment of emotional labels, each utterance was further annotated with one of three sentiment labels: 0 for negative", 1 for neutral", and 2 for positive."

This dual-label annotation approach facilitated a more efficient sentiment classification process. For example, utterances categorized as "anger" were generally associated

with negative sentiment. By first identifying the emotional context, annotators were able to assign sentiment labels with greater accuracy and consistency. A comprehensive analysis of these emotional annotations is planned for future research, as described in the Future Work section.

To establish a foundation for sentiment and emotion labeling, we initially annotated the dataset ourselves. These annotations served as the ground truth labels, which were consistently utilized in all subsequent experiments in this paper to ensure reliability and consistency throughout the analysis.

To further enhance the dependability of our ground truth labels, we involved four experienced volunteers familiar with sentiment analysis. Their role was to validate and refine the existing labels rather than directly contributing new annotations for use in the experiments. The feedback from these annotators was specifically used to assess the quality and reliability of our initial ground truth labels.

To ensure consistency and minimize subjectivity among the annotators, we developed a comprehensive, rule-based guideline document grounded in the principles set forth by the Paul Ekman Group [89]. This document provided explicit criteria for recognizing and categorizing sentiments and emotions based on textual cues. While Paul Ekman's work is renowned for identifying universal emotions through facial expressions and body language, our guidelines adapt these concepts to the conversational context by focusing on the language that typically reflects these emotions—*anger, disgust, fear, happiness, sadness,* and *surprise*—through specific lexical choices, tone, and contextual analysis. The detailed explanations for each emotion are given below:

- **Anger** is annotated when utterances include strong, confrontational words or phrases that suggest frustration or aggression. For example, "Why do you always mess this up?" could indicate anger, especially when combined with exclamation points or harsh adjectives.

- **Happiness** is identified through the use of positive expressions, exclamation marks, and affirming language that typically conveys joy or satisfaction. For example, "That's fantastic! I'm thrilled to hear that."

- **Sadness** involves words or phrases that express loss, longing, or disappointment. Phrases like "I really miss those days" or "I wish things were different" often convey sadness.

- **Surprise** is indicated by expressions that show astonishment or unexpected reactions, often marked by exclamatory phrases such as "Wow! I wasn't expecting that!" or "How did that happen?"

- **Fear** can be deduced from expressions that reflect anxiety, worry, or concern. For instance, "What if it doesn't work out?" or "I'm worried about this" suggests fear.

- **Disgust** is marked by words that express strong aversion or disapproval, such as "That's horrible!" or "I can't stand this."

- **No Emotion** is identified when the text is factual or neutral, lacking any emotional words or phrases. An example might be "The meeting is scheduled for 3 PM."

3.6 Training and Calibration of Annotators

During the annotation process, annotators initially assigned one of the seven emotional states to each utterance. Sentiment labels were then assigned in separate sessions to minimize the risk of immediate decision-making bias. The annotators participated in a comprehensive training program that involved examining case studies and engaging in trial scoring sessions. These sessions allowed them to practice and refine their ability to apply the labeling guidelines to sample utterances. The purpose of this training was to ensure consistency in understanding and implementing the labeling criteria.

The guidelines provided clear and detailed instructions for linking emotional expressions to their corresponding sentiment categories. To assist annotators further, we also included a set of general rules presented as pseudocode, as shown below:

```
function detect_sentiment(emotion, context):
    if emotion == "No Emotion":
        return "Sentiment: Neutral"

    elif emotion == "Happiness":
        return "Sentiment: Positive"

    elif emotion == "Surprise":
        # Context Check
        if context == "positive outcome \
        (e.g., a surprise party)":
            return "Sentiment: Positive"
        elif context suggests concern for safety:
            return "Sentiment: Negative"
        else:
            return "Sentiment: Neutral"

    elif emotion in ["Anger", "Disgust"]:
        return "Sentiment: Negative"

    elif emotion == "Sadness":
        # Context Check
        if context suggests a resigned or \
        matter-of-fact discussion:
            return "Sentiment: Neutral"
        else:
            return "Sentiment: Negative"

    elif emotion == "Fear":
        # Context Check
        if context == "immediate threat or loss":
            return "Sentiment: Negative"
        elif context == "mild or cautionary \
        without immediate threat":
```

```
            return "Sentiment: Neutral"

    else:
        return "Sentiment: Neutral"
```

Our annotation guidelines provide explicit categorization rules while also addressing the complexities of human emotions, particularly when they appear in mixed or layered forms. This challenge is especially pronounced in conversational settings, where subtle expressions like sarcasm or subdued emotions can obscure the true sentiment of an utterance. The included example plays a vital role within the guidelines, serving both as a training resource and a reference to help annotators identify and differentiate subtle emotional nuances. Such cases often necessitate a thorough examination of both the specific lexical content and the broader conversational context to accurately identify the predominant emotional state.

For example, an utterance that initially appears to convey happiness may, upon closer inspection, reflect hidden sarcasm or disappointment. By participating in detailed, scenario-driven training, annotators are equipped to handle these complexities, ensuring the consistent and precise labeling of emotional and sentiment categories throughout our dataset. Additional examples can be found in Appendix A.

3.7 Example: Happiness vs. Sarcasm

- **Dialogue Snippet:** "Harika, yine bir tartışma daha." / "Great, another argument."
- **Context:** Said after a series of arguments.
- **Annotation Challenge:** Determining whether this is an expression of genuine happiness or sarcasm.
- **Guideline Application:** The phrase implies dissatisfaction despite its positive wording, leading to the interpretation of sarcasm.

- **Final Sentiment:** If interpreted as sarcasm within a negative context, the sentiment is **Negative**.

We also showed examples of the concept of emotional blends, where multiple emotions might be present in a single utterance, reflecting the complexity of human interactions. For example, in a case study involving mixed emotions of happiness and surprise, annotators learned to distinguish primary from secondary emotions based on contextual cues such as preceding and following utterances. In the guideline, we incorporated relevant example given below:

3.8 Example: Anger vs. Neutral

- **Dialogue Snippet:** "Yine geç kaldın, bu gerçekten sinir bozucu!" / "You're late again, this is really annoying!"
  "Ama artık daha dikkatli olacağını biliyorum." / "But I know you'll be more careful from now on."

- **Context:** Said during a discussion about repeated lateness.

- **Annotation Challenge:** Distinguishing between anger due to frustration and a neutral statement showing understanding.

- **Guideline Application:** Consider the tone and content of both sentences. The first sentence indicates anger with strong, confrontational words (labeled as Anger), while the second sentence shows a neutral tone with a hopeful and understanding statement (labeled as Neutral).

- **Final Sentiment:** If the anger is more prominent and intense, the final label should be **Anger** (Negative). Otherwise, if the neutral tone significantly balances the overall sentiment, it can be **Neutral**.

Other relevant examples were included in Appendix B.

Guideline documents were meticulously prepared in Microsoft Word format and introduced to annotators during the Initial Workshop, where each section of the document was thoroughly explained. These documents included definitions, examples, and instructions for handling edge cases. Following the introductory session, hands-on practice was conducted using real and simulated examples drawn from the Couple Dialogue dataset to expose annotators to practical scenarios they would encounter in the data. These practice sessions helped bridge the gap between theoretical knowledge and real-world application by allowing annotators to engage directly with complex or ambiguous cases. Annotated utterances were shared in CSV format via Google Drive, where annotators could download, annotate, and re-upload their files. Each annotator worked independently initially to ensure a diverse range of interpretations.

To minimize erroneous labeling and promote consistency, calibration meetings were regularly held. During these sessions, challenging cases where annotators disagreed were discussed collectively to reach a consensus. This collaborative approach ensured a high level of accuracy and alignment with the project's standards.

3.9 Agreement Assessment

Due to resource constraints, we enlisted four volunteer annotators to validate and refine the baseline annotations we had previously established. To maintain reliability, we concentrated on annotating a detailed subset comprising 10% of the dialogues. Specifically, the volunteers annotated the dialogues of 12 couples, encompassing a total of 1609 utterances. The reliability of these annotations was evaluated using various statistical measures, with our initial annotations serving as the ground truth for comparison. Out of the 1609 utterances, 656 were labeled as negative, 631 as neutral, and 322 as positive.

To determine the average agreement score, we compared the labels provided by

the five annotators (our initial labels plus those of the four volunteers). For each utterance, if three or four volunteer annotators agreed on a label (positive, negative, or neutral), that label was defined as the "agreed" label. We then compared this agreed label to the ground truth label; a match was considered an agreement. The overall agreement score, representing the proportion of agreed labels matching the ground truth, was approximately 74.02% across all utterances. Furthermore, we assessed agreement by grouping sentiments into two broader categories: subjective (positive and negative) and objective (neutral). Under this categorization, we achieved an average agreement score of 78.06

The 4.04% difference between the overall agreement score and the subjectivity/objectivity agreement score underscores the difficulty of distinguishing between subjective and objective sentiments. This discrepancy may stem from the inherent challenges in consistently identifying subjective sentiments, particularly given the smaller proportion of positive samples. The higher agreement score for the broader subjective versus objective classification suggests that annotators found it relatively easier to agree on the overall tone (subjective vs. objective) than on the precise sentiment classification (positive, negative, or neutral).

Further, we employed Fleiss' Kappa, a statistical measure that evaluates the consistency of agreement among multiple raters, accounting for the possibility of agreement occurring by chance. This metric is ideal for analyses involving more than two annotators, making it more suitable for our study with five annotators than Cohen's Kappa, which is designed for just two raters. The formula for Fleiss' Kappa is defined as:

$$\kappa = \frac{\mathrm{P}_0 - \mathrm{P}_\mathrm{e}}{1 - \mathrm{P}_\mathrm{e}} \tag{3.1}$$

where P_0 is the observed agreement among raters, and P_e is the expected agreement by chance.

To calculate P_e, we first determine the proportion of each sentiment label assigned by all raters over all utterances, then square these proportions and sum them to reflect the

likelihood of random agreement if raters were guessing based on the distribution of labels.

Applying this formula, we achieved a Kappa score of 59.4%, which places our agreement level in the "moderate" category, traditionally defined as scores between 41% and 60%. Notably, our score is close to the "substantial" category, which begins at 60%, indicating a relatively high level of consensus among the annotators given the complexity of sentiment analysis. The calculated expected agreement (P_e) was 36.01%, which is significantly lower than the observed Kappa score. This indicates that our annotation process is far from random, reflecting a robust and reliable procedure, as Fleiss' Kappa specifically accounts for the likelihood of chance agreement, which the expected agreement percentage does not. The higher observed Kappa score relative to the expected agreement underscores the effectiveness of our annotation protocol and the consistency of our annotators' decisions.

In the context of Couple Dialogue data, where the presence of mixed emotions, sarcasm, and informal language is common, achieving a Kappa value approaching the "substantial" category highlights the strength of our annotation process. Despite these challenges, our annotators demonstrated a relatively high level of agreement, underscoring the reliability and consistency of the annotations.

Alongside the overall metrics, we also calculated the average agreement among the four volunteer annotators separately. This agreement was slightly lower at 72.48% compared to the overall score of 74.02%. Such a minor reduction was anticipated and reflects the natural variability in human interpretations of sentiment categories. However, this variability is expected in manual annotation tasks and does not compromise the overall reliability of the dataset. Importantly, the strong level of agreement among the volunteers further confirms the consistency of the sentiment labels.

It is worth emphasizing that the labels provided by the volunteers were used solely to evaluate the reliability of our initial ground truth annotations. This validation step was

crucial to ensure that our baseline labels were robust and dependable before their use in all subsequent experiments. Given that we achieved an agreement score of 74.02% based on 10% of the data, the baseline annotations—designated as the ground truth for experiments with both Contextual and Non-Contextual Models—proved to be reliable and required no additional refinement during the volunteer annotation phase.

For transparency, all relevant statistics and data distributions are summarized in Table 3.2.

Table 3.2: *Summary of Agreement Scores, Kappa Statistics, and Data Distribution in the Couple Dialogue Dataset. The table reflects the results of annotating 10% of the dialogues (1609 utterances) selected randomly from the dataset. Four volunteer annotators validated and refined the initial baseline annotations, which served as the ground truth for comparison. The data distribution includes 656 negative samples, 631 neutral samples, and 322 positive samples.*

Metric	**Value**
Overall Agreement Score	74.02%
Subjectivity/Objectivity Agreement Score	78.06%
Fleiss' Kappa Score	59.4%
Expected Agreement (P_e)	36.01%
Volunteer Annotators' Agreement Score	72.48%
Data Distribution	**Count**
Negative Samples	656
Neutral Samples	631
Positive Samples	322

4. METHODOLOGY

4.1 Models

4.1.1 The Non-Contextual Model

Our baseline Non-Contextual Model employs embeddings like BoW, TF-IDF, Word2Vec, and pre-trained Turkish BERT, using standard machine learning and deep learning techniques. This model assumes all utterances in a conversation are independent, meaning there is no shared information between contextual utterances. This approach overlooks the awareness of the speaker, the context of preceding utterances, and the influence of one speaker on another within the conversation.

Suppose that our dataset M consists of multiple dialogues, $M = \{D_i\}_{i=1}^{N}$ where N represents the number of dialogues. Each dialogue involves an input sequence of T utterances $[(u_1, s_1), (u_2, s_2)..., (u_T, s_T)]$ where each utterance $u_i = [u_{i,1}, u_{i,2}..., u_{i,K}]$ encompasses K words $u_{i,j}$ and is uttered by speaker s_i. Our goal is to predict the sentimental category $sent_i$ corresponding to each utterance u_i, where $sent_i \in$ C includes all sentimental categories, including positive, negative, and neutral, utilizing the non-contextual classifier. The plot of our context-independent architecture is given as Figure 4.1.

4.1.2 The Contextual Model

The contextual architecture captures individual speakers encapsulating the speaker, the context from the prior utterance, and the inter-dependency among utterances as conversation properties. Let our dataset M consists of multiple dialogues, $M = \{D_i\}_{i=1}^{N}$ where N represents the number of dialogues. Each dialogue involves an input sequence of T utterances $[(u_1, s_1), (u_2, s_2)..., (u_T, s_T)]$ where each utterance $u_i = [u_{i,1}, u_{i,2}..., u_{i,K}]$ encompasses K words $u_{i,j}$ and is uttered by speaker s_i. Our goal is to predict the sentimental category $sent_i$ corresponding to each utterance u_i, where $sent_i \in$ C with C, being

the set of all sentimental categories, including positive, negative, and neutral, by taking into account conversational dynamics utilizing the contextual classifier. Our context-dependent architecture schema is given as Figure 4.2.

Each model outlined in the Introduction section was selected for its relevance to context-based sentiment analysis and was specifically configured to enhance performance on the Couple Dialogue dataset. The prompt-based models, including GPT-4 and Llama 2, were customized using specific prompts. The fine-tuning approaches, involving BERT, GPT-3.5 Turbo, and Llama 2, required parameter adjustments to improve sentiment detection capabilities. The embedding-based strategy, employed by DialogueRNN, used enriched representations from both pre-trained and fine-tuned BERT models, as well as GloVe and GPT-based embeddings, enabling a comprehensive analysis of conversational dynamics.

In this study, Figures 4.1 and 4.2 were created using draw.io[1], a freely available online diagramming tool.

4.2 Text Preprocessing Steps

Preprocessing is essential in natural language processing tasks as it significantly impacts the performance of machine learning algorithms. The goal is to improve data quality by removing noise, reducing variability, and standardizing text.

4.2.1 Data Cleaning

Data cleaning generally comprises multiple tasks, such as standardizing data formats using encodings like ISO/IEC or UTF-8, expanding contractions (e.g., "we'll" to "we will"), and removing unnecessary components such as Hypertext Markup Language

[1] `https://app.diagrams.net/`

(HTML)/Extensible Markup Language (XML) tags, punctuation, numerical values, single characters, and redundant spaces. Additional steps include breaking text into smaller units like words, sentences, or n-grams through tokenization, eliminating stopwords, converting text to lowercase, and applying stemming or lemmatization to derive base forms of words.

For our Turkish dialogues, we employed International Organization for Standardization (ISO) encoding, expanded contractions, removed numerical values, single characters, and excessive spaces, and ensured all text was converted to lowercase. These preprocessing operations were applied to both Contextual and Non-Contextual Models, collectively termed as "basic-preprocessing."

4.2.2 Morphological Parsing

Owing to Turkish's complex morphology, we explored the influence of specific suffixes on sentiment classification within our Non-Contextual Model. For this purpose, we used the Dilbaz morphological analyzer[2] [90]. Initially, we corrected misspellings, identified English words, and refined compound nouns using the Turkish SpellChecker[3] tool [91].

Subsequently, we conducted sentence-level morphological parsing. A disambiguation tool was utilized to determine the most appropriate parse for each word, enabling us to obtain the surface form of each word within the utterance.

In Turkish, removing suffixes can negatively affect sentiment classification, as certain suffixes carry strong sentiment-related cues. Therefore, we generated a partial surface form that retains these key suffixes, a technique also used in [67].

Our focus was directed towards three critical morphemes: negation, pity, and wishes. Negation morphemes were detected in specific verbal suffixes (e.g., ma/me) and

[2] https://github.com/StarlangSoftware/TurkishMorphologicalAnalysis
[3] https://github.com/starlangsoftware/TurkishSpellChecker

word suffixes like "siz," "suz," and "süz." For example, the word "yaptım" ("I did") conveys a positive sentiment, whereas "yapmadım" ("I did not do") becomes negative due to the "ma" suffix. Similarly, the morpheme "siz" in "midesiz" ("tasteless") transforms the noun "mide" ("stomach").

Expressions of desire or regret were also identified, such as "keşke görse" ("I wish he/she saw"), where the word "keşke" signals a wish. Additionally, morphemes like "-cagız" in "çocukcagız" ("poor child") reflect pity. These sentiment-laden morphemes were shown to influence classification results, as discussed in the Results section. Table 4.1 provides an example of the preprocessing steps applied to a sample utterance.

4.2.3 Preprocessing Modules

For the Non-Contextual Model, we developed three distinct data forms. Since punctuation can contribute to data sparsity and increase the feature size of word embedding vectors, and stopwords often lack relevance for context-free tasks, we adopted varying preprocessing strategies.

In the first form, referred to as “detailed preprocessing applied on the root form,” we eliminated all punctuation and stopwords. This approach aimed to streamline the data by reducing unnecessary elements that might otherwise inflate the feature space.

The second form, named “partial preprocessing applied on the root form,” retained stopwords and specific punctuation marks, including exclamation points and ellipses. While some stopwords provide critical prepositional or negational meaning, punctuation marks can capture emotional intensity, which is vital for sentiment analysis. For example, the exclamation mark “!” in the utterance “Kalk buradan!” (“Get out of here!”) amplifies the sense of anger and highlights the negative sentiment. Removing such punctuation risks losing essential emotional cues, which could degrade the performance of sentiment-based embeddings.

Table 4.1: *Illustrative examples of preprocessing operations applied to a sample utterance in Turkish, demonstrating steps from raw text to more processed forms.*

Method	**Utterance**
Raw text	İçimi boşaltamamış oluyorum, rahatlayamamış oluyorum. (I can't empty myself, I can't relax.)
Tokenization	içimi boşaltamamış oluyorum , rahatlayamamış oluyorum .
Morphological parsing	**içimi**: iç+noun+A3sg+P1sg+Acc **boşaltamamış**: boşalt+Verb+Neg+Narr+Adj **oluyorum**: ol+Verb+Pos+Prog1+A1sg **,**: ,+Punc **rahatlayamamış** : rahatla+Verb+Neg+Narr+Adj **oluyorum**: ol+Verb+Pos+Prog1+A1sg **.**: .+Punc
Negation handling	boşaltamamış_ rahatlayamamış_

The final form, called “word-partial,” was derived from the second form. It preserved words that contained significant suffixes, a strategy referred to as “partial preprocessing applied on the surface form.”

For the Contextual Model utilizing Pre-trained Turkish BERT, we applied only basic preprocessing steps. These tasks retained both punctuation marks and stopwords to ensure the model fully leveraged the rich contextual information embedded in the text.

Figure 4.1: *Non-Contextual Sentiment Detection Architecture.*

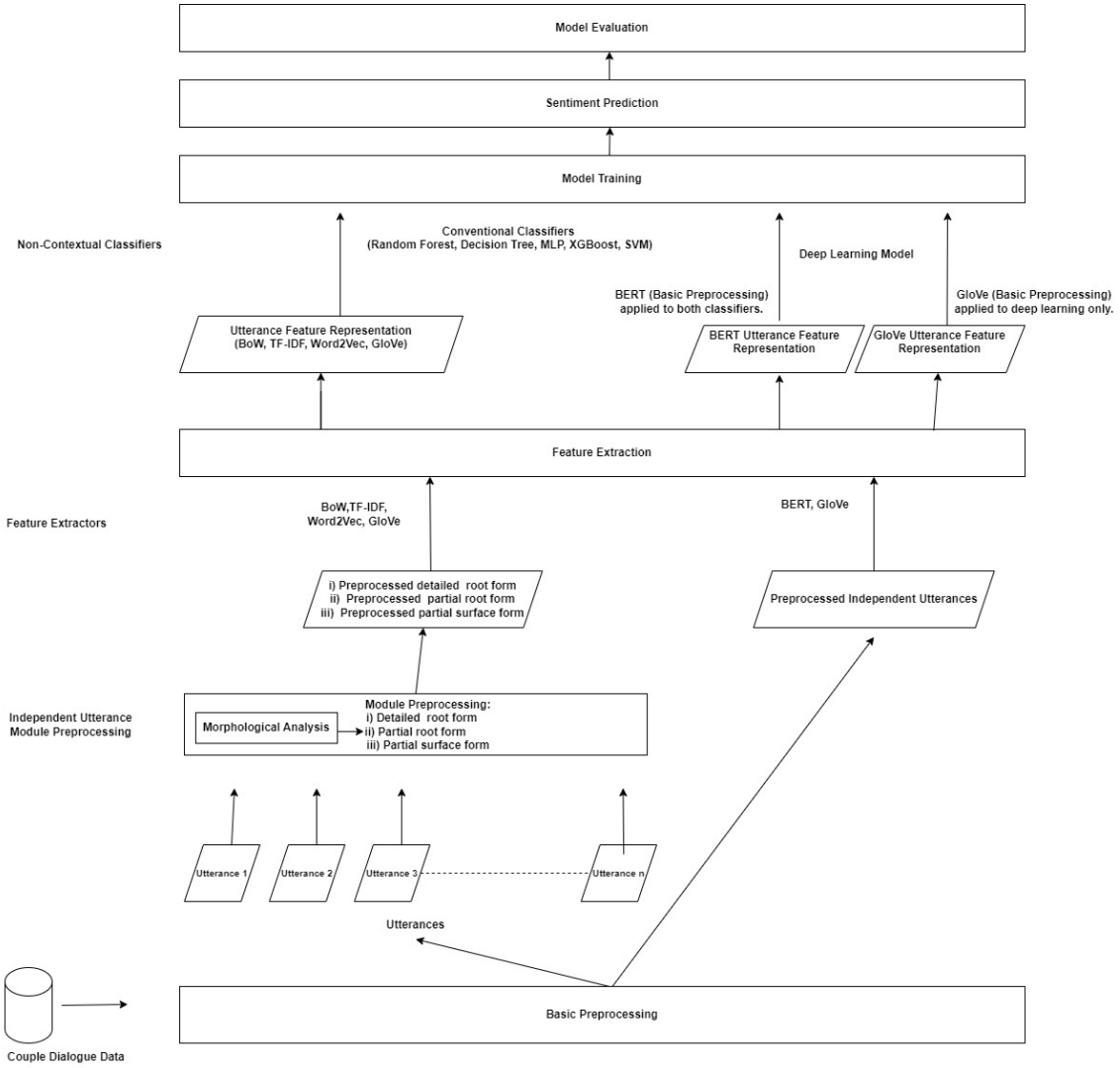
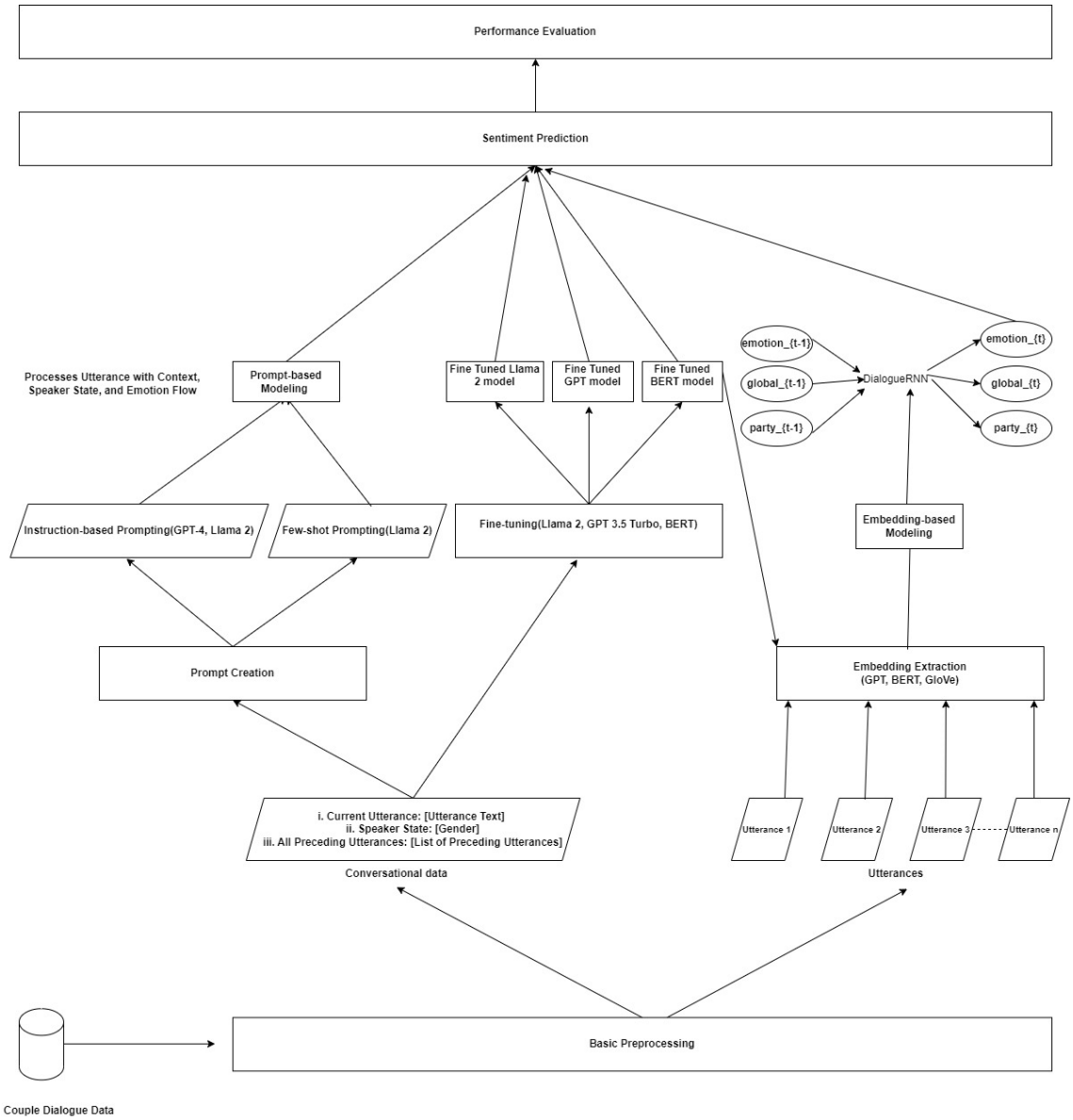


Figure 4.2: *Contextual Sentiment Detection Architecture.*



5. UTTERANCE FEATURE EXTRACTION

In this study, we applied various word representation techniques to convert text data into vector formats, making it suitable for machine learning algorithms. Techniques included "BoW", "TF-IDF", "Word2vec", "GloVe", "Pre-trained Turkish BERT", "Fine-tuned Turkish BERT", and OpenAI's "text-embedding-3-small". For the Non-Contextual Model, we used BoW, TF-IDF, Word2vec, GloVe, and pre-trained Turkish BERT. The Contextual Model, specifically DialogueRNN, used embeddings from GloVe, pre-trained Turkish BERT, fine-tuned Turkish BERT, and text-embedding-3-small at the utterance level.

5.1 BoW

BoW [92] is a straightforward method that quantifies word frequencies within sentences, transforming them into vectors where the length corresponds to the corpus vocabulary size. We used the Gensim library's [93] "doc2bow" model to convert text data into BoW vectors, which were then used as input for various machine-learning algorithms to perform sentiment classification.

5.2 TF-IDF

TF-IDF measures a word's relevance to a document within a collection or corpus [94]. It increases the importance of words unique to specific documents and reduces the significance of words common across all documents. We generated TF-IDF vectors for each utterance and used them to train classifiers for effective sentiment analysis.

5.3 Word2vec

Word2vec provides nuanced representations by capturing semantic and syntactic relationships between words [95]. We used both the CBOW and Skip-gram models to

develop dense vector representations, with settings including vector dimensions of 50, a context window of 3, a minimum word frequency of 2, and three worker threads on a corpus of 4,656 unique words.

5.4 GloVe

GloVe creates word embeddings using a co-occurrence matrix that captures the frequency of word pairs within a corpus. We applied pre-trained GloVe embeddings with 300 dimensions, trained on the Boun Web Corpus and the Huawei Corpus [96]. These embeddings were tuned for optimal quality, with adjustments to the window size, iteration limits, and other parameters.

5.5 Pre-Trained Turkish BERT

BERT [12] is built on the Transformer architecture [3], which leverages an attention mechanism to capture word context through bidirectional training. Unlike sequential models such as LSTMs [2], which process input step by step, BERT enables parallel computation, significantly improving training efficiency.

The architecture is composed of multiple components: the Embedding Layer utilizes BertTokenizer for subword tokenization, dividing sentences into smaller segments. These tokens are then processed by BertEmbeddings to generate word, positional, and type embeddings, which are combined and refined using layer normalization and dropout. The Encoder consists of 12 layers (BertLayers), where each layer includes a BertAttention module that computes attention scores, producing 12 contextual word representations for each layer. The final output from the encoder is pooled using the [CLS] token, which is particularly important for classification tasks.

For our study on Turkish couple dialogues, we utilized BERTurk [97], a pre-trained Turkish BERT model trained on a large Turkish corpus. This model features a

vocabulary size of 128k, 12 encoders, 12 self-attention heads, and a hidden dimension of 768, resulting in a total of 110 million parameters. We employed this configuration to derive utterance-level feature vectors, which were further integrated into advanced deep-learning frameworks such as DialogueRNN for sentiment analysis. Specifically, the [CLS] token output from the pre-trained Turkish BERT served as a crucial input for both the Non-Contextual and Contextual Models.

5.6 Fine-Tuning Turkish BERT on Couple Dialogue Data

We fine-tuned the pre-trained Turkish BERT model on our custom Couple Dialogue dataset, specifically designed for conversational sentiment analysis. The fine-tuning process was carefully optimized for this task, employing a learning rate of 1e-5, a batch size of 16, and three training epochs. To mitigate overfitting, a weight decay of 0.01 was applied. Throughout training, performance was closely monitored using the loss metric, and the iteration with the lowest loss was selected for subsequent usage.

Once the model achieved optimal performance, it was saved to retain its trained parameters. This fine-tuned model was then used to extract high-quality embeddings, specifically leveraging the [CLS] token, which encapsulates a comprehensive representation of the input sequence. These embeddings were provided as input to the DialogueRNN architecture, a model designed for sentiment analysis in conversational contexts. This approach effectively illustrates how integrating pre-trained models with task-specific fine-tuning techniques can significantly enhance sentiment classification accuracy.

5.7 text-embedding-3-small

Our project also incorporated OpenAI's text-embedding-3-small, a recent advancement in text embedding technology. This model showed significant improvements over

its predecessor, text-embedding-ada-002 model[1], in both performance metrics and cost efficiency. Notable enhancements were seen in benchmark assessments like the Multi-Language Retrieval (MIRACL) [98] and the English Task Benchmark (MTEB) [99], with scores of 44.0% and 62.3%, respectively. Additionally, the cost-effectiveness of text-embedding-3-small improved dramatically, with costs reduced by fivefold.

Although the more powerful text-embedding-3-large model[2] was available, we chose text-embedding-3-small due to budget constraints. This model effectively handled complex language tasks within financial limits. Using the OpenAI Python package and API, we generated embeddings as 1536-dimension floating-point vectors for each utterance. These embeddings served as crucial features in the DialogueRNN framework, highlighting the model's effectiveness in understanding human language nuances.

5.8 Incorporation of Morphological Features

We stated in the Morphological Parsing subsection that we included a morphological parser, Dilbaz, to dissect the intricacies of the Turkish language. The parser effectively handled morphological structures, identifying three key features: negation, pity, and wishing forms, which are crucial for accurately capturing the semantic content of the language [67]. Below, we show how we detected these morphemes.

- **Negation**: We identified negations by seeking the "neg" morpheme in the parsed form.
  - Exp: `Yorgun+Adj değil+Neg -im+Copula` (I am not tired)
- **Pity**: We identified pity by seeking the `root+Noun+Dim` structure.
  - Exp: `kız+Noun+Dim` (kızcağız - dear/little girl)

[1] `https://openai.com/blog/new-and-improved-embedding-model`
[2] `https://openai.com/blog/new-embedding-models-and-api-updates`

- **Wishing**: We identified wishing by seeking the `gül+Verb+Cond` structure.
  - Exp: `gül+Verb+Cond` (gülse - if he/she/it laughs)

We underlined the identified forms along with their corresponding morphemes. For example, we distinguished between `gel` (come), `gelse` (if he/she/it comes), and `gelmese` (if he/she/it does not come) as separate entities within the feature space when they appeared in the same utterance. Specifically, for the TF-IDF model, we calculated TF-IDF scores for `gel`, `gelse`, and `gelmese` individually. Similarly, for the GloVe embedding model, the co-occurrence scores for the three forms were calculated separately. In the BoW model, each morphological variant was treated as a distinct token, while in the Word2 Vec model, each form was considered a separate word, allowing the model to generate unique embeddings for each morphological variation. This form separation was applied exclusively in the third pre-processing configuration, termed "partial preprocessing applied on the surface form", where morphemes were explicitly taken into account.

Conversely, in the second form, labeled "partial preprocessing applied on the root form", stopwords were retained, and punctuation such as exclamation marks and ellipses was preserved. In this case, `gel`, `gelse`, and `gelmese` were represented using the same vector, as only the root form (`gel`) was considered. Similarly, in the first configuration, known as "detailed preprocessing applied on the root form", where all punctuation and stopwords were removed, all three variants were again represented as `gel`, similar to the second form.

Following the creation of word embedding models, including BoW, TF-IDF, Word2-Vec, and GloVe, we extracted feature embeddings for use in the Non-Contextual Model. These embeddings were subsequently fed into traditional classifiers under the Non-Contextual Model framework. The results obtained from these experiments are reported in detail in the Experimental Studies and Results section.

6. UTTERANCE CLASSIFIER

Utterance-level features were extracted using BoW, TF-IDF, Word2Vec, GloVe, and the pre-trained Turkish BERT model. These features were then classified using Non-Contextual Model classifiers, which employ algorithms that disregard the interdependence among different speakers in a conversation.

In contrast, the Contextual Model classifiers, which are designed to capture the flow and dynamics of the conversation, consider both the preceding and subsequent utterances. These classifiers are divided into three categories: prompt-based, fine-tuned, and embedding-based. The embedding-based models utilize a variety of features obtained from GloVe, pre-trained Turkish BERT, fine-tuned Turkish BERT, and text-embedding-3-small, enabling a detailed analysis of conversational context.

All computational tasks were executed on a local high-performance workstation equipped with an Intel® Core™ i7-9700K, 64GB of random-access memory (RAM), and an NVIDIA GeForce RTX 4090 graphics processing unit (GPU) with 24 GB of memory.

This section will further elaborate on both the Non-Contextual and Contextual classifiers, explaining their mechanisms and the rationale behind their use.

6.1 The Non-Contextual Classifier

We classify the classification methods employed in this study into two categories. The first category involves the application of conventional machine learning algorithms. In the Preprocessing Modules Section, we elaborated on three types of preprocessing modules, namely, “detailed preprocessing applied on the root form”, “partial preprocessing applied on the root form”, and “partial preprocessing applied on the surface form”. These three different data forms of utterance representations obtained from the BoW, TF-IDF, Word2Vec, and GloVe models were employed as inputs for these methods. As BERT constitutes a contextual embedding model, we generated a unified data form of utterance

feature representation, implementing fewer preprocessing operations as explained in the Data Cleaning section, and incorporated it into this network.

The second group consists of a deep-learning architecture that utilizes pre-trained GloVe word embeddings and the pre-trained Turkish BERT model as inputs. For these embeddings and the model, we applied basic preprocessing operations to the utterances.

6.1.1 Classification Algorithms

The sentiment classification of each utterance is performed using several machine learning algorithms, including Decision Tree, Random Forest, Gradient Boosting, XGBoost, SVM, and MLP. Our generated feature embeddings became the input of those classifiers. All classifiers were implemented using Scikit-Learn library's algorithms [100].

6.1.2 Deep Learning Model

Unlike commonly known classifiers, we developed a deep-learning architecture using Keras [101] under the Non-Contextual Model classifier framework. The utterance feature representations of pre-trained GloVe and the pre-trained Turkish BERT model were employed as inputs for this network for the sentiment classification task. It is important to note that these inputs underwent only basic preprocessing before being fed into the model.

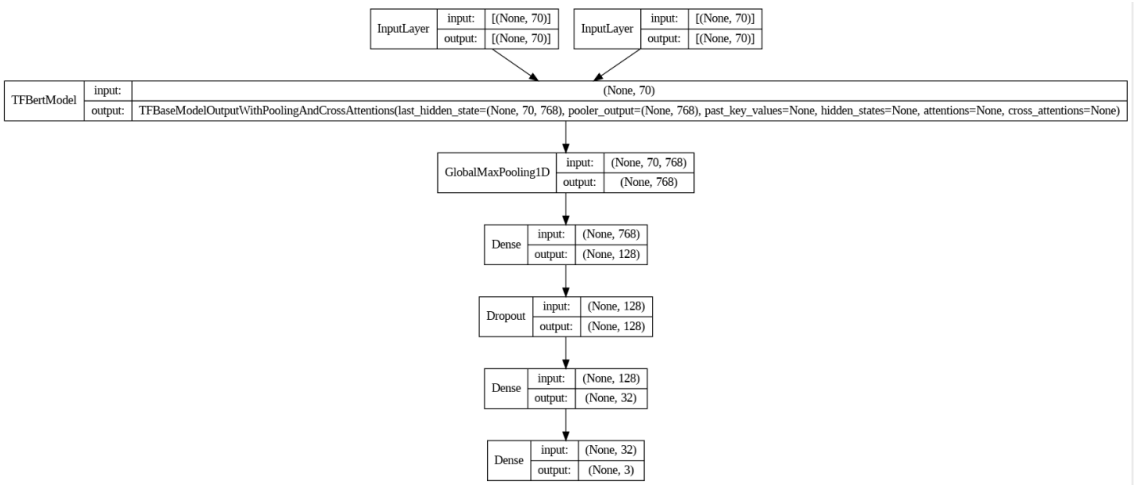
We created a BERT layer using Keras and integrated BERT utterance embeddings from TF-Hub for the sentiment classification problem. The BERT layer requires two input layers, namely, input_ids and attention_mask. Keras provides an Embedding layer containing hidden states of the BERT layer. Subsequently, we constructed GlobalMaxPooling1D, Dense, and Dropout layers, and trained our model using the parameters outlined in Table 6.1. For the GloVe experiment, an Embedding layer encompassed an embedding matrix derived from pre-trained GloVe embeddings.

The constructed architecture including Bert Layer is represented in Figure 6.1.

Table 6.1: *Specification of Parameters for the Deep Neural Network Architecture Utilized in the Study.*

Parameter name	**Value**
Optimizer	Adam
Epsilon	1e-08
Loss function	Categorical cross-entropy
Maximum length of each sentence	70
Number of epochs	30
Batch size	4

Figure 6.1: *Deep learning architecture including a BERT Layer.*



6.2 The Contextual Classifier

To provide some background, we explore the advanced contextual LLMs that have been pivotal in our conversational sentiment analysis task. These models demonstrate significant technological advancements that enhance their suitability for complex linguistic processing.

6.2.1 Background for LLMs

6.2.1.1 Background for GPT-3.5

GPT-3.5 Turbo acts as a bridge between GPT-3 and GPT-4, maintaining a high parameter count while integrating substantial architectural enhancements. This model improves efficiency in data processing and responsiveness, with optimized attention mechanisms that enhance its performance in dynamic conversational settings. GPT-3.5 Turbo is particularly effective in real-time applications requiring fast and accurate text generation, such as conversational agents.

GPT-3.5 Turbo's efficiency in processing and generating text makes it ideal for analyzing the rapid and context-rich exchanges typical in real-world couple dialogues. Its ability to quickly grasp and respond to dynamic conversational contexts enhances its effectiveness in sentiment analysis tasks where understanding subtle emotional cues is essential.

6.2.1.2 Background for GPT-4

Launched in 2023, GPT-4 builds on OpenAI's legacy with approximately 175 billion parameters, further enhancing its transformer-based architecture. It features an advanced encoder-decoder configuration that improves the model's ability to handle longer context windows. This capability is particularly valuable for generating coherent, detailed, and context-aware outputs, making GPT-4 well-suited for complex content generation, sophisticated dialogue systems, and tasks requiring deep linguistic understanding.

GPT-4's architectural advancements make it particularly effective for sentiment analysis in complex and extensive couple dialogues, where maintaining coherence across longer conversational exchanges is vital. Its ability to produce contextually relevant and detailed outputs ensures accurate detection of sentiment changes throughout the dialogue.

6.2.1.3 Background for Llama 2

Released in July 2023 as part of Meta GenAI's open-source initiative, LLaMA 2 marks a significant improvement over its predecessor, LLaMA. The model family includes versions ranging from 7 billion to 70 billion parameters. It supports a context length of up to 4K tokens—double the capacity of the original LLaMA—which enables deeper comprehension of extensive text sequences. This improvement is particularly crucial for managing long, complex dialogues and maintaining contextual continuity.

LLaMA 2 departs from conventional Transformer architectures by integrating Root Mean Square Layer Normalization (RMSQLN) [102] and Swish-Gated Linear Units (SwiGLU) [103] instead of the standard ReLU activation function. Additionally, Rotary Position Embedding (RoPE) [104], which combines absolute and relative positional information, enhances both learning dynamics and the attention mechanism. These architectural modifications improve text generation and comprehension, particularly for inputs with complex positional relationships.

The training process for LLaMA 2 begins with self-supervised learning on a vast, unlabeled text corpus. It is then refined through supervised fine-tuning for targeted applications, such as dialogue systems, and further optimized using Reinforcement Learning with Human Feedback (RLHF) [105] to enhance real-world conversational performance. In this study, we utilized the Trendyol LLM chat variant, based on the LLaMA 2 7B model, for both our prompt-based and fine-tuning methodologies.

LLaMA 2's strength in managing long conversational sequences and preserving contextual integrity makes it particularly effective for analyzing emotional dynamics in extended couple dialogues. Its improved attention mechanisms and enhanced positional embeddings enable it to process intricate dialogues with precision, making it a robust choice for sentiment analysis tasks involving real-world conversational data.

6.2.2 Prompt-based Modeling

Prompt engineering plays a critical role in AI applications by ensuring that prompts are precise, clear, and specific, thereby minimizing the risk of generating irrelevant or overly generalized responses. A well-designed prompt provides the necessary context and directs the model to produce outputs that align succinctly with the intended task, making it particularly effective for tasks that require contextual precision, such as sentiment analysis.

For our sentiment analysis tasks, we applied two distinct prompting techniques using GPT-4 and LLaMA 2:

- **Instruction-based Prompting:** In this approach, we constructed prompts that explicitly instruct the AI to perform a given task, ensuring that the output meets specific, predefined requirements. This strategy was applied to both GPT-4 and LLaMA 2 for the conversational sentiment analysis tasks. For instance, an instruction-based prompt might ask the AI to evaluate the emotional tone of the final exchange in a dialogue and classify it as either positive, negative, or neutral, providing only a single-word response.

  Special care was taken to consider the nuances of male-female interactions within couple dialogues. By factoring in speaker gender and dialogue dynamics, the prompts were designed to help the model accurately assess sentiment, ensuring that subtle variations in communication patterns were appropriately captured for more reliable sentiment classification.

- **Few-shot Prompting:** This technique involves providing the model with a series of examples—each comprising an input and its corresponding correct output—to guide the model in understanding the task requirements. Few-shot prompting allows the model to identify patterns or structures necessary for the task without

requiring additional programming. For our study, we implemented this method with LLaMA 2 and compared its performance against the results obtained through instruction-based prompting.

The examples used for few-shot prompting were carefully curated to reflect a variety of common emotional exchanges within male-female couple dialogues. We included examples that covered diverse levels of sentiment intensity and varying degrees of dialogue complexity, ensuring the model could generalize effectively across a wide range of conversational scenarios.

During the development of our prompt structures for conversational sentiment analysis, we conducted numerous trials. We found limited guidance in existing literature, leading us to devise a prompt structure that specifies for each utterance within a dialogue, the AI should be provided with all previous utterances and tasked with predicting the sentiment of the current utterance. Thus, the model is asked to determine the sentiment of the most recent utterance, considering all prior context. Since our dataset is written in Turkish, we applied the instructions in Turkish, as shown in Table 6.2 and Table 6.3. The English versions are provided to accommodate readers not familiar with Turkish, ensuring broader accessibility and understanding.

The Table 6.2 illustrates the application of an instruction-based prompt designed for GPT-4 and Llama 2, highlighting the directive nature of our prompts and their focus on critical dialogue aspects to assess sentiment accurately.

Table 6.3 shows how few-shot prompting can be utilized to enhance the model's understanding of diverse emotional contexts within dialogues, providing several scenarios with predetermined emotional tones.

By employing these advanced prompt engineering techniques, we have tailored our training, validation, and test datasets to effectively correspond with the required chat completion styles, thus improving the models' performance in practical AI applications.

Table 6.2: *Example of an instruction-based prompt for sentiment analysis tasks using GPT-4 and Llama 2, presented in both Turkish and English. The primary task is conducted in Turkish, with the English translation provided for clarity and broader accessibility.*

Turkish Version	**English Version**
Son konuşmanın duygusal tonunu sadece "pozitif", "negatif", ya da "nötr" olarak değerlendir. Cevabın tek bir kelime olmasına dikkat et. Konuşmacı cinsiyeti ve diyalog dinamiklerinin önemini göz önünde bulundur. Sen yardımcı bir asistansın. En iyi cevabı vermek için verilen talimatları dikkate al. Her konuşma kadın veya erkek konuşmacılardan biri tarafından yapılır. "None" öncesi konuşmaların olmadığını gösterir. **Yeni Diyalog**: Diyalog: Konuşma 1: [Kadın: Gerçekten sinirliyim, sen başlayabilirsin.] Konuşma 2: [Erkek: Ne başlayacağım?] Konuşma 3: [Kadın: Hıh!] Konuşma 4: [Erkek: Hani ne sinirleniyorsun?] Konuşma 5: [Kadın: Arkadaşlar konusuna sen direkt başlayabilirsin bence.] **En Sonki Konuşmanın Beklenen Duygu Tonu**:	Evaluate the emotional tone of the last conversation as either "positive", "negative", or "neutral". Ensure your response is a single word. Consider the importance of speaker gender and the dynamics of the dialogue. You are a helpful assistant. Make sure your response follows the given instructions. Each conversation is conducted by either a male or female speaker. "None" indicates there are no prior conversations. **New Dialogue**: Dialogue: Conversation 1: [Female: I'm really annoyed, you start.] Conversation 2: [Male: Start what?] Conversation 3: [Female: Hmph!] Conversation 4: [Male: What are you annoyed about?] Conversation 5: [Female: I think you should start talking about friends directly.] **Expected Emotional Tone of the Last Conversation**:

6.2.2.1 Prompt-based Sentiment Prediction with GPT-4

In our research, we utilized OpenAI's GPT-4 for conversational sentiment analysis, emphasizing the critical role of instruction-based prompt engineering. This process involved configuring the technical aspects of the model to enhance its performance for our specific needs. Access to the GPT-4 engine was facilitated through a paid subscription to the OpenAI API, ensuring we could leverage the full capabilities of this advanced tool.

We configured the GPT-4 model as follows: The engine was set to "text-davinci-004" with a temperature of 0.5 to allow a moderate level of creativity in responses. The

response length was limited to 4096 tokens, suitable for longer conversations, and Top_p was set to 1.0, allowing the model to explore all possible response options. No frequency or presence penalties were applied, and responses were clearly segmented by a newline character. These settings enabled us to fine-tune the prompts, maximizing GPT-4's potential and achieving high accuracy in sentiment analysis.

6.2.2.2 Prompt-based Sentiment Prediction with Llama 2

For the implementation of Llama 2, we utilized the Trendyol LLM model, which is based on the open-source Llama 2 framework. The open-source nature of this framework makes it accessible to researchers and the broader community without incurring the costs associated with commercial platforms like OpenAI. This accessibility is particularly valuable in academic environments, where budget limitations can often restrict the scale and depth of research initiatives.

We customized the Trendyol LLM model configuration to align with the specific requirements of our project. Our tailored settings included setting do_sample=True to enable stochastic response generation, a temperature value of 0.3 to ensure stable and consistent responses, and selection controls of top_k=50 and top_p=0.9 to refine the relevancy of the outputs. Although the model's official documentation indicates a maximum token size of 1024, we extended this limit to 4096 tokens to accommodate the entire conversational context. This adjustment was essential for capturing the intricate nuances present in the extended dialogues we analyzed. These carefully chosen settings ensured the model configuration was optimized for the demands of our conversational sentiment analysis tasks.

The design and structure of the prompts played a pivotal role in determining the performance of Llama 2. Initially, we adopted an instruction-based approach that directed the AI to assign sentiment labels with minimal context. However, this method did

not achieve the desired accuracy levels, prompting us to implement a few-shot prompting strategy. By incorporating illustrative examples of dialogues representing various emotional tones—examples outlined in Table 6.3—we significantly improved the model's ability to comprehend and classify sentiments. This improvement highlights the critical role of effective prompt engineering in enhancing AI models' performance for specialized analytical tasks like conversational sentiment analysis.

6.2.3 Fine-Tuning

Fine-tuning in machine learning refers to modifying a pre-trained model to improve its performance on a particular task. This method is particularly advantageous in NLP, where large pre-trained models, such as those developed by OpenAI, can be adapted to specific tasks using relatively small datasets. Through fine-tuning, we capitalize on the model's extensive language comprehension while refining it to capture the detailed nuances of a specific application, such as sentiment analysis in conversational contexts.

6.2.3.1 Fine-tuning with GPT-3.5 Turbo

Due to budget constraints, our research utilized the paid fine-tuning capabilities of the GPT-3.5 Turbo model, specifically the "gpt-3.5-turbo-0125 variant" [1], available through the OpenAI API.

1. Fine-Tuning Process: We structured our dataset, for training, validation, and testing, into ".jsonl" files. Each data entry followed a structured format where sequences of messages were assigned roles like "system," "user," and "assistant." Table 6.4 is an example of the data structure used:

In this setup, "new_dialogue" encompasses both the current and all preceding utterances, while "expected_sentiment" classifies the sentiment of the latest utterance as

[1] `https://platform.openai.com/docs/models/gpt-3-5-turbo`

“positive”, “negative”, or “neutral”. The fine-tuning process employs a system prompt meticulously crafted to impartially evaluate the emotional tone of the final conversation. This prompt generates a single-word sentiment, taking into account variables such as speaker gender and dialogue dynamics, similar to the instructional guidance used in our prompt-based strategies for conversational sentiment analysis.

2. Training Strategy: For the training and validation phases, we provided the model with the label for the last utterance, allowing it to learn the relationship between dialogue context, speaker states (e.g., female, male), and sentiment. This method helped the model understand the intricate dynamics of conversational sentiment. The simplified example is shown in Table 6.5.

The model sees the entire dialogue and learns that the sentiment of the final utterance ("Kadın: Arkadaşlar konusuna sen direkt başlayabilirsin bence." / "Female: I think you should start talking about friends directly.") is negative. The fine-tuning process employs such examples to train the model effectively in evaluating the emotional tone of conversations.

During the fine-tuning process, we initially used OpenAI’s default training parameters, which were as follows: 2 epochs, a learning rate multiplier of 2, and a batch size of 12. These defaults are typically recommended for fine-tuning as they are tailored to the size of the dataset being used. However, as we monitored the training, validation, and total loss metrics via the OpenAI Fine-tuning API, we observed that the validation loss did not decrease significantly after 2 epochs.

To address this, we increased the number of epochs to 3, which led to a more effective reduction in the loss. This adjustment was critical in preventing the model from underfitting and ensuring it could better capture the nuanced patterns in the training data. The validation set was continuously used to monitor performance and ensure the model

did not overfit.

With the default parameters and the increased epoch count, the fine-tuned model processed a total of 6,682,384 tokens, respecting the 4096-token capacity of the GPT-3.5 Turbo model.

3. Outcome and Usage: The fine-tuning session was completed in just 0.78 hours (47 minutes). This fine-tuned model demonstrated effectiveness comparable to that of the DialogueRNN combined with fine-tuned BERT embeddings as shown in the Experimental Studies and Results section, showcasing GPT-3.5 Turbo's robust capabilities in nuanced conversational sentiment analysis. These results support previous findings that GPT models, through fine-tuning, are adept at handling tasks requiring sophisticated language understanding and generation in dynamic conversational scenarios.

6.2.3.2 Fine-tuning with Llama 2

PEFT [106] allows for more efficient fine-tuning of large pre-trained models by updating only a small subset of the model's parameters, rather than the entire model. This approach preserves the extensive knowledge base the model acquired during its initial training while allowing it to be refined for specific tasks. LoRA [107] specifically introduces low-rank matrices into the transformer layers, enabling targeted adjustments with minimal additional parameters. This technique is particularly useful in maintaining the model's efficiency while enhancing its performance for specialized tasks.

1. Fine-Tuning Process: Given our resource constraints, we implemented 4-bit quantization, which significantly reduces the memory requirements of our models. Quantization compresses the model, enabling it to operate within the 24 GB memory limit of our hardware while speeding up inference—an essential feature for real-time applications.

For this project's conversational sentiment analysis, we fine-tuned the Trendyol

LLM, a model based on Llama 2, to optimize it for chatbot interactions. We used the same fine-tuning parameters as those provided by Trendyol LLM's documentation, with one key modification: the learning rate. While the Trendyol LLM documentation suggests a learning rate of 2e-4 in the model configuration, we experimented with three smaller learning rates, each lower than 2e-4, to find the optimal setting. Through this experimentation, we observed that the validation loss decreased most effectively with a learning rate of 1e-6. Based on these results, we selected 1e-6 as the final learning rate for fine-tuning, alongside 3 epochs, to ensure a more gradual adaptation of the model. This combination helped reduce the risk of overfitting while allowing the model to learn effectively from the smaller and more specialized dataset used in this project.

The other parameters were set as follows:

- **LoRA alpha:** 128, which controls the scaling of the low-rank matrices introduced by LoRA.
- **Dropout rate:** 0.05, to prevent overfitting by randomly dropping units during training.
- **LoRA rank:** 64, specifying the rank of the low-rank matrices, focusing on all linear modules involved in causal language modeling.
- **Compute dtype:** `bfloat16`, chosen for its efficiency in handling the 4-bit quantized model.

2. Training Strategy: Training was conducted over three epochs with a micro-batch size of two, using an AdamW optimizer with torch-fused operations. This optimizer was selected due to its effectiveness in handling the nuances of transformer-based architectures. The maximum sequence length was set to 4096 tokens to ensure the model retained

comprehensive context from all previous utterances, which is crucial for understanding the flow of dialogue.

Since we obtained significant improvement in Weighted F1 scores with Few-shot Prompting compared to Instruction-based Prompting, we fine-tuned our data, including few-shot examples similar to sentiment examples given in Table 6.3. During the training and validation phases, the model was provided with labels for the current utterance, enabling it to learn the relationships between dialogue context, speaker states (e.g., female, male), and sentiment. In contrast, during the testing phase, the model was required to predict the sentiment without being shown the label, assessing its ability to generalize from learned examples.

It is important to note that, due to budget constraints, we were unable to incorporate such extensive examples into the fine-tuning of GPT-3.5 Turbo.

3. Outcome and Usage: Evaluations were conducted every 25 steps to continuously refine the model's performance, with the entire process lasting 47 hours. This training regime aimed to achieve minimal loss and was meticulously tracked using TensorBoard for detailed analysis, aligning with best practices in recent research. The overall parameter list is represented in Table 6.6.

6.2.3.3 Fine-tuning with Pre-Trained Turkish BERT

In this study, we fine-tuned the pre-trained Turkish BERT model on a specialized Couple Dialogue dataset tailored for conversational sentiment analysis. The fine-tuning process was carefully optimized to enhance performance for this specific task.

1. Fine-Tuning Process: We utilized a learning rate of 1e-5 and a batch size of 16 during both the training and evaluation phases. These settings were chosen based on initial testing to balance training speed and model performance. The model was trained

over three epochs, with the number of epochs determined based on the validation loss to ensure optimal performance without overfitting.

A weight decay of 0.01 was applied to further prevent overfitting while allowing the model to learn effectively from the data. Performance was rigorously monitored using the loss metric, and the model corresponding to the iteration with the lowest loss was saved. This approach ensured that the best-performing model was preserved for subsequent use in sentiment analysis.

2. Training Strategy: During fine-tuning, all preceding utterances of the current dialogue, the current utterance label, and speaker states (e.g., female, male) were included in the training and validation phases. This comprehensive inclusion allowed the model to learn the relationships between dialogue context, speaker characteristics, and sentiment. In the testing phase, the model was evaluated on its ability to predict sentiment labels without access to the correct answers, thus assessing its generalization capability.

3. Outcome and Usage:

- **Embedding Extraction:** The saved model was used to extract high-quality embeddings from the [CLS] token, which captures the holistic essence of the dialogues. These fine-tuned Turkish BERT embeddings were then utilized within a DialogueRNN architecture for sentiment analysis, as explained in the Utterance Feature Extraction section.

- **Direct Sentiment Classification:** The fine-tuned model directly classified sentiments within these conversations, demonstrating its precision in sentiment detection and its broad applicability in sentiment analysis tasks.

6.2.4 Embedding-based Modeling

This section details the technical implementation of DialogueRNN, which utilizes a variety of embedding sources to enhance its dialogue processing capabilities. Specifically, the model incorporates embeddings from pre-trained Turkish BERT, its fine-tuned variant, and what we refer to as text-embedding-3-small, herein identified as GPT-embeddings. Additionally, GloVe embeddings are used, all of which are discussed in the Utterance Feature Extraction section.

6.2.4.1 DialogueRNN Approach

DialogueRNN [29] efficiently models conversational dynamics by monitoring the evolving emotional states of each speaker, the contextual flow of the dialogue, and interactions between utterances. It accomplishes this through the use of three state representations: party states, global states, and emotional states, all updated using gated recurrent units (GRU) [108].

This model extends beyond traditional sequence-to-sequence (seq2seq) methods by integrating attention mechanisms, as explored in earlier studies [109, 110]. The attention mechanism allows the model to dynamically focus on the most relevant portions of the input sequence. This capability is especially critical for processing lengthy dialogues, as it helps the model retain a coherent understanding of the conversational context across multiple utterances. DialogueRNN operates by maintaining and updating three key states:

- **Global State:** Utilizes past utterances and states to model the current context of a dialogue. The global state is updated by a GRU and refined through an attention mechanism to ensure relevance to the current utterance.
- **Party State:** Represents individual speaker states throughout the dialogue, updated with each new utterance to reflect the latest conversational context and content.

- **Emotion/Sentiment State:** Initially designed to track emotions, this has been adapted in our study to the sentiment state to classify sentiments as positive, negative, or neutral effectively. This state is crucial for our conversational sentiment analysis, determining the sentiment based on the current dialogue dynamics.

6.2.4.2 Training and Evaluation

We utilized both k-fold cross-validation and standard split methods to evaluate the models. K-fold cross-validation provides a more comprehensive assessment by ensuring that every data point is used for both training and validation, which is especially useful in smaller datasets. In contrast, the standard split method was chosen for certain models due to its lower computational requirements, making it more feasible for models with extensive processing demands, such as those utilizing LLMs.

K-Fold Cross-Validation: A 10-fold cross-validation was conducted for the DialogueRNN classifier, where each fold consisted of 118 couple dialogues split into 7 training, 1 validation, and 2 testing folds. We optimized hyperparameters, specifically learning rate and epoch size, during the validation phase.

Standard Split Training: For our experiments, we implemented a dataset split of 70% for training, 10% for validation, and 20% for testing, applied uniformly to both the computationally demanding LLMs and the DialogueRNN model. This consistent evaluation setup ensured direct comparisons could be made between the DialogueRNN and the more computationally advanced LLMs.

We employed k-fold cross-validation for the Non-Contextual and DialogueRNN classifiers to thoroughly test using each part of the dataset. In contrast, LLM-based classifiers were evaluated using a standard split, aligning with common practices for large models.

6.2.4.3 Hyperparameters and Model Configuration

The standard split hyperparameters were:

- **Glove-DialogueRNN:** Learning rate of 1×10^{-3}, 30 epochs.
- **Bert-DialogueRNN and Other Bert Variants:** Learning rate of 1×10^{-5}, 50 epochs.
- **GPT-DialogueRNN:** Learning rate of 1×10^{-4}, 40 epochs.

Other model parameters for the k-fold cross-validation and standard split are given in Table 6.7. All models were trained with a batch size of one dialogue, due to memory constraints. The optimizer, loss function, and output size dimensions (D_G, D_P, D_E) for all Bert variants, Glove, and GPT-embedding were consistently maintained as per foundational parameters outlined in the literature [53].

Our approach in the Embedding-based Modeling phase involved extracting embeddings, incorporating them into the DialogueRNN framework, performing hyperparameter optimization for learning rates and the number of epochs during validation, and subsequently evaluating the outcomes on the test data. Table 6.8 provides a summary of the training durations for each type of embedding under both k-fold cross-validation and standard data split setups. As expected, k-fold cross-validation required longer training times due to the repeated hyperparameter tuning across multiple folds.

The training durations for embeddings derived from the pre-trained Turkish BERT model were longer compared to those from the fine-tuned BERT model. This is because the pre-trained BERT embeddings required additional processing to adapt the general-purpose features to the specific nuances of the Couple Dialogue dataset. Additionally, these reported training times account for both the embedding extraction process and the subsequent integration and processing within the DialogueRNN architecture.

6.2.5 Hyperparameter Tuning

This study employed a 10-fold cross-validation method to stratify the data for both Non-Contextual and DialogueRNN classifiers within the Contextual Model category. Each fold contained 118 couple dialogues, segmented into 7 training, 1 validation, and 2 testing folds. Due to the extensive computational demands of LLMs, k-fold cross-validation was deemed impractical for these classifiers. Instead, the dataset was divided into segments of 70% for training, 10% for validation, and 20% for testing to ensure uniform evaluation conditions across all models.

Hyperparameter optimization was performed using the Optuna framework [111], with the goal of maximizing the Weighted-average F1 score. Optuna was chosen for its ability to efficiently explore extensive hyperparameter search spaces through techniques like Bayesian optimization. This approach proved particularly effective in identifying the optimal set of parameters while reducing the number of iterations required, thus saving both computational resources and time compared to traditional grid search methods.

The optimization process involved executing 50 trials to identify the best-performing hyperparameters. These optimal parameters were subsequently used to train the models on the training folds and evaluate their performance on the testing folds. This process was repeated for each validation fold, and the final performance results were reported as the average across all folds.

The parameters tuned included:

- **Tree-based algorithms** (Decision Tree, Random Forest, Gradient Boosting, and XGBoost): tree depth, number of estimators, and learning rate.
- **SVM**: C value, gamma, and kernel type.
- **MLP**: activation function, alpha value, hidden layer sizes, learning rate, and solver type.

For the Deep Learning Model within our Non-Contextual Model framework, hyperparameter optimization was carried out as well. During each of the 10 validation folds, we fine-tuned the dropout value, activation function, and learning rate. Dropout values were adjusted to mitigate overfitting, while the activation function and learning rate were optimized to achieve a balance between the model's convergence speed and its accuracy.

Once the optimal hyperparameters were determined for each fold, they were applied to the corresponding test folds. The final performance was reported as the average Weighted F1 score, consistent with the methodology used for other Non-Contextual classifiers.

As discussed in the Embedding-based Modeling section, DialogueRNN also underwent hyperparameter optimization, specifically focusing on learning rate and epoch number adjustments during the validation phase of the k-fold cross-validation and standard split training setups.

6.2.6 Evaluation Metrics

Due to the imbalanced nature of our dataset, we adopted the Weighted-averaged F1 score to assess the performance of our classifiers. This metric was crucial for representing our success rates accurately, especially given the distribution of classes as depicted in Figure 4.1, where only 2083 out of 14294 utterances are positive. The Weighted F1 score is particularly effective in unbalanced datasets as it considers both precision and recall, crucial for a comprehensive evaluation of model performance.

$$\text{recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{6.1}$$

$$\text{precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{6.2}$$

$$F_1 = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} \tag{6.3}$$

Given the imbalanced nature of our dataset, relying solely on per-class scores could result in a distorted understanding of model performance. To address this, we calculated the Weighted-average F1 score, which averages the F1 scores of each class while weighting them by the number of true instances in each class (i.e., the class support). This approach ensures that classes with a larger number of samples have a proportionally greater impact on the overall score, aligning the evaluation metric with the actual class distribution.

For instance, if the positive class accounts for only 15% of the data, its contribution to the Weighted-average F1 score will be appropriately scaled to reflect this imbalance. This weighting ensures that the final metric accurately represents the distribution of classes, offering a more balanced and realistic evaluation of the classifier's performance under the varying conditions within our dataset.

The complete pseudocode of the conversational sentiment analysis on the Couple Dialogue dataset is provided in Appendix C.

Table 6.3: *Examples of few-shot prompting for sentiment analysis in dialogues with Llama 2, presented in both Turkish and English. The primary task utilizes the Turkish version, with English translations provided for accessibility and clarity.*

Turkish Version	**English Version**
Görev: Son konuşmanın duygusal tonunu sadece "pozitif", "negatif", ya da "nötr" olarak değerlendir. Cevabın tek bir kelime olmasına dikkat et. Konuşmacı cinsiyeti ve diyalog dinamiklerinin önemini göz önünde bulundur. Sen yardımcı bir asistansın. Talimatları dikkate al. 'None' öncesi konuşmaların olmadığını gösterir.	**Task**: Evaluate the emotional tone of the last conversation as "positive", "negative", or "neutral". Ensure your response is a single word. Consider the importance of speaker gender and dialogue dynamics. You are a helpful assistant. Follow the given instructions. 'None' indicates no prior conversations.
Örnek 1 Diyalog: Konuşma 1: [Kadın: Bugün çok güzel bir gün geçirdik.] Konuşma 2: [Erkek: Evet, harikaydı. Seninle olmak her zaman çok güzel.] **En Sonki Konuşmanın Beklenen Duygu Tonu**: pozitif	**Example 1** Dialogue: Conversation 1: [Female: We had a really nice day today.] Conversation 2: [Male: Yes, it was wonderful. Being with you is always great.] **Expected Emotional Tone of the Last Conversation**: positive
Örnek 2 Diyalog: Konuşma 1: [Erkek: Dün akşamki tartışma için üzgünüm.] Konuşma 2: [Kadın: Ben de. Bir daha olmasın.] **En Sonki Konuşmanın Beklenen Duygu Tonu**: nötr	**Example 2** Dialogue: Conversation 1: [Male: I'm sorry about the argument last night.] Conversation 2: [Female: Me too. Let's not let it happen again.] **Expected Emotional Tone of the Last Conversation**: neutral
Örnek 3 Diyalog: Konuşma 1: [Kadın: Seni hiç anlamıyorum!] Konuşma 2: [Erkek: Bunu nasıl söyleyebilirsin?] **En Sonki Konuşmanın Beklenen Duygu Tonu**: negatif	**Example 3** Dialogue: Conversation 1: [Female: I just don't understand you!] Conversation 2: [Male: How can you say that?] **Expected Emotional Tone of the Last Conversation**: negative
Yeni Diyalog Diyalog: Konuşma 1: [Kadın: Gerçekten sinirliyim, sen başlayabilirsin.] Konuşma 2: [Erkek: Ne başlayacağım?] Konuşma 3: [Kadın: Hıh!] Konuşma 4: [Erkek: Hani ne sinirleniyorsun?] Konuşma 5: [Kadın: Arkadaşlar konusuna sen direkt başlayabilirsin bence.] **En Sonki Konuşmanın Beklenen Duygu Tonu**:	**New Dialogue** Dialogue: Conversation 1: [Female: I'm really annoyed, you start.] Conversation 2: [Male: Start what?] Conversation 3: [Female: Hmph!] Conversation 4: [Male: What are you annoyed about?] Conversation 5: [Female: I think you should start talking about friends directly.] **Expected Emotional Tone of the Last Conversation**:

Table 6.4: *Example of a JSON Data Structure: This table outlines a typical JSON format used to represent dialogue data, detailing roles and their corresponding content keys and values for a conversational system.*

Role	Content Key	Content Value
system	content	system_prompt
user	content	new_dialogue
assistant	content	expected_sentiment

Table 6.5: *Example of a JSON structure used in the fine-tuning process for sentiment analysis, presented in both Turkish and English. The table illustrates how dialogue data is structured, including roles such as "system," "user," and "assistant," to help the model learn the relationship between dialogue context, speaker states, and sentiment.*

Turkish Version	English Version
"system": {"content": "Son konuşmanın duygusal tonunu sadece “pozitif”, “negatif”, ya da “nötr” olarak değerlendir. Cevabın tek bir kelime olmasına dikkat et. Konuşmacı cinsiyeti ve diyalog dinamiklerinin önemini göz önünde bulundur."}, **"user"**: {"content": "[Kadın: Gerçekten sinirliyim, sen başlayabilirsin.] [Erkek: Ne başlayacağım?] [Kadın: Hıh!] [Erkek: Hani ne sinirleniyorsun?] [Kadın: Arkadaşlar konusuna sen direkt başlayabilirsin bence.]"}, **"assistant"**: {"content": "negatif"}	**"system"**: {"content": "Evaluate the emotional tone of the last conversation as “positive”, “negative”, or “neutral”. Ensure your response is a single word. Consider the importance of speaker gender and the dynamics of the dialogue."}, **"user"**: {"content": "[Female: I’m really annoyed, you can start.] [Male: Start what?] [Female: Hmph!] [Male: What are you annoyed about?] [Female: I think you should start talking about friends directly.]"}, **"assistant"**: {"content": "negative"}

Table 6.6: *Fine-Tuning Parameters for the Trendyol LLM Based on Llama 2: Detailed specifications of the parameters used in the fine-tuning process of the Trendyol LLM, which is built upon the Llama 2 framework. These parameters were customized for optimizing performance on the Couple Dialogue dataset used in this study.*

Parameter	Value
Quantization Type	4-bit ('nf4' double quantization)
Compute dtype	bfloat16
LoRA Alpha	128
Dropout Rate	0.05
LoRA Rank	64
Optimizer	AdamW with torch fused operations
Learning Rate	1e-6 (constant)
Epochs	3
Micro Batch Size	2
Maximum Sequence Length	4096 tokens
Evaluation Frequency	Every 25 steps
Total Training Duration	47 hours

Table 6.7: *Detailed Hyperparameter Settings for DialogueRNN Variants: This table provides the hyperparameters for different embedding technologies—GloVe, BERT Variants, and GPT—used within the DialogueRNN framework.*

Parameter	GloVe	BERT-Variants	GPT
Optimizer	AdamW	AdamW	AdamW
Loss Function	Categorical Cross-Entropy		
Batch Size	1	1	1
Output Size (D_G, D_P, D_E)	100	200	200

Table 6.8: *Training Times for Different Embedding Types for DialogueRNN architecture.*

Embedding Type	K-Fold Cross Validation Time	Standard Split Time
Pre-trained Turkish BERT	93 hours	17 hours
Fine-tuned Turkish BERT	41 hours	9 hours
Text-Embedding-3-Small	43 hours	6 hours
GloVe	47 hours	7.5 hours

7. EXPERIMENTAL STUDIES AND RESULTS

7.1 Non-Contextual Classifier Results

7.1.1 Conventional Classifier Results

The tables in this section display the average Weighted F1 scores, derived from two testing folds within a 10-fold cross-validation framework. Each cross-validation cycle involved segmenting 118 couple dialogues into seven training, one validation, and two testing folds. The results reported here represent the average of the Weighted F1 scores across these testing folds.

Table 7.1, Table 7.2, Table 7.3, and Table 7.4 present the classification performances on data prepared by three different types of preprocessing modules when the utterances are considered independent of each other.

7.1.1.1 BoW

Table 7.1 displays the Weighted F1 scores of the BoW when we applied several machine learning algorithms on three different data forms. We observe a general trend of improvement in performance when moving from the Root form + Detailed Preprocessing data form to Partial surface form + Partial Preprocessing for the BoW model feature extractor.

As couples used informal language to address issues about their marriage, the inclusion of stopwords and punctuation, including exclamation marks and ellipses, contributed to the sentiment classification task. Each classifier exhibited a notable positive performance impact upon incorporating stopwords and specified punctuations. Furthermore, flagging a word when it includes one of the morphemes carrying pity, sentimental meaning, wishes, or negation morphemes facilitated the utterance label classification task, as observed in the Partial surface form + Partial Preprocessing row in Table 7.1 in general.

Table 7.1: *Non-Contextual Classification Performance Using BoW Embeddings: This table displays the Weighted F1 scores (%) for different non-contextual classification methods utilizing BoW embeddings processed through various preprocessing methods. The scores represent averages from two testing folds in a 10-fold cross-validation, highlighting how each preprocessing variant impacts the classification score.*

Algorithms	Preprocessing Modules	Weighted F1
SVM	Root form + Detailed Preprocessing	47.62
	Root form + Partial Preprocessing	54.01
	Partial surface form + Partial Preprocessing	54.63
Decision Tree	Root form + Detailed Preprocessing	44.37
	Root form + Partial Preprocessing	48.42
	Partial surface form + Partial Preprocessing	49.61
Random Forest	Root form + Detailed Preprocessing	48.14
	Root form + Partial Preprocessing	49.92
	Partial surface form + Partial Preprocessing	50.02
Gradient Boosting	Root form + Detailed Preprocessing	41.49
	Root form + Partial Preprocessing	49.13
	Partial surface form + Partial Preprocessing	51.01
XGBoost	Root form + Detailed Preprocessing	47.1
	Root form + Partial Preprocessing	54.03
	Partial surface form + Partial Preprocessing	**55.01**
Multi Layer Perceptron	Root form + Detailed Preprocessing	47.09
	Root form + Partial Preprocessing	52.78
	Partial surface form + Partial Preprocessing	53.32

7.1.1.2 TF-IDF

On the other hand, the inclusion of certain morpheme forms did not always yield a positive impact on TF-IDF word embeddings, as is evident in Table 7.2, in terms of Weighted F1 performance. Nevertheless, it is noteworthy that the incorporation of stop words and specific punctuations consistently demonstrated a significant improvement in classification scores for the utterance features extracted by the TF-IDF model.

Table 7.2: *Non-Contextual Classification Performance Using TF-IDF Embeddings: This table presents the Weighted F1 scores (%) for various non-contextual classification methods utilizing TF-IDF feature embeddings. Scores are averaged from two testing folds within a 10-fold cross-validation, demonstrating the impact of different preprocessing approaches on classification performance.*

Algorithms	Preprocessing Modules	Weighted F1
SVM	Root form + Detailed Preprocessing	46.84
	Root form + Partial Preprocessing	51.91
	Partial surface form + Partial Preprocessing	**53.42**
Decision Tree	Root form + Detailed Preprocessing	47.31
	Root form + Partial Preprocessing	50.35
	Partial surface form + Partial Preprocessing	50.39
Random Forest	Root form + Detailed Preprocessing	48.11
	Root form + Partial Preprocessing	49.71
	Partial surface form + Partial Preprocessing	51.16
Gradient Boosting	Root form + Detailed Preprocessing	43.41
	Root form + Partial Preprocessing	51.04
	Partial surface form + Partial Preprocessing	50.09
XGBoost	Root form + Detailed Preprocessing	45.12
	Root form + Partial Preprocessing	53.14
	Partial surface form + Partial Preprocessing	52.69
Multi Layer Perceptron	Root form + Detailed Preprocessing	47.85
	Root form + Partial Preprocessing	51.89
	Partial surface form + Partial Preprocessing	51.07

7.1.1.3 Word2Vec

Word2Vec is recognized for its ability to understand and represent the semantic relationships between words. This neural network-based technique significantly surpasses the limitations of both BoW and TF-IDF by capitalizing on contextual information. TF-IDF addresses BoW's drawback of treating all words equally by assigning higher weights to less common words, thereby emphasizing their potential importance in text analysis. Improvements in classification performance are not consistently seen across classifiers when transitioning from traditional methods like BoW to more advanced embeddings such as Word2Vec. Although Word2Vec offers significant semantic enhancements, it does not always surpass the performance of simpler models. For example, as indicated in Table 7.1, traditional BoW features sometimes outperform both TF-IDF and Word2Vec in setups involving SVM (54.63%), XGBoost (55.01%), and Multi-Layer Perceptron (53.32%).

Remarkably, XGBoost consistently benefits from Word2Vec embeddings, as evidenced by an F1 score of 54.31%. This suggests that XGBoost is particularly adept at utilizing the semantic depth offered by Word2Vec, likely due to its capability to manage complex data interactions more effectively. The preprocessing approaches—ranging from detailed to partial processing, which includes the preservation of contextual nuances through morphemes—may further enhance this performance, enabling XGBoost to leverage these nuanced semantic signals optimally.

The classification effectiveness across all models generally increases with less stringent preprocessing, as detailed in Table 7.3. For example, the Decision Tree classifier's F1 score rises from 45.53% under detailed preprocessing to 51.04% with partial preprocessing in root forms, though it slightly drops to 49.91% when partial surface forms are used. This trend underscores the potential of moderate preprocessing to enhance generalizability and feature capture by classifiers.

Table 7.3: *Non-Contextual Classification Performance Using Word2Vec Embeddings: This table displays the weighted F1 scores (%) for various non-contextual classification methods utilizing Word2Vec embeddings. Scores are averaged from two testing folds within a 10-fold cross-validation, illustrating the effect of different preprocessing methods on classification efficacy.*

Algorithms	Preprocessing Modules	Weighted F1
SVM	Root form + Detailed Preprocessing	46.32
	Root form + Partial Preprocessing	48.21
	Partial surface form + Partial Preprocessing	53.66
Decision Tree	Root form + Detailed Preprocessing	45.53
	Root form + Partial Preprocessing	51.04
	Partial surface form + Partial Preprocessing	49.91
Random Forest	Root form + Detailed Preprocessing	46.31
	Root form + Partial Preprocessing	50.45
	Partial surface form + Partial Preprocessing	52.92
Gradient Boosting	Root form + Detailed Preprocessing	45.92
	Root form + Partial Preprocessing	50.03
	Partial surface form + Partial Preprocessing	51.34
XGBoost	Root form + Detailed Preprocessing	45.53
	Root form + Partial Preprocessing	50.14
	Partial surface form + Partial Preprocessing	**54.31**
Multi Layer Perceptron	Root form + Detailed Preprocessing	43.87
	Root form + Partial Preprocessing	50.52
	Partial surface form + Partial Preprocessing	52.37

7.1.1.4 GloVe

While the GloVe model is based on analyzing word-to-word co-occurrence statistics from the entire corpus, Word2Vec focuses on co-occurrence within the local context. We expected the pre-trained GloVe embeddings to yield better Weighted F1 scores due to its more advanced modeling approach. However, when we applied the same classification algorithms to the utterance features generated with these embeddings, the results were similar to those obtained with Word2Vec, contrary to our expectations, as shown in Table 7.4.

7.1.1.5 A Paired T-Test Study of Morpheme Effects on Sentiment Analysis:

The analysis presented in Table 7.1, Table 7.2, Table 7.3, and Table 7.4 illustrates a consistent trend of performance enhancement from the Root form + Detailed Preprocessing to Partial surface form + Partial Preprocessing, which integrates various morpheme forms. This transition showed an increase of up to 9.89% in the Weighted F1 score for the XGBoost classifier using the GloVe model. To verify the significance of this increase, we conducted paired t-tests between the most restrictive preprocessing, Root form + Detailed Preprocessing, and the more inclusive, Partial surface form + Partial Preprocessing. This latter form retains more morphological and syntactic information, potentially providing richer features for classifiers to utilize.

This statistical method is crucial for substantiating performance improvement claims by assessing whether observed differences in F1 scores are statistically significant or merely due to random variations. The paired t-test, conducted on the same sample sets, allows for a meticulous comparison of means by considering the variance within their differences. This provides a precise measure of whether performance discrepancies are coincidental or reflect a genuine effect. We set the significance threshold at $p < 0.05$, applying the tests to the average scores from two testing folds in a 10-fold cross-validation

Table 7.4: *Non-Contextual Classification Performance Using GloVe Embeddings: This table displays the Weighted F1 scores (%) for various non-contextual classification methods utilizing GloVe embeddings. Scores are averaged from two testing folds within a 10-fold cross-validation, illustrating the impact of different preprocessing methods on classification efficacy.*

Algorithms	Preprocessing Modules	Weighted F1
SVM	Root form + Detailed Preprocessing	47.11
	Root form + Partial Preprocessing	47.93
	Partial surface form + Partial Preprocessing	53.13
Decision Tree	Root form + Detailed Preprocessing	46.41
	Root form + Partial Preprocessing	49.43
	Partial surface form + Partial Preprocessing	49.42
Random Forest	Root form + Detailed Preprocessing	46.01
	Root form + Partial Preprocessing	50.76
	Partial surface form + Partial Preprocessing	53.23
Gradient Boosting	Root form + Detailed Preprocessing	47.31
	Root form + Partial Preprocessing	50.71
	Partial surface form + Partial Preprocessing	51.82
XGBoost	Root form + Detailed Preprocessing	44.83
	Root form + Partial Preprocessing	51.69
	Partial surface form + Partial Preprocessing	**54.72**
Multi Layer Perceptron	Root form + Detailed Preprocessing	45.16
	Root form + Partial Preprocessing	50.1
	Partial surface form + Partial Preprocessing	52.33

setup. This approach rigorously determines the significance of the performance differences between the compared models.

In evaluating sentiment classification performance across different embedding models, we strategically reduced the number of paired t-tests. This reduction is driven by the need to maintain statistical integrity and optimize resource use, focusing on the most substantial differences. Specific classifier and preprocessing combinations were selected based on their potential to demonstrate significant performance disparities. Multi-Layer Perceptron and XGBoost were chosen for their pronounced improvements when including morphological features and transitioning from more detailed to partial preprocessing settings. This selection is instrumental in showcasing the impact of preprocessing within the Non-Contextual Model framework, as detailed in Table 7.5.

The results from the paired t-tests evaluating the improvements in Weighted F1 scores when moving from Root form + Detailed Preprocessing to Partial surface form + Partial Preprocessing reveal a significant trend: less restrictive preprocessing, which preserves more morphological and syntactic features, generally results in substantial performance improvements in sentiment analysis models. This observation holds across most classifier and embedding combinations, where improvements are marked by significant p-values ($p < 0.05$), indicating strong enhancements.

The results show that the most substantial improvement was observed with GloVe embeddings using XGBoost, where the "1st vs 3rd" comparison yielded a significant 9.89% increase in Weighted F1 scores ($p = 0.013$), demonstrating the effectiveness of this preprocessing strategy. However, some experiments, such as those involving TF-IDF embeddings with MLP ($p = 0.059$) and GloVe embeddings with MLP ($p = 0.053$), marginally met or exceeded the accepted p-value threshold of 0.05, indicating that the statistical significance was either borderline or not achieved. These findings suggest that the efficacy of preprocessing strategies may vary depending on the unique characteristics of

the embedding models and their interaction with classifiers. Notably, the improvements with MLP using BoW embeddings were particularly significant ($p = 0.0024$), highlighting the advantage of incorporating a broader linguistic context in preprocessing.

Table 7.5: *Paired T-Test Results: This table compares the Weighted F1 scores of classifiers (MLP, XGBoost) across different preprocessing techniques. "1st vs 3rd" refers to the comparison between "Root form + Detailed Preprocessing" (1st) and "Partial surface form + Partial Preprocessing" (3rd), highlighting the impact of morpheme inclusion on sentiment analysis performance. The **F1 Diff (%)** shows the performance improvement, and the **T-Test** column indicates the statistical significance, with p-values below 0.05 considered significant. Abbreviations: **RF-DP** = Root Form + Detailed Preprocessing, **PS-PP** = Partial Surface Form + Partial Preprocessing.*

Embed.	**Class.**	**Comp.**	**RF-DP**	**PS-PP**	**F1 Diff (%)**	**T-Test**
TF-IDF	MLP	1st vs 3rd	47.85	51.07	+3.22	0.059
GloVe	MLP	1st vs 3rd	45.16	52.33	+7.17	0.053
TF-IDF	XGBoost	1st vs 3rd	45.12	52.69	+7.57	0.041
BoW	MLP	1st vs 3rd	47.09	53.32	+6.23	0.0024
BoW	XGBoost	1st vs 3rd	47.1	55.01	+7.91	0.023
Word2Vec	XGBoost	1st vs 3rd	45.53	54.31	+8.78	0.027
Word2Vec	MLP	1st vs 3rd	43.87	52.37	+8.50	0.0049
GloVe	**XGBoost**	**1st vs 3rd**	**44.83**	**54.72**	**+9.89**	**0.013**

7.1.1.6 Pre-trained Turkish BERT

In Table 7.6, we present the performance of supervised methods on pre-trained Turkish BERT utterance embeddings when ignoring the contextual, speaker, and interdependency of utterances. Both stopwords and punctuation marks may provide context about the sentiment of the subject; therefore, we did not eliminate them. We only performed basic preprocessing tasks for data preparation. As seen, we obtained the best sentiment classification score with the Multi-Layer Perceptron, which achieved 56.84%. When we compare the results in Table 7.6 with the results in Table 7.1, Table 7.2, Table 7.3, and Table 7.4, the Weighted F1 score of SVM, Decision Tree, Random Forest,

Table 7.6: *BERT Embeddings Performance: Weighted F1 scores (%) for classifiers using BERT embeddings after basic preprocessing are shown. Scores from two testing folds in a 10-fold cross-validation demonstrate superior performance over traditional embeddings, with the Multi-Layer Perceptron achieving the highest at 56.84%.*

Classifier	**Weighted F1**
SVM	55.71
Decision Tree	54.12
Random Forest	55.49
Gradient Boosting	53.91
XGBoost	53.70
Multi-Layer Perceptron	**56.84**

and Gradient Boosting classifiers also outperformed the scores obtained using BoW, TF-IDF, Word2Vec, and GloVe feature embeddings on the Non-Contextual Model.

In our Introduction section, we emphasized the exceptional performance of BERT-based utterance features over other vector embedding models within both non-contextual and contextual frameworks. To substantiate this for the non-contextual component, we conducted paired t-tests, specifically examining the weighted F1 scores obtained using the Multi-Layer Perceptron (MLP) across different embedding models. The MLP was selected due to its consistently superior performance when utilizing pre-trained Turkish BERT embeddings, making it an ideal candidate for this analysis.

The results of paired t-tests provided compelling statistical evidence of BERT's superiority. The tests, comparing the performance of BERT to that of Word2Vec, BoW, and TF-IDF under identical preprocessing configurations, yielded p-values of 0.0025, 0.0015, and 0.041 respectively. Each of these values falls below the conventional significance threshold of 0.05, strongly supporting the superior capability of BERT in enhancing MLP's ability to classify sentiment accurately and underscoring its effectiveness in capturing complex semantic relationships within the dataset.

However, the comparison between BERT and GloVe, conducted using the 'Partial surface form + Partial Preprocessing' setup for GloVe and the basic preprocessing

form for BERT, resulted in a p-value of 0.057. This value, slightly above the standard significance threshold of 0.05, indicates a non-significant difference in performance under these differing conditions. BERT, processed with basic preprocessing, achieved an F1 score of 56.84%, while GloVe, under the more detailed 'Partial surface form + Partial Preprocessing,' scored 52.33%. It's important to note that a p-value just above 0.05 does not conclusively indicate no performance difference between BERT and GloVe; rather, it suggests that the evidence is not robust enough to definitively reject the hypothesis of equal performance under the test conditions. This outcome implies that while the performance gap narrows with the specific preprocessing applied to GloVe, the difference does not reach statistical significance. This suggests that GloVe's performance relative to BERT might be competitive under this detailed preprocessing but is not sufficient to claim equivalence or superiority.

Several factors underscore the superiority of BERT over traditional models. While TF-IDF and BoW lack contextual understanding, and GloVe and Word2Vec offer only a basic level of context comprehension, BERT stands out for its ability to capture nuanced meanings across diverse contexts. BERT is built on the Transformer architecture, enabling it to learn intricate patterns and representations. It effectively handles out-of-vocabulary words and comprehends word morphemes, a capability absent in traditional models like TF-IDF and BoW. BERT embeddings not only capture semantic relationships between words but also encode syntactic information such as sentence structure and grammar, contributing to a deeper understanding of sentence meaning compared to Word2Vec. Given that BERT is pre-trained on an extensive dataset and has a larger number of parameters, it can capture effectively a broad spectrum of language patterns and knowledge.

7.1.2 Deep Learning Model Results

Table 7.7 displays the performance of a deep neural network architecture using pre-trained GloVe and pre-trained Turkish BERT model feature vectors. As explained and illustrated in the Non-Contextual Classifier section, the generated model is simplified. In contrast, DialogueRNN, generated using three GRUs to track individual speaker states, global context, and sentiment state during the conversation, introduces increased model complexity. Our intention was to observe and compare the success rates of our GloVe and pre-trained Turkish BERT embeddings when utilized as input in a simpler model, prior to being trained with a more complex one such as DialogueRNN.

The deep learning model, particularly the architecture designed with Keras that integrates BERT embeddings, demonstrates a distinct advantage as seen in Table 7.7. In the deep learning setup, BERT's Weighted F1 score reaches 60.31%, outperforming GloVe's 55.73%. This superior performance highlights BERT's effectiveness in leveraging contextual clues and polysemy inherent in language, which is crucial for parsing the subtleties of sentiment in dialogue. The architecture, which includes a BERT layer that processes utterance embeddings with attention mechanisms to emphasize relevant contextual information, coupled with layers like GlobalMaxPooling1D and Dense layers, optimizes the neural network's ability to discern and categorize sentiment more accurately than simpler embedding techniques such as GloVe.

On the other hand, when BERT embeddings are utilized within traditional machine learning classifiers, as shown in Table 7.6, the results are robust but do not reach the performance levels observed with deep learning models. The highest score among these traditional classifiers is observed in the Multi-Layer Perceptron (MLP), achieving a Weighted F1 of 56.84%. Although MLP benefits from the rich representations provided by BERT, it lacks the complex sequential processing capabilities that deep learning models possess, which are crucial for effectively handling the dynamics of language used in

Table 7.7: *GloVe vs. pre-trained Turkish BERT Embeddings Performance: This table compares Weighted F1 scores (%) for sentiment prediction using GloVe and BERT in the Non-Contextual Model. Scores are averaged from two testing folds in a 10-fold cross-validation, emphasizing BERT's superior ability to handle word polysemy and sequence context.*

Feature	Weighted F1
GloVe	55.73
BERT	**60.31**

couple dialogues. A paired t-test comparing the performance of deep learning models incorporating pre-trained Turkish BERT embeddings against MLP, which also utilized the pre-trained Turkish BERT model, yielded a significant result (p-value = 0.019), affirming the superiority of deep learning approaches under these testing conditions.

7.2 Contextual Classifier Results

7.2.1 DialogueRNN Results

As previously outlined, we utilized a dual evaluation method for the DialogueRNN algorithm, comprising k-fold cross-validation and a standard training-validation-test split. Table 7.8 presents the 10-fold cross-validation results, with Weighted F1 scores averaged from two testing folds.

The DialogueRNN model incorporates various embeddings, including GloVe, pre-trained Turkish BERT, and text-embedding-3-small, each adapted for use in the DialogueRNN framework. These are respectively named GloVe-DialogueRNN, GPT-DialogueRNN, and Turkish BERT-DialogueRNN. Additionally, we have integrated embeddings from a version of the pre-trained Turkish BERT that has been specifically fine-tuned on Couple Dialogue data. This version is referred to as "Turkish BERT-DialogueRNN (Couple Dialogue Fine-tuned)".

7.2.1.1 Embedding Effect

Table 7.8 shows that the integration of various embeddings in the DialogueRNN framework—specifically GloVe, GPT, and Turkish BERT—reveals distinct influences on sentiment analysis performance. GloVe-DialogueRNN demonstrated a lower Weighted F1 score of 52.36%, highlighting its limitations in capturing the dynamic nuances necessary for effective conversational sentiment analysis due to its context-independent nature. In contrast, GPT-DialogueRNN, which utilizes embeddings generated from a broad spectrum of internet text via the OpenAI API, outperformed Turkish BERT-DialogueRNN with a score of 63.14%. This suggests that GPT's generative pre-training provides a more comprehensive language understanding, beneficial for processing the varied linguistic expressions found in couple dialogues.

Contextual embeddings like GPT and Turkish BERT significantly outperformed the less advanced GloVe model, underscoring the importance of context sensitivity in embeddings. These embeddings dynamically adjust their interpretations based on the surrounding text, offering a more nuanced understanding of dialogue intricacies. It is important to note that the paired t-test conducted between the Turkish BERT-DialogueRNN (Couple Dialogue Fine-tuned) and the GloVe-DialogueRNN revealed a statistically significant improvement (p=0.05), with a notable difference of 14.46% in performance. Notably, the fine-tuned Turkish BERT-DialogueRNN (Couple Dialogue Fine-tuned) showcased superior performance with the highest F1 score of 66.82%, illustrating the profound impact of domain-specific fine-tuning. By aligning the model more closely with the specific language patterns and sentiment expressions within the Couple Dialogue dataset, fine-tuning enhances the model's ability to interpret complex emotional cues effectively.

The DialogueRNN's ability to model inter-speaker dependencies and track conversational states is significantly enhanced by the advanced capabilities of these embeddings. This synthesis not only boosts the performance of sentiment classification tasks but also

demonstrates the potential of combining sophisticated neural architectures with highly contextualized embeddings. The success of Turkish BERT-DialogueRNN, particularly after fine-tuning, underscores the benefit of embedding models that incorporate a deep understanding of both semantic and syntactic elements, proving essential in the evolving field of conversational AI.

7.2.1.2 Context Effect

The DialogueRNN model consistently outperforms Non-contextual Model experiments in conversational sentiment analysis, as evidenced by the comparative results in the provided tables (Table 7.8 versus Table 7.6 and Table 7.7). This advantage can primarily be attributed to DialogueRNN's ability to incorporate and dynamically update contextual information, speaker state, and previous sentiment. By maintaining stateful information across dialogues, DialogueRNN captures the flow and evolution of conversations more effectively than static, non-contextual models. This is particularly crucial in settings like couple dialogues, where the emotional undertone and sentiment can shift rapidly and are often influenced by the interaction history between speakers. The Contextual Model's superior performance underscores the importance of tailored architectures that align closely with the inherent structure of dialogic data, enabling a deeper understanding of sentiment dynamics within conversations.

We conducted paired t-tests to determine the significance of the performance improvements of the DialogueRNN model over the Non-Contextual Model experiments, as shown in Tables 7.6 and Table 7.7. These tests were applied specifically to comparisons where the DialogueRNN model scores were higher. The first t-test compared the performance between the Turkish BERT-DialogueRNN (Couple Dialogue Fine-tuned) from the Contextual Model and the Multi-Layer Perceptron using pre-trained Turkish BERT from the Non-Contextual Model. The second t-test compared the Turkish BERT-DialogueRNN

Table 7.8: *Performance Comparison of Embedding Types with DialogueRNN: Weighted F1 scores (%) for various embeddings. Turkish BERT fine-tuned on Couple Dialogue achieves the best performance.*

Embedding Type	Weighted F1
GloVe-DialogueRNN	52.36
Turkish BERT-DialogueRNN	62.16
GPT-DialogueRNN	63.14
Turkish BERT-DialogueRNN (Couple Dialogue Fine-tuned)	**66.82**

(Couple Dialogue Fine-tuned) from the Contextual Model to the deep learning model using pre-trained Turkish BERT from the Non-Contextual Model. The results of these tests yielded p-values of 0.0012 and 0.041, respectively, indicating statistically significant improvements.

7.2.2 LLM Model Results

For the computationally demanding LLMs, k-fold cross-validation was not feasible. Instead, we used a dataset split of 70% for training, 10% for validation, and 20% for testing to ensure consistent evaluation conditions. Table 7.9 presents the Weighted F1 scores from testing each LLM on the same test data, showcasing their performance across different setups.

7.2.2.1 Few-shot Prompting vs. Instruction-based Prompting in Llama 2

Table 7.9 demonstrates that Llama 2 with Few-shot Prompting outperforms its Instruction-based Prompting counterpart in conversational sentiment analysis on Couple Dialogue data, achieving a Weighted F1 score of 40.03% compared to 34.49%. This enhancement can be attributed to the few-shot model's ability to leverage multiple example interactions to better understand the nuances of the task as illustrated in Table 6.3. Few-shot prompting effectively contextualizes the model's responses by providing it with clear examples of the desired output, helping it to adapt to the subtleties of positive, negative, and neutral emotional tones in conversations.

7.2.2.2 Effectiveness of Fine-tuning Llama 2

As demonstrated in Table 7.9, fine-tuning Llama 2 results in superior performance, achieving a Weighted F1 score of 54.67%, compared to 40.03% with few-shot prompting. Fine-tuning allows the model to adjust its neural network weights to better reflect the specific linguistic and sentiment patterns of the Couple Dialogue dataset. While prompt engineering directs the model's focus, fine-tuning reshapes its understanding, inherently making it more aligned with the dataset's unique characteristics. This adjustment promotes a deeper and more accurate recognition of sentiment cues, surpassing the surface-level comprehension achieved through prompt engineering.

7.2.2.3 Performance of Llama 2 on Turkish Data

As discussed earlier, all Llama 2 tasks, including prompting and fine-tuning, have been conducted using the Trendyol LLM. This model has been fine-tuned with a substantial bilingual dataset that includes 180,000 instruction sets in both English and Turkish. Despite these efforts, the model's performance in sentiment analysis tasks for Turkish couple dialogues, as shown in Table 7.9, remained suboptimal. This might indicate that the proportion of Turkish examples in the training set does not adequately represent the linguistic diversity and complexity needed for our specific application. Although technically equipped to handle Turkish, the nuances and idiosyncrasies of conversational Turkish as manifested in couple dialogues may not be fully captured. The base Llama 2 model's predominance of English data training could still overshadow its capabilities in Turkish, leading to a performance gap when the language context is primarily Turkish.

7.2.2.4 Limitations of GPT-4's Prompt Engineering

GPT-4's performance in prompt engineering might seem underwhelming as shown in Table 7.9, particularly due to its exclusive reliance on instruction-based prompting,

which lacks the richer contextual understanding provided by few-shot scenarios. Furthermore, if GPT-4's training included limited exposure to Turkish data, it would further impair its capability to process effectively and respond to the complexities of Turkish-language interactions. The absence of domain-specific fine-tuning also restricts the model's ability to adjust its learned behaviors to the specific nuances of the sentiment analysis task within the Turkish Couple Dialogue dataset.

Notably, despite GPT-3.5 Turbo being technologically less advanced than GPT-4, its superior performance post-fine-tuning highlights the importance of targeted training. We opted not to use few-shot prompting for GPT-4 primarily due to budget constraints. The increased token requirements for adding few-shot examples would have tripled our costs with OpenAI, which was not feasible within our project budget. This limitation prevented us from leveraging the full potential of few-shot prompting, which might have enhanced GPT-4's performance in our specific application.

7.2.2.5 Superiority of GPT-3.5 Turbo Fine-tuning

The performance metrics presented in Table 7.9 highlight GPT-3.5 Turbo's superior efficacy in fine-tuning tasks compared to Llama 2, underscoring potential advantages in its architecture that are conducive to handling complex linguistic data, such as Turkish conversational sentiment analysis. Despite the lack of explicit architectural details due to proprietary constraints, the results from fine-tuning GPT-3.5 Turbo suggest that its underlying structure is responsive optimally to nuanced language features. This is likely due to the model's extensive pre-training across a diverse range of multilingual datasets, enabling it to acquire a substantial foundational understanding of language variations. When this broad base is fine-tuned to specific datasets, such as the Turkish Couple Dialogue, GPT-3.5 Turbo is able to adapt its learned patterns more effectively, enhancing its capacity to discern and process the intricate emotional dynamics within the dialogue.

This adaptability and depth of learning, inferred from its performance, contribute to its notable success in this context, distinguishing it from Llama 2.

7.2.2.6 Success of Fine-Tuned Pre-Trained Turkish BERT

In the sentiment analysis of Turkish couple dialogues, as depicted in Table 7.9, the fine-tuned Pre-Trained Turkish BERT achieved the highest Weighted F1 score of 65.61%, surpassing both GPT-3.5 Turbo and Llama 2. This superior performance can be attributed to BERT's architecture, which is particularly adept at capturing the context within sentences and leveraging bidirectional training—distinct advantages when processing the complexities of conversational data. Unlike GPT models, which are trained unidirectionally and may not capture preceding context as effectively, BERT's bidirectional approach allows for a more nuanced understanding of the sentence structure and sentiment, adapting more precisely to the intricate language patterns found in Turkish dialogues. Additionally, the fine-tuning process on Turkish couple dialogue data further hones this model's ability to discern subtle emotional cues and linguistic intricacies unique to the dataset. This specialized adaptation to the linguistic context of the dialogue, combined with the inherent strengths of the BERT architecture, likely contributed to its outstanding performance, making it exceptionally well-suited for tasks requiring deep linguistic understanding and context sensitivity.

7.2.2.7 The Impact of Contextual Modeling on Sentiment Analysis Accuracy in LLMs

The contrasting outcomes from the application of LLMs in contextual versus non-contextual frameworks for conversational sentiment analysis on Couple Dialogue data clearly demonstrate the significance of context in enhancing model performance. As detailed in Table 7.7 and Table 7.9, the Fine-Tuned Pre-Trained Turkish BERT achieved the highest Weighted F1 score of 65.61% in the Contextual Model setup, surpassing all

models in the non-contextual framework, where the highest score was 60.31% by BERT.

This significant enhancement is attributed to the Contextual Model's capability to utilize the immediate content and the broader conversational context, including historical interactions and speaker-specific nuances. These results underscore the superiority of contextual awareness in boosting the accuracy of sentiment analysis models. Although direct comparison is limited due to the use of different evaluation methods—fixed data splits for LLMs versus k-fold cross-validation for non-LLMs—and computational constraints that restricted the application of k-fold cross-validation for LLMs, the top-performing LLMs exhibit robust effectiveness. Specifically, even when utilizing only 20% of the data for testing during fine-tuning, models like the Fine-Tuned Pre-Trained Turkish BERT significantly outperformed other models. This highlights their potential under more extensive and rigorous testing frameworks. While some LLMs like the Llama 2 Fine-Tuning did not surpass Non-Contextual Model approaches, the overall higher scores among leading LLMs advocate for the benefits of contextual models in enhancing sentiment analysis. Further evaluations with statistical methodologies could substantiate these findings more comprehensively, yet the current results strongly advocate for the use of contextual models in enhancing sentiment analysis.

7.2.2.8 Evaluating the Efficacy of Fine-Tuning vs. Prompt-Based Approaches in Sentiment Analysis

Table 7.9 demonstrates that the fine-tuning approach significantly outperformed the prompt-based approach in our conversational sentiment analysis task, achieving an impressive improvement of up to 31.12% in the Weighted F1 score. Specifically, while the Llama 2 with Instruction-based Prompting yielded a Weighted F1 score of 34.49%, the score escalated to 65.61% with the Fine-Tuned Pre-Trained Turkish BERT. A paired t-test was conducted to determine the statistical significance of this change, resulting in a p-value of 0.037, under the standard threshold of 0.05. This confirms that the observed

improvement is statistically significant, validating the superior performance of the fine-tuning approach over prompt-based methods in enhancing sentiment analysis accuracy.

7.2.2.9 Superiority of DialogueRNN Using Fine-tuned Turkish BERT Embeddings

Table 7.10 presents the results of all Contextual Model approaches, including DialogueRNN, tested on a data split comprising 70% training, 10% validation, and 20% testing of the Couple Dialogue dataset. This setup facilitates direct comparisons between DialogueRNN and various LLM models.

As illustrated in Table 7.10, the DialogueRNN model that integrates embeddings from the fine-tuned Turkish BERT, specifically customized for our Couple Dialogue dataset, achieved the highest performance with a Weighted F1 score of 66.17%. This configuration, labeled as Turkish BERT-DialogueRNN (Couple Dialogue Fine-tuned), leverages the fine-tuning process to closely align the model with the linguistic characteristics and sentiment nuances specific to our conversational data. The fine-tuning process adapts the pre-trained embeddings to better capture the contextual subtleties and emotional variances within the dialogue, offering more precise understanding and classification of sentiment. This is contrasted with the Turkish BERT-DialogueRNN, which uses general pre-trained embeddings and achieved a lower score of 61.36%, indicating the significant impact of domain-specific fine-tuning. Moreover, when compared to the Fine-Tuned Pre-Trained Turkish BERT (65.61%) and GPT-3.5 Turbo fine-tuning (65.42%), it is evident that embedding the fine-tuned model within the DialogueRNN framework enhances its capability to process and analyze the conversational dynamics more effectively, resulting in superior sentiment analysis performance.

The DialogueRNN framework is specifically designed to handle the complexities of sequential conversational data, effectively modeling the flow and dependencies of

dialogue. By integrating fine-tuned embeddings, the DialogueRNN can leverage its architectural strengths to better utilize the contextual cues and speaker-specific information, which are crucial for accurate sentiment analysis in dialogue systems.

7.2.2.10 Statistical Significance and Confidence Interval Analysis

Based on the statistical analysis and the results obtained in our study, we can confidently conclude that the contributions we outlined in the introduction are both valid and significant as shown in Table 7.11. The comparison between Non-Contextual and Contextual Models clearly demonstrates the superiority of Contextual Model, with a statistically significant improvement of 9.98% in Weighted F1 scores when employing DialogueRNN with fine-tuned Turkish BERT embeddings. Similarly, the fine-tuned approach outperformed instruction-based prompting with a notable 31.12% increase in Weighted F1 score, further validated by a paired t-test. Our investigation into the integration of morphological features also revealed significant enhancements in classification performance, with improvements of up to 9.89% in Weighted F1 scores, particularly in the Non-Contextual Model where BERT embeddings showed an increase of 7.42%. Lastly, the BERT-based utterance features consistently outperformed other embedding models across both Non-Contextual and Contextual frameworks, with a marked 14.46% improvement in the Contextual Model. These findings are supported by robust statistical evidence, as reflected in the calculated t-values and confidence intervals, which provide a range within which the true difference in performance is likely to fall, offering additional assurance of the reliability of these findings.

Table 7.9: *LLM Sentiment Analysis Performance: This table displays Weighted F1 scores (%) for different LLMs using methods like instruction-based prompting, few-shot prompting, and fine-tuning on Turkish couple dialogues. The results emphasize the superior accuracy of the Fine-Tuned Pre-Trained Turkish BERT, highlighting the benefits of language-specific tuning.*

LLM Model	**Weighted F1**
Llama 2 with Instruction-based Prompting	34.49
Llama 2 with Few-shot Prompting	40.03
GPT-4 with Prompt Engineering	48.92
Llama 2 Fine-Tuning	54.67
GPT-3.5 Turbo with Fine-Tuning	65.42
Fine-Tuned Pre-Trained Turkish BERT	**65.61**

Table 7.10: *Performance of Contextual Models on Couple Dialogue Dataset: This table displays the Weighted F1 scores for various Contextual Model algorithms tested with a standard dataset split of 70% training, 10% validation, and 20% testing. It highlights the comparative effectiveness of DialogueRNN and various LLMs such as Llama 2, GPT-4, and Turkish BERT, with detailed performance metrics for different implementation strategies like prompting, fine-tuning, and DialogueRNN adaptations.*

Contextual Model	**Weighted F1**
Llama 2 with Instruction-based Prompting	34.49
Llama 2 with Few-shot Prompting	40.03
GPT-4 with Prompt Engineering	48.92
GloVe-DialogueRNN	52.01
Llama 2 Fine-Tuning	54.67
Turkish BERT-DialogueRNN	61.36
GPT-DialogueRNN	62.84
GPT 3.5 Turbo with Fine-Tuning	65.42
Fine-Tuned Pre-Trained Turkish BERT	65.61
Turkish BERT-DialogueRNN (Couple Dialogue Fine-tuned)	**66.17**

Table 7.11: *The statistical analysis of the Weighted F1 scores (%) across different settings, focusing on the significance of performance improvements. Comparisons are made between Non-Contextual and Contextual Models, as well as different preprocessing techniques. The F1 Difference (%) reflects the improvement in performance, while the T-Test Value assesses statistical significance, with p-values below 0.05 considered significant. Confidence Intervals (CI) are calculated based on a 95% confidence level. Abbreviations used: "DNN" refers to a Deep Neural Network architecture incorporating Pre-trained Turkish BERT, "IBP" refers to Instruction-Based Prompting, "W. F1" refers to Weighted F1, and "Diff." refers to Difference.*

Model 1	**Model 1 W. F1**	**Model 2**	**Model 2 W. F1**	**F1 Diff. (%)**	**T-Test Val.**	**CI**
Decision Tree (Partial Surface + Partial Preprocessing, GloVe)	49.42	MLP (Pre-trained Turkish BERT)	56.84	+7.42	0.0023	[3.43, 11.41]
XGBoost (Root Form + Detailed Preprocessing, GloVe)	44.83	XGBoost (Partial Surface + Partial Preprocessing, GloVe)	54.72	+9.89	0.013	[2.64, 17.14]
MLP (Pre-trained Turkish BERT)	56.84	Turkish BERT-DialogueRNN (Couple Dialogue Fine-tuned)	66.82	+9.98	0.0012	[5.13, 14.83]
DNN (Pre-trained Turkish BERT)	60.31	Turkish BERT-DialogueRNN (Couple Dialogue Fine-tuned)	66.82	+6.51	0.041	[0.33, 12.69]
Llama 2 (IBP)	34.49	Fine-Tuned Turkish BERT	65.61	+31.12	0.037	[2.34, 59.90]
GloVe-DialogueRNN	52.36	Turkish BERT-DialogueRNN (Couple Dialogue Fine-tuned)	66.82	+14.46	0.0032	[6.24, 22.68]

8. DISCUSSION

At the start of this study, we defined several research questions to answer through our experiments. In this section, we propose an answer to each question and present possible options for future work. The research questions defined were the following:

1. **Does the Contextual model consistently outperform the Non-Contextual model on the Couple Dialogue data?**

   Our comparative analysis between contextual and non-contextual models highlights a consistent trend: contextual models demonstrate superior performance in conversational sentiment analysis, particularly when applied to the Couple Dialogue dataset. This is reflected in the significant improvement in Weighted F1 scores, where models like the fine-tuned Turkish BERT-DialogueRNN excel. These contextual models effectively capture the emotional subtleties and dynamics present in couple interactions by leveraging context, speaker roles, and sentiment evolution over time. The results clearly emphasize that contextual models are not only advantageous but essential for accurately analyzing dialogue data, offering a more comprehensive and precise understanding of conversational intricacies.

2. **What is the most efficient prompt structure that would be understandable for both Llama 2 and GPT-3.5 Turbo in the context of conversational sentiment analysis on the Couple Dialogue data? (This question arises because, to the best of our knowledge, a suitable prompt structure for this task has not been documented in the literature.)**

   Analyzing the effectiveness of prompts for Llama 2 and GPT-3.5 Turbo in the conversational sentiment analysis of Couple Dialogue data, we observed that Few-shot Prompting outperforms Instruction-based Prompting. Llama 2, in particular, achieves better results with Few-shot Prompting by leveraging examples to identify

and classify emotional tones within dialogues. Consequently, the most effective prompt structure includes multiple annotated dialogue examples, offering concrete illustrations that enable the models to distinguish sentiments such as positive, negative, and neutral. This strategy enhances the models' ability to grasp conversational dynamics and speaker roles more accurately, thereby improving the overall performance of sentiment analysis.

3. **Can our conversational sentiment analysis task, which is based on the Turkish language, be effectively achieved by Llama 2 and GPT-3.5 Turbo, both of which are primarily pre-trained in English?**

   Even when fine-tuned as the Trendyol LLM on bilingual data, Llama 2 struggles to accurately capture the nuances of Turkish conversational sentiment analysis. On the other hand, GPT-3.5 Turbo demonstrates a stronger understanding and adaptability to Turkish, likely attributed to its extensive pre-training across a diverse range of languages. Fine-tuning GPT-3.5 Turbo on the specialized Turkish Couple Dialogue dataset further enhances its capability to detect the subtle emotional nuances inherent in the language, making it notably more effective for sentiment analysis in this context. Therefore, while both models show promise, GPT-3.5 Turbo proves to be better suited for Turkish sentiment analysis due to its greater adaptability and capacity for nuanced learning.

4. **How much improvement do fine-tuned models offer over prompt-based approaches for this conversational sentiment analysis task?**

   Fine-tuned models demonstrate a clear advantage over prompt-based methods in conversational sentiment analysis, particularly for linguistically complex tasks such as analyzing Turkish couple dialogues. The primary reason is that fine-tuning allows models to adapt to the specific linguistic features and contextual intricacies of

the target dataset. In contrast, prompt-based approaches often rely on generic inputs that may fail to capture the full scope of conversational dynamics. Fine-tuned models, however, leverage the entire conversational context, enabling them to effectively recognize and interpret subtle shifts in tone and sentiment. This deeper understanding of dialogue patterns results in significantly more accurate sentiment analysis.

5. **Do contextual embeddings outperform non-contextual vector representations like BoW and TF-IDF, as well as word embeddings with minimal contextual understanding such as GloVe and Word2Vec, for tasks involving Couple Dialogue data?**

   Contextual embeddings from models like BERT and GPT consistently outperform non-contextual representations, such as BoW, TF-IDF, Word2Vec, and GloVe, in sentiment analysis of Couple Dialogue data. BERT's Transformer-based architecture is particularly adept at capturing sentence-level context and subtle linguistic nuances, which non-contextual methods often fail to grasp. These embeddings excel at managing complex language structures and intricate patterns, which are critical for accurately interpreting sentiments in couple conversations. Furthermore, BERT embeddings seamlessly integrate both syntactic and semantic information, enabling a more in-depth and precise sentiment analysis compared to traditional vector-based approaches.

6. **What are the effects of preprocessing and morphological analysis on a language with rich morphology, like Turkish, in enhancing sentiment labeling success rates?**

Preprocessing and morphological analysis play a crucial role in enhancing sentiment labeling accuracy for morphologically rich languages like Turkish. Adjustments in preprocessing, combined with the inclusion of morphological details, greatly improve the performance of sentiment analysis classifiers. Retaining stopwords and key punctuation marks allows classifiers to capture the informal nuances and emotional expressions often found in conversational Turkish, particularly in couple dialogues. The use of morpheme-specific flags that highlight emotional, sentimental, or negational components within words enables models to detect subtle linguistic cues, thereby enhancing sentiment classification accuracy. This tailored preprocessing approach preserves essential linguistic features and aligns with the natural usage of the language, ensuring a more accurate and nuanced sentiment analysis. Overall, this underscores the importance of adopting customized preprocessing techniques to address the challenges of morphologically complex languages, ultimately leading to improved model performance and deeper insights into language patterns.

7. **Does the use of a more complex neural network architecture in the Non-Contextual model help close the performance gap with the Contextual model?**

   Leveraging complex neural network architectures in non-contextual models significantly narrows the performance gap with contextual models, particularly in conversational sentiment analysis. Advanced architectures, which incorporate BERT embeddings and attention mechanisms, enhance the model's ability to process linguistic nuances more effectively by capturing contextual cues and addressing polysemy—factors critical for accurate sentiment interpretation. These sophisticated designs, utilizing components such as GlobalMaxPooling1D and Dense layers,

demonstrate superior capability in detecting subtle emotional variations within dialogues compared to simpler approaches, leading to notable improvements in sentiment classification accuracy. Nonetheless, despite these advancements, non-contextual models still do not match the effectiveness of fully contextual frameworks like DialogueRNN. Designed specifically to handle conversational dynamics, DialogueRNN excels by tracking speaker-specific states and evolving dialogue contexts, enabling a more profound understanding of interactions and achieving greater accuracy in sentiment analysis.

9. CONCLUSIONS

This study's investigation into the "Couple Dialogue" dataset, a groundbreaking effort in Turkish conversational sentiment analysis, highlights the vital importance of contextual modeling for unraveling the intricacies of human interactions. The comparison between Non-Contextual and Contextual Models demonstrates that contextual approaches, particularly those utilizing advanced LLMs like BERT, GPT-3.5 Turbo, Llama 2, and GPT-4, significantly improve sentiment analysis accuracy. By integrating cutting-edge techniques and architectures, such as DialogueRNN, which efficiently captures the relational and sequential dynamics of conversations, our research achieved a 9.98% improvement in Weighted F1 scores compared to traditional Non-Contextual Model approaches. This achievement not only underscores the capability of Contextual Models to capture emotional transitions and subtle nuances within dialogues but also establishes a strong foundation for future sentiment analysis research in linguistically complex contexts.

Thorough testing and analysis across different modeling frameworks further confirm that contextual embeddings dramatically outperform non-contextual representations like BoW, TF-IDF, and basic word embeddings, including GloVe and Word2Vec. These results reinforce the superiority of contextual approaches in handling intricate language structures and subtle emotional patterns within the Turkish Couple Dialogue dataset. Our findings emphasize the pivotal role of fine-tuning and carefully designed prompt structures in enhancing the effectiveness of LLMs, enabling them to capture nuanced sentiment expressions in couple interactions with greater accuracy. Consequently, the Contextual Model not only consistently outperformed its Non-Contextual counterpart but also set a new benchmark for sentiment analysis in complex conversational datasets.

9.1 Challenges in Conversational Sentiment Analysis in Couple Dialogue data

Analyzing emotional dynamics through conversational sentiment analysis poses significant challenges, requiring advanced approaches to model conversational contexts effectively. Our dataset, which focuses on Turkish couple interactions, cannot be directly compared with other datasets using the DialogueRNN architecture. Translating this dataset into English fails to preserve its original meaning due to the complexity and length of the utterances. As highlighted in [112] and reinforced by recent research, issues such as translation accuracy, cultural nuances, and idiomatic expressions create substantial obstacles. These challenges include the loss of subtle linguistic cues and culturally embedded contexts that machine translation systems frequently fail to capture, resulting in reduced accuracy in sentiment analysis.

Idioms and colloquial expressions, which are prominent in informal couple dialogues, often lack direct equivalents in other languages. This makes accurate sentiment interpretation across linguistic boundaries particularly difficult and demands more sophisticated techniques. Furthermore, the absence of comparable studies on conversational sentiment analysis in Turkish has left us without established benchmarks for performance evaluation. One of the key challenges we encountered was the model's difficulty in detecting sarcasm within our Couple Dialogue dataset, given its informal nature and focus on intimate, interpersonal interactions. This highlights the need for tailored methods capable of handling the unique dynamics and intricacies of Turkish conversational data.

In Table 9.1 and Table 9.2, we present the utterance predictions from a script derived from the 20% testing set, following a standard dataset split of 70% for training, 10% for validation, and 20% for testing. The script includes dialogues with a sarcastic theme, which are labeled as "0" (negative) by human annotators.

Our analysis of the Couple Dialogue dataset reveals that the Turkish BERT - DialogueRNN (Couple Dialogue Fine-tuned) model exhibited the highest performance in terms of Weighted F1 score, as detailed in Table 7.10. Despite its overall effectiveness in sentiment analysis, this model, along with other contextual model approaches, faced challenges in accurately classifying sarcastic expressions. These challenges are evident in Table 9.1 and Table 9.2, where the nuanced and context-dependent nature of sarcasm becomes apparent. Our experimental results indicate that GPT-based models, despite their general effectiveness, struggled with accurately classifying sarcastic expressions in the Turkish language. This is evident from the performance metrics and examples provided in Table 9.1 and Table 9.2. The nuanced and context-dependent nature of sarcasm presents a significant challenge, and the models showed limited success in this regard. This observation highlights a gap in the current capabilities of these models to fully interpret the subtle cues necessary for sarcasm detection, which often depends heavily on cultural and contextual insights beyond the literal text.

9.2 Comprehensive Strategies for Enhancing Sarcasm Detection

The difficulty of detecting sarcasm using text-only models highlights the need to incorporate additional modalities, such as audio data, to more effectively capture the full range of communicative cues. Elements like vocal tone and intonation play a crucial role in expressing sarcasm, and their inclusion can significantly improve model performance. To address this, we have initiated the collection of audio data from couple interactions, with plans to integrate it into our current models. This multimodal approach will enhance the models' ability to identify sarcasm more accurately.

Furthermore, our dataset is enriched with detailed annotations for both sentiments and emotional tones, providing a strong basis for training more advanced models. By leveraging these annotations, the models can better interpret the emotional context of

individual utterances, enabling them to capture not only the literal content but also the underlying emotional subtext. This contextual awareness is critical for improving sarcasm detection.

Looking ahead, further progress in sarcasm detection within the complex linguistic framework of Turkish couple dialogues can be achieved through advanced Large Language Model (LLM) techniques. Models such as BERT and GPT, which are pre-trained on vast multilingual datasets, will be fine-tuned using task-specific datasets containing examples of sarcasm. Approaches like transfer learning will adapt these pre-trained models for the Turkish language, while strategies such as few-shot learning will address the challenge of detecting sarcasm with limited annotated examples. By combining these advanced LLM techniques with multimodal inputs, we expect significant improvements in the models' ability to detect sarcasm. This will, in turn, enhance the overall accuracy and reliability of sentiment analysis in conversational settings.

9.3 Confusion Matrix and Positive Sample Prediction Analysis

Below Table 9.3 is the confusion matrix for the Fine-Tuned Pre-Trained Turkish BERT model tested on the Couple Dialogue Dataset, employing a standard dataset split of 70% training, 10% validation, and 20% testing:

Table 9.3: *Confusion Matrix for Fine-Tuned Turkish BERT Model.*

True Labels / Predicted Labels	Negative	Neutral	Positive
Negative	1300	300	40
Neutral	300	720	5
Positive	170	240	130

9.3.1 True Positive Analysis

- **Negative Class:**

- Total Instances: $1300 + 300 + 40 = 1640$
- Correctly Predicted (True Positives): 1300
- Accuracy: $\frac{1300}{1640} \approx 79.27\%$

- **Neutral Class:**
  - Total Instances: $300 + 720 + 5 = 1025$
  - Correctly Predicted (True Positives): 720
  - Accuracy: $\frac{720}{1025} \approx 70.24\%$

- **Positive Class:**
  - Total Instances: $170 + 240 + 130 = 540$
  - Correctly Predicted (True Positives): 130
  - Accuracy: $\frac{130}{540} \approx 24.07\%$

From this analysis, it is evident that positive samples are the most challenging to predict accurately, while the model performs relatively well on negative and neutral classes. Other contextual models tested on the Couple Dialogue Dataset exhibited similar patterns, with better performance on negative and neutral classes but poor accuracy for positive samples. This suggests that the issue lies not with the specific model but with the inherent challenges posed by the dataset and sentiment dynamics.

9.3.2 Challenges in Predicting Positive Samples

9.3.2.1 Imbalanced Dataset

The dataset includes 6,161 negative, 6,050 neutral, and 2,083 positive samples, leading to a significant imbalance. This disparity biases the model toward the majority classes (negative and neutral), as it has fewer examples to learn the unique features of

positive samples. As a result, the model struggles to generalize effectively for the under-represented positive class.

9.3.2.2 Positive Sentiment Complexity

Positive sentiment is often subtle and context-dependent, with less explicit linguistic cues compared to negative or neutral sentiments. For instance:

- **Turkish Example:** *"Harika, yine çok düzenlisin!"*
- **English Translation:** *"Great, you're so organized again!"*

This sentence might seem positive but could carry sarcasm, making it hard to classify correctly.

9.3.2.3 Sarcasm and Ambiguity

Sarcasm is prevalent in the dataset, particularly within emotionally charged conversations. Text-only models often fail to detect sarcastic tones due to the absence of vocal or tonal cues. For example:

- **Turkish Example:** *"Tabii, bu da harika bir karar oldu!"*
- **English Translation:** *"Of course, this was such a great decision!"*

Despite positive wording, the tone and context suggest sarcasm or negativity.

9.3.2.4 Context Dependencies

Positive sentiments frequently depend on conversational context. For instance:

- **Turkish Example:** *"Bu konuda konuştuğumuz için mutluyum."*
- **English Translation:** *"I'm glad we talked about this."*

The sentiment may only be interpreted as positive if it follows a resolved conflict.

9.3.2.5 Insufficient Linguistic Cues

Positive utterances often lack explicit markers, unlike negative or neutral statements, which frequently include clear linguistic indicators of sentiment. For example:

- **Turkish Example:** *"Saat 5'te görüşürüz."*
- **English Translation:** *"See you at 5 PM."*

This factual statement is easy to classify as neutral, while positive sentiments often rely on subtler emotional expressions.

9.3.3 Proposed Solutions for Improvement

9.3.3.1 Class Rebalancing

Techniques such as oversampling (e.g., SMOTE) for positive samples or undersampling for negative and neutral classes can mitigate the bias toward majority classes, allowing the model to better learn positive sentiment patterns.

9.3.3.2 Multimodal Data Integration

Incorporating audio data, such as tone, pitch, and intonation, can aid in detecting sarcasm and emotional nuances. This multimodal approach would enhance the model's ability to differentiate between genuine positivity and sarcasm.

9.3.3.3 Synthetic Data Generation

Using advanced LLMs like GPT-4 to generate synthetic positive samples can address dataset imbalance and expose the model to a wider variety of positive sentiment expressions. These synthetic examples can reflect the complexity and conversational dynamics of real-world scenarios.

9.4 Advanced LLM Applications in Real-Time Sentiment Analysis and Practical Use Cases

Our research underscores the pivotal role played by advancedLLMs like GPT-3.5 Turbo, Llama 2, BERT, and GPT-4 in enhancing the accuracy and depth of sentiment analysis across various domains. These models have notably improved digital assistants' capabilities in customer service, allowing them to quickly interpret and respond to intricate customer emotions with greater empathy and precision. For example, the integration of LLMs into customer service bots within the telecommunications sector has led to reduced response times and improved customer satisfaction through contextually aware and emotionally intelligent interactions.

Real-time sentiment analysis holds tremendous potential, particularly within customer support systems, where it can significantly enhance user experiences. Incorporating contextual sentiment analysis models into live platforms allows virtual agents to deliver immediate and contextually relevant responses. This proves particularly advantageous for high-traffic customer service centers that need to manage large volumes of interactions efficiently. Real-time sentiment detection can also route customer queries to the appropriate departments or specialists, thus improving operational efficiency and elevating customer satisfaction levels. Research conducted by Zhu [65] on smart assistants with emotion recognition highlights the importance of incorporating emotional intelligence in user interactions, aligning with our findings that real-time sentiment analysis enables more responsive and personalized service delivery.

Nevertheless, implementing these models in real-time scenarios introduces significant challenges. These include the requirement for a robust computational infrastructure and effective data streaming mechanisms to manage large, fast-moving datasets. To deliver timely and accurate responses, systems must be capable of instantaneous data processing and analysis, necessitating infrastructure that guarantees stability, scalability, and

high performance while managing dynamic data flows.

In this context, our Couple Dialogue dataset proves invaluable as a resource for testing and validating sentiment analysis models under conditions that closely mimic real-world scenarios. Although the data were collected in a controlled laboratory setting, the captured conversations are highly naturalistic, as they reflect genuine, unscripted marital discussions on personally significant, contentious topics. These interactions, sourced from 103 recently married heterosexual couples in Turkey, showcase the real dynamics of couple dialogues, including overlapping speech, informal expressions, and unstructured conversation patterns.

While the lab environment introduces some constraints, the authenticity of the conversations, coupled with precise transcription and annotation, ensures the dataset closely resembles the complexity and unpredictability of real-life communication. This positions the Couple Dialogue dataset as a robust, though not flawless, benchmark for assessing sentiment analysis models' ability to handle conversational subtleties. It also highlights the challenges these models may face in truly noisy, real-world conditions, where the variability and intricacies of human communication are even more pronounced.

9.5 Cross-Domain Integration for Advanced Healthcare Sentiment Analysis

Our upcoming initiative focuses on utilizing the advanced sentiment analysis capabilities of state-of-the-art LLMs, including Llama 3.1 and GPT-4o. GPT-4o, an optimized and faster version of GPT-4, has demonstrated exceptional performance across benchmarks such as basic mathematics, language comprehension, and vision understanding, as noted by OpenAI [1]. Concurrently, Meta's Llama 3.1, recently introduced [2], brings enhancements such as an increased context window, improved tool usage capabilities,

[1] https://openai.com/index/hello-gpt-4o/
[2] https://huggingface.co/meta-llama/Meta-Llama-3.1-8B

and strengthened multilingual support. Our project seeks to implement these advanced models in the healthcare domain, specifically for analyzing psychiatrist-patient therapy sessions targeting conditions like depression and Alzheimer's disease.

Building on the sentiment detection methodologies developed for the "Couple Dialogue" dataset, this project aims to track emotional variations throughout therapy sessions. This analysis will assist psychiatrists in assessing emotional shifts, thereby providing valuable insights into therapeutic progress. We have obtained ethical permissions for the collection and analysis of this data, which will be conducted in Turkish at a public hospital in Turkey. After completing the data collection phase, our refined Contextual Model approaches will be applied to extract meaningful insights, with the detailed results set to be published in future research.

Additionally, we see significant opportunities to integrate the psychiatric session data with our existing Couple Dialogue dataset. Given that both datasets are in Turkish, our current models can be seamlessly adapted to this new domain, offering an enriched training process and improving the detection of intricate emotional dynamics across diverse conversational contexts. This cross-domain application will test the adaptability and robustness of our models in identifying emotional states within mental health evaluations.

This initiative not only demonstrates the versatility of our sentiment analysis models but also extends their application into critical mental health settings. By facilitating a more nuanced understanding of emotional transitions during therapy, our work has the potential to support therapeutic interventions more effectively. Ultimately, these advancements will enhance the precision and reliability of our models, making them powerful tools in clinical psychology and psychiatry. Our goal is to provide actionable insights that could influence therapeutic strategies and significantly improve patient care outcomes.

9.6 Exploring Cross-Linguistic Models: A Multifaceted Approach

9.6.1 Challenges and Opportunities

Building on our proven success in sentiment analysis using contextual embeddings specifically tailored for Turkish, we aim to broaden the scope of our research by exploring cross-linguistic models. These models are trained on vast multilingual datasets, enabling sentiment analysis without requiring language-specific models. While this approach enhances their applicability for global platforms such as customer service systems and social media analysis, it also introduces several challenges. These include cultural variations in expressing sentiment, ambiguities in word meanings, and syntactic differences across languages, all of which can hinder a model's ability to accurately comprehend and analyze context.

9.6.2 Exploring Multilingual Model Performance and Future Directions for Enhanced Cross-Linguistic Understanding

Although advanced models such as LLaMA 2, GPT-4, and GPT-3.5 Turbo possess cross-linguistic capabilities, they are primarily optimized for English. This focus limits their ability to effectively capture the unique linguistic characteristics of other languages like Turkish. As observed in Table 7.10, Turkish BERT, pre-trained exclusively on Turkish data, demonstrates superior performance for Turkish-specific tasks. For instance, while fine-tuning generalist models like GPT-3.5 Turbo on the Couple Dialogue dataset improved their performance (achieving a score of 65.42), these models still underperformed compared to Turkish BERT, emphasizing the advantage of pre-training on language-specific datasets.

To overcome these challenges, we plan to fine-tune our Couple Dialogue dataset on multilingual models like mBERT [113] and XLM-R [114]. These models are designed for multilingual processing and are better equipped to address Turkish's structural

complexities. Our future efforts will focus on evaluating whether these multilingual models can surpass the limitations of generalist LLMs and achieve enhanced performance in cross-linguistic sentiment analysis, particularly within the Turkish context.

9.6.3 Synthesizing Multilingual Data Sources

To further strengthen our sentiment analysis framework, we propose combining the Turkish Couple Dialogue dataset with an English dataset of couple dialogues. This integration will enable us to fine-tune advanced models such as LLaMA 3.1, optimizing their performance across both languages and enhancing their ability to capture the unique linguistic and emotional nuances inherent in each.

One significant challenge lies in accessing open-source datasets specifically focused on dyadic couple dialogues. To address this, we have reached out to various authors for access to couples therapy session data. Additionally, we plan to leverage the IEMOCAP dataset [7], a widely used open-source resource featuring emotional dyadic exchanges through acted dialogues. These interactions closely mimic real-world couple conversations, making them an ideal complement to our dataset.

To address the scarcity of real-world data, we will also employ the latest generative models like GPT-4o. Building on the success of GPT-4 in generating synthetic data, these advancements will enable us to augment our existing datasets, bridging gaps in data availability and ensuring more comprehensive training for our models.

Integrating datasets from Turkish and English offers multiple advantages. The rich morphological structure of Turkish provides a strong contrast to the simpler syntactic patterns of English, creating a valuable comparative framework for cross-linguistic research. Furthermore, English datasets benefit from a wealth of benchmarks and resources, enhancing model training and facilitating smoother cross-linguistic transfer. By fine-tuning cutting-edge pre-trained models such as GPT-4o and LLaMA 3.1, we will

expand the training landscape, ensuring these models gain versatility and robustness in processing and generating diverse languages. This approach significantly improves their performance in sentiment analysis tasks across multilingual contexts, demonstrating their adaptability to the complexities of human conversation.

Table 9.1: *Examples of sarcasm from the Couple Dialogue dataset: predictions from models utilizing Prompting and Baseline approaches (IBP, FSP, PE, GloVe-DRNN). Sentiment shifts are annotated with "2" for positive, "0" for negative, and "1" for neutral.*

Speaker	**Utterance (Turkish and English Translation)**	**Annotated Sentiment**	**Llama 2 IBP**	**Llama 2 FSP**	**GPT-4 PE**	**GloVe-DRNN**
F	Oh, çamaşırlar kendi kendine makineye girecek mi sanıyorsun? Ben atmasam bir yere gitmezler! (Oh, do you think the laundry will just walk itself to the washer? If I don't put them in, they're not going anywhere!)	0	1	1	1	1
F	Neden sadece kadınların işi gibi ev işleri, hep oturup kraliçe gibi yaşamıyor muyum zaten? (Why are house chores like only a woman's job? Am I not already living like a queen sitting around?)	0	2	2	1	2
M	Evlilikte mutluluk mu? Ha, evet, herkes için ama öncelikle senin için tabii. (Marriage and happiness? Ha, yes, for everyone, but first and foremost for you, of course.)	0	2	1	1	1
F	Tabii, çamaşır dolabını her gün selamlayıp konuşuyorum, sen de görebilirsin aslında! (Of course, I greet and talk to the laundry cabinet every day, you could see it too if you tried!)	0	2	2	2	2
M	Çoraplar bitti, belki ayaklarım donar, belki o zaman anlarsın. (Socks are finished, maybe my feet will freeze, maybe then you'll understand.)	0	1	0	0	0
M	Çamaşır sepetine at, tabii canım, sanki başka işim yokmuş gibi. (Put it in the laundry basket, sure dear, as if I have nothing else to do.)	0	1	1	1	1
F	Çamaşır makinesine atamazsan büyük bir trajedi olacak, hayat duracak. (If you don't put them in the washing machine, it'll be a great tragedy, life will stop.)	0	1	0	0	1

Table 9.2: *Examples of sarcasm from the Couple Dialogue dataset: predictions from models utilizing Fine-Tuning and DialogueRNN approaches. Sentiment shifts are annotated with "2" for positive, "0" for negative, and "1" for neutral.*

Speaker	Utterance (Turkish and English Translation)	Annotated Sentiment	Llama 2 FT	Turkish BERT-DRNN	GPT-DRNN	GPT 3.5 Turbo FT	Fine-Tuned Pre-Trained Turkish BERT
F	Oh, çamaşırlar kendi kendine makineye girecek mi sanıyorsun? Ben atmasam bir yere gitmezler! (Oh, do you think the laundry will just walk itself to the washer? If I don't put them in, they're not going anywhere!)	0	1	1	1	0	0
F	Neden sadece kadınların işi gibi ev işleri, hep oturup kraliçe gibi yaşamıyor muyum zaten? (Why are house chores like only a woman's job? Am I not already living like a queen sitting around?)	0	1	0	1	1	0
M	Evlilikte mutluluk mu? Ha, evet, herkes için ama öncelikle senin için tabii. (Marriage and happiness? Ha, yes, for everyone, but first and foremost for you, of course.)	0	1	1	0	1	1
F	Tabii, çamaşır dolabını her gün selamlayıp konuşuyorum, sen de görebilirsin aslında! (Of course, I greet and talk to the laundry cabinet every day, you could see it too if you tried!)	0	2	1	2	0	1
M	Çoraplar bitti, belki ayaklarım donar, belki o zaman anlarsın. (Socks are finished, maybe my feet will freeze, maybe then you'll understand.)	0	0	0	0	0	0
M	Çamaşır sepetine at, tabii canım, sanki başka işim yokmuş gibi. (Put it in the laundry basket, sure dear, as if I have nothing else to do.)	0	1	1	1	0	1
F	Çamaşır makinesine atamazsan büyük bir trajedi olacak, hayat duracak. (If you don't put them in the washing machine, it'll be a great tragedy, life will stop.)	0	0	0	0	0	0

REFERENCES

[1] M. W. Morris and D. Keltner, "How emotions work: The social functions of emotional expression in negotiations," *Research in Organizational Behavior*, vol. 22, pp. 1–50, 2000.

[2] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, pp. 1735–80, Dec. 1997.

[3] A. Vaswani, N. M. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. Neural Information Processing Systems*, 2017.

[4] M. Farhadloo and E. Rolland, "Multi-class sentiment analysis with clustering and score representation," in *2013 IEEE 13th International Conference on Data Mining Workshops*, pp. 904–912, 2013.

[5] S. I. Omurca, E. Ekinci, and H. Turkmen, "An annotated corpus for turkish sentiment analysis at sentence level," in *2017 International Artificial Intelligence and Data Processing Symposium (IDAP)*, pp. 1–5, 2017.

[6] S. Guha, A. Joshi, and V. Varma, "Siel: Aspect based sentiment analysis in reviews," in *Proc. of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pp. 759–766, Jun. 2015.

[7] C. Busso *et al.*, "Iemocap: Interactive emotional dyadic motion capture database," *Language Resources and Evaluation*, vol. 42, pp. 335–359, Dec. 2008.

[8] Y. Li *et al.*, "Dailydialog: A manually labelled multi-turn dialogue dataset," in *Proc. of the Eighth International Joint Conference on Natural Language Processing*, 2017.

[9] 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*, 2019.

[10] X. Wang, W. Shi, R. Kim, Y. Oh, S. Yang, J. Zhang, and Z. Yu, "Persuasion for good: Towards a personalized persuasive dialogue system for social good," *arXiv preprint arXiv:1906.06725*, 2020.

[11] Y. Zhang *et al.*, "Scenariosa: A publicly available conversational database for interactive sentiment analysis," *IEEE Access*, vol. 8, pp. 1–1, 2020.

[12] 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.

[13] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever, *et al.*, "Language models are unsupervised multitask learners," in *OpenAI Blog*, vol. 1, p. 9, 2019.

[14] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, *et al.*, "Language models are few-shot learners," in *Advances in Neural Information Processing Systems* (H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, eds.), vol. 33, pp. 1877–1901, Curran Associates, Inc., 2020.

[15] OpenAI *et al.*, "Gpt-4 technical report," *arXiv preprint arXiv:2303.08774*, 2024.

[16] H. Touvron *et al.*, "Llama 2: Open foundation and fine-tuned chat models," *arXiv preprint arXiv:2307.09288*, 2023.

[17] J. Sun, C. Zheng, E. Xie, Z. Liu, *et al.*, "A survey of reasoning with foundation models," *arXiv preprint arXiv:2312.11562*, 2024.

[18] J. Krugmann and J. Hartmann, "Sentiment analysis in the age of generative ai," *Customer Needs and Solutions*, vol. 11, Mar. 2024.

[19] A. Buscemi and D. Proverbio, "Chatgpt vs gemini vs llama on multilingual sentiment analysis," *arXiv preprint arXiv:2402.01715*, 2024.

[20] B. Zhang, H. Yang, and X.-Y. Liu, "Instruct-fingpt: Financial sentiment analysis by instruction tuning of general-purpose large language models," *arXiv preprint arXiv:2306.12659*, 2023.

[21] I. Carvalho, H. G. Oliveira, and C. Silva, "The importance of context for sentiment analysis in dialogues," *IEEE Access*, vol. 11, pp. 86088–86103, 2023.

[22] K. Kheiri and H. Karimi, "Sentimentgpt: Exploiting gpt for advanced sentiment analysis and its departure from current machine learning," *arXiv preprint arXiv:2307.10234*, 2023.

[23] J. Wei, X. Wang, D. Schuurmans, M. Bosma, *et al.*, "Chain-of-thought prompting elicits reasoning in large language models," *arXiv preprint arXiv:2201.11903*, 2023.

[24] Y. Fu, H. Peng, A. Sabharwal, P. Clark, and T. Khot, "Complexity-based prompting for multi-step reasoning," *arXiv preprint arXiv:2210.00720*, 2023.

[25] C. Xu, P. Li, W. Wang, H. Yang, S. Wang, and C. Xiao, "Cosplay: Concept set guided personalized dialogue generation across both party personas," in *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, ACM, 2022.

[26] S. Chen, Y. Hou, Y. Cui, W. Che, T. Liu, and X. Yu, "Recall and learn: Fine-tuning deep pretrained language models with less forgetting," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 7870–7881, Association for Computational Linguistics, Nov. 2020.

[27] C. Anil, Y. Wu, A. J. Andreassen, *et al.*, "Exploring length generalization in large language models," in *Advances in Neural Information Processing Systems*, 2022.

[28] J. Pennington, R. Socher, and C. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1532–1543, Oct. 2014.

[29] N. Majumder, S. Poria, D. Hazarika, R. Mihalcea, A. Gelbukh, and E. Cambria, "Dialoguernn: An attentive rnn for emotion detection in conversations," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, pp. 6818–6825, Jul. 2019.

[30] E. Kouloumpis, T. Wilson, and J. Moore, "Twitter sentiment analysis: The good the bad and the omg!," in *Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media (ICWSM)*, 2011.

[31] R. Moraes, J. F. Valiati, and W. P. Gavião Neto, "Document-level sentiment classification: An empirical comparison between svm and ann," *Expert Systems with Applications*, vol. 40, no. 2, pp. 621–633, 2013.

[32] C. Lin *et al.*, "Weakly supervised joint sentiment-topic detection from text," *IEEE Transactions on Knowledge and Data Engineering*, vol. 24, no. 6, pp. 1134–1145, 2012.

[33] T. Wilson, J. Wiebe, and P. Hoffmann, "Recognizing contextual polarity in phrase-level sentiment analysis," in *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, Oct. 2005.

[34] A. Zadeh, M. Chen, S. Poria, E. Cambria, and L.-P. Morency, "Tensor fusion network for multimodal sentiment analysis," *arXiv preprint arXiv:1707.07250*, 2017.

[35] W. L. Hamilton, K. Clark, J. Leskovec, and D. Jurafsky, "Inducing domain-specific sentiment lexicons from unlabeled corpora," *arXiv preprint arXiv:1606.02820*, 2016.

[36] D. G. Maynard and M. A. Greenwood, "Who cares about sarcastic tweets? investigating the impact of sarcasm on sentiment analysis," in *Language Resources and Evaluation Conference (LREC)*, Mar. 2014.

[37] H. P. N. C. A. C. N. Majumder, S. Poria and A. Gelbukh, "Sentiment and sarcasm classification with multitask learning," *IEEE Intelligent Systems*, vol. 34, no. 3, pp. 38–43, 2019.

[38] T. C. R. W. A. Caliskan, P. P. Ajay and M. R. Banaji, "Gender bias in word embeddings," in *Proceedings of the 2022 AAAI/ACM Conference on AI, Ethics, and Society*, Jul. 2022.

[39] M. Y. R. C. V. O. J. Zhao, T. Wang and K. W. Chang, "Gender bias in contextualized word embeddings," *arXiv preprint arXiv:1904.03310*, 2019. [Online].

[40] J. Howard and S. Ruder, "Universal language model fine-tuning for text classification," *arXiv preprint arXiv:1801.06146*, 2018.

[41] C. X. B. McCann, J. Bradbury and R. Socher, "Learned in translation: Contextualized word vectors," *arXiv preprint arXiv:1708.00107*, 2017.

[42] T. Thongtan and T. Phienthrakul, "Sentiment classification using document embeddings trained with cosine similarity," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, Jul. 2019.

[43] X. Xie, S. Ge, F. Hu, M. Xie, and N. Jiang, "An improved algorithm for sentiment analysis based on maximum entropy," *Soft Computing*, vol. 23, pp. 1–13, Jan. 2019.

[44] S. Poria *et al.*, "Context-dependent sentiment analysis in user-generated videos," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, pp. 873–883, Jan. 2017.

[45] 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 conference. Association for Computational Linguistics. North American Chapter. Meeting*, vol. 2018, pp. 2122–2132, Jun. 2018.

[46] 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*, Oct.–Nov. 2018.

[47] P. Zhong, D. Wang, and C. Miao, "Knowledge-enriched transformer for emotion detection in textual conversations," *arXiv preprint arXiv:1909.10681*, 2019.

[48] 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), Nov. 2019.

[49] Z. Lian, B. Liu, and J. Tao, "Ctnet: Conversational transformer network for emotion recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 985–1000, 2021.

[50] W. Jiao, M. R. Lyu, and I. King, "Real-time emotion recognition via attention gated hierarchical memory network," *arXiv preprint arXiv:1911.09075*, 2019.

[51] Y. Z. et al., "Learning interaction dynamics with an interactive lstm for conversational sentiment analysis," *Neural Networks*, vol. 133, pp. 40–56, 2021.

[52] Y. Z. et al., "A quantum-like multimodal network framework for modeling interaction dynamics in multiparty conversational sentiment analysis," *Information Fusion*, vol. 62, pp. 14–31, 2020.

[53] R. M. D. Ghosal, N. Majumder and S. Poria, "Utterance-level dialogue understanding: An empirical study," *arXiv preprint arXiv:2009.13902*, 2020.

[54] N. G. J. D. M. J. D. C. O. L. M. L. L. Z. Y. Liu, M. Ott and V. Stoyanov, "Roberta: A robustly optimized bert pretraining approach," *arXiv preprint arXiv:1907.11692*, 2019.

[55] Y. S. et al., "Conversational emotion recognition studies based on graph convolutional neural networks and a dependent syntactic analysis," *Neurocomputing*, vol. 501, pp. 629–639, 2022.

[56] W. L. et al., "Bieru: Bidirectional emotional recurrent unit for conversational sentiment analysis," *Neurocomputing*, vol. 467, pp. 73–82, 2022.

[57] G. T. et al., "Context- and sentiment-aware networks for emotion recognition in conversation," *IEEE Transactions on Artificial Intelligence*, vol. 3, no. 5, pp. 699–708, 2022.

[58] M. D. M. Abdul-Mageed and M. Korayem, "Subjectivity and sentiment analysis of modern standard arabic," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, (Portland, OR, USA), pp. 587–591, Jun. 2011.

[59] P. B. A. Joshi and R. Balamurali, "A fall-back strategy for sentiment analysis in hindi: a case study," in *Proceedings of the Conference*, Dec. 2010.

[60] H.-L. Yang and A. F. Y. Chao, "Sentiment analysis for chinese reviews of movies in multi-genre based on morpheme-based features and collocations," *Information Systems Frontiers*, vol. 17, pp. 1335–1352, Dec. 2015.

[61] N. Medagoda, "Sentiment analysis on morphologically rich languages: An artificial neural network (ann) approach," in *Advances in Computational Intelligence*, vol. 628, pp. 377–393, Feb. 2016.

[62] N. Medagoda, "Framework for sentiment classification for morphologically rich languages: A case study for sinhala," in *Proceedings of the Conference*, 2017.

[63] H. L. H. Zhang and C. Kim, "Semantic and instance segmentation in coastal urban spatial perception: A multi-task learning framework with an attention mechanism," *Sustainability*, vol. 16, no. 2, 2024.

[64] J. Y. X. Si, H. He and D. Ming, "Cross-subject emotion recognition brain-computer interface based on fnirs and dbjnet," *Cyborg and Bionic Systems*, vol. 4, Jul. 2023.

[65] C. Zhu, "Research on emotion recognition-based smart assistant system: Emotional intelligence and personalized services," *Journal of System and Management Sciences*, vol. 13, no. 5, pp. 227–242, 2023.

[66] P. L. X. L. X. L. X. Gu, X. Chen and Y. Du, "Simalstm-snp: novel semantic relatedness learning model preserving both siamese networks and membrane computing," *The Journal of Supercomputing*, vol. 80, pp. 3382–3411, Feb. 2024.

[67] C. R. Aydın and T. Güngör, "Sentiment analysis in turkish: Supervised, semi-supervised, and unsupervised techniques," *Natural Language Engineering*, vol. 27, no. 4, pp. 455–483, 2021.

[68] Y. S. R. Dehkharghani, B. Yanikoglu and K. Oflazer, "Sentiment analysis in turkish at different granularity levels," *Natural Language Engineering*, vol. 23, no. 4, pp. 535–559, 2017.

[69] K. Tohma and Y. Kutlu, "Challenges encountered in turkish natural language processing studies," *Natural and Engineering Sciences*, vol. 5, pp. 204–211, Nov. 2020.

[70] T. G. H. Sak and M. Saraclar, "Resources for turkish morphological processing," *Language Resources and Evaluation*, vol. 45, pp. 249–261, May 2011.

[71] E. Arısoy, *Statistical and discriminative language modelling for Turkish large vocabulary continuous speech recognition*. PhD thesis, Boğaziçi University, İstanbul, Turkey, 2009.

[72] G. Güler and A. C. Tantuğ, "Comparison of turkish word representations trained on different morphological forms," *arXiv preprint arXiv:2002.05417*, 2020.

[73] A. Köksal and A. Özgür, "Twitter dataset and evaluation of transformers for turkish sentiment analysis," in *2021 29th Signal Processing and Communications Applications Conference (SIU)*, pp. 1–4, 2021.

[74] O. Ozturk and A. Ozcan, "Sentiment analysis in turkish using transformer-based deep learning models," in *4th International Conference on Artificial Intelligence and Applied Mathematics in Engineering* (U. K. D. J. Hemanth, T. Yigit and U. Guvenc, eds.), pp. 1–15, Springer International Publishing, 2023.

[75] B. B. U. U. Acikalin and M. Kutlu, "Turkish sentiment analysis using bert," in *2020 28th Signal Processing and Communications Applications Conference (SIU)*, pp. 1–4, 2020.

[76] Z. A. Guven, "The comparison of language models with a novel text filtering approach for turkish sentiment analysis," *ACM Trans. Asian Low-Resour. Lang. Inf. Process.*, vol. 22, Dec. 2022.

[77] M. M. Mutlu and A. Özgür, "A dataset and bert-based models for targeted sentiment analysis on turkish texts," in *Proc. 60th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, (Dublin, Ireland), pp. 467–472, May 2022.

[78] S. Yildirim, "Fine-tuning transformer-based encoder for turkish language understanding tasks," *arXiv preprint arXiv:2401.17396*, 2024.

[79] Y. B. Kaya and A. C. Tantuğ, "Effect of tokenization granularity for turkish large language models," *Intelligent Systems with Applications*, vol. 21, p. 200335, 2024.

[80] Y. K. K. Tohma, H. I. Okur and A. Sertbas, "Sentiment analysis in turkish question answering systems: An application of human-robot interaction," *IEEE Access*, vol. 11, pp. 66522–66534, 2023.

[81] B. K. M. B. G. Sogancıoglu, T. Cekic and O. Agin, "Dialog management for credit card selling via finite state machine using sentiment classification in turkish language," in *INTELLI 2017*, p. 47, 2017.

[82] Y. M. T. C. O. F. Kilic, E. B. Dundar and O. Deniz, "Conversation management in task-oriented turkish dialogue agents with dialogue act classification," in *Proceedings of the International Conference on Knowledge Discovery and Information Retrieval (KDIR 2020)*, pp. 29–35, 2020.

[83] F. A. M. T. O. G. G. Uludoğan, Z. Yirmibeşoğlu Balal and S. Üsküdarlı, "Turna: A turkish encoder-decoder language model for enhanced understanding and generation," *arXiv preprint arXiv:2401.14373*, 2024.

[84] M. E. A. M. Turker and A. Han, "Vbart: The turkish llm," *arXiv preprint arXiv:2403.01308*, 2024.

[85] A. Najafi and O. Varol, "Turkishbertweet: Fast and reliable large language model for social media analysis," *arXiv preprint arXiv:2311.18063*, 2023.

[86] N. B. Çam and A. Ozgur, "Evaluation of chatgpt and bert-based models for turkish hate speech detection," in *Proc. IEEE Conf.*, pp. 229–233, 2023.

[87] S. Dehghan and B. Yanikoglu, "Evaluating chatgpt's ability to detect hate speech in turkish tweets," in *Proc. 7th Workshop on Challenges and Applications of Automated Extraction of Socio-political Events from Text (CASE 2024)*, (St. Julians, Malta), pp. 54–59, Mar. 2024.

[88] C. L. Heavey, D. S. Gill, and A. Christensen, "The couples interaction rating system." Unpublished manuscript, 1998.

[89] P. Ekman, "Basic emotions," in *Handbook of Cognition and Emotion* (T. Dalgleish and M. J. Powers, eds.), pp. 4–5, Wiley, 1999.

[90] O. T. Yildiz, B. Avar, and G. Ercan, "An open, extendible, and fast turkish morphological analyzer," in *Proc. International Conference on Recent Advances in Natural Language Processing (RANLP 2019)*, (Varna, Bulgaria), pp. 1364–1372, INCOMA Ltd., Sep. 2019.

[91] G. Uludogan, R. Özçelik, S. Parlar, G. Ercan, and O. Yildiz, "Türkçe doğal dil İşleme için arayüzler," in *Proc. 2019 Conference on Natural Language Processing Interfaces*, 2019.

[92] Y. Zhang, R. Jin, and Z. Zhou, "Understanding bag-of-words model: a statistical framework," *International Journal of Machine Learning and Cybernetics*, vol. 1, pp. 43–52, 2010.

[93] R. Rehurek and P. Sojka, "Gensim–python framework for vector space modelling," *NLP Centre, Faculty of Informatics, Masaryk University, Brno, Czech Republic*, vol. 3, no. 2, 2011.

[94] J. E. Ramos, "Using tf-idf to determine word relevance in document queries," in *Proc. 2003 International Conference on Document Queries*, 2003.

[95] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," *arXiv preprint arXiv:1310.4546*, 2013.

[96] O. Güngör and E. Yildiz, "Linguistic features in turkish word representations," in *25th Signal Processing and Communications Applications Conference, SIU 2017, Antalya, Turkey, May 15-18, 2017*, pp. 1–4, IEEE, 2017.

[97] S. Schweter, "Berturk - bert models for turkish," Apr. 2020. [Online].

[98] O. O. E. K. D. A.-H. X. L. Q. L. M. R. X. Zhang, N. Thakur and J. Lin, "Miracl: A multilingual retrieval dataset covering 18 diverse languages," *Transactions of the Association for Computational Linguistics*, vol. 11, pp. 1114–1131, 2023.

[99] L. M. N. Muennighoff, N. Tazi and N. Reimers, "Mteb: Massive text embedding benchmark," *arXiv preprint arXiv:2210.07316*, 2022.

[100] F. P. et al., "Scikit-learn: Machine learning in python," *The Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[101] F. Chollet *et al.*, "Keras: The python deep learning library," *Astrophysics Source Code Library*, June 2018. [Online].

[102] B. Zhang and R. Sennrich, "Root mean square layer normalization," in *Advances in Neural Information Processing Systems* (A. B. F. d.-B. E. F. H. Wallach, H. Larochelle and R. Garnett, eds.), vol. 32, Curran Associates, Inc., 2019.

[103] N. Shazeer, "Glu variants improve transformer," *arXiv preprint arXiv:2002.05202*, 2020.

[104] S. P. A. M.-B. W. J. Su, Y. Lu and Y. Liu, "Roformer: Enhanced transformer with rotary position embedding," *arXiv preprint arXiv:2104.09864*, 2023.

[105] Z. Y. Z. Li and M. Wang, "Reinforcement learning with human feedback: Learning dynamic choices via pessimism," *arXiv preprint arXiv:2305.18438*, 2023.

[106] L. D. Y. B.-S. P. S. Mangrulkar, S. Gugger and B. Bossan, "Peft: State-of-the-art parameter-efficient fine-tuning methods," 2022. [Online].

[107] P. W. Z. A.-Z. Y. L. S. W. L. W. E. J. Hu, Y. Shen and W. Chen, "Lora: Low-rank adaptation of large language models," *arXiv preprint arXiv:2106.09685*, 2021.

[108] K. C. J. Chung, Ç. Gülçehre and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *arXiv preprint CoRR abs/1412.3555*, 2014.

[109] K. C. D. Bahdanau and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473*, 2016.

[110] H. P. M.-T. Luong and C. D. Manning, "Effective approaches to attention-based neural machine translation," *arXiv preprint arXiv:1508.04025*, 2015.

[111] T. Y. T. O. T. Akiba, S. Sano and M. Koyama, "Optuna: A next-generation hyperparameter optimization framework," *arXiv preprint arXiv:1907.10902*, 2019.

[112] T. S. M. S. S. A. Md S. U. Miah, Md Kabir and M. Mridha, "A multimodal approach to cross-lingual sentiment analysis with ensemble of transformer and llm," *Scientific Reports*, vol. 14, 2024.

[113] E. S. T. Pires and D. Garrette, "How multilingual is multilingual bert?," in *Proc. 57th Annu. Meeting of the Association for Computational Linguistics*, (Florence, Italy), pp. 4996–5001, Jul. 2019.

[114] N. G. V. C. G. W. F. G. E. G. M. O. L. Z. A. Conneau, K. Khandelwal and V. Stoyanov, "Unsupervised cross-lingual representation learning at scale," *arXiv preprint arXiv:1911.02116*, 2020.

APPENDICES

APPENDIX A. DISTINGUISHING BETWEEN SIMILAR EMOTIONS

A.1 Example 1: Fear vs. Surprise

- **Dialogue Snippet:** "Ne yaptın? Gece oraya yalnız mı gittin?" / "You did what? You went there alone at night?"
- **Context:** Reaction to someone recounting an unexpected solo visit to a known dangerous area.
- **Annotation Challenge:** Deciding whether the emotion is surprise at the action taken or fear for the person's safety.
- **Guideline Application:** The follow-up concern about safety indicates fear.
- **Final Sentiment:** Given the safety concern, the sentiment is **Negative**.

A.2 Example 2: Sadness vs. Neutral

- **Dialogue Snippet:** "Eskisi gibi değiliz." / "We are not like we used to be."
- **Context:** Said during a calm discussion reflecting on relationship changes.
- **Annotation Challenge:** Determining whether the statement expresses sadness about changes or a neutral observation.
- **Guideline Application:** If the tone conveys a sense of loss, it indicates sadness.
- **Final Sentiment:** If reflective of loss, the sentiment is **Negative**; if merely observational, it is **Neutral**.

A.3 Example 3: Happiness vs. Surprise

- **Dialogue Snippet:** "Sana sürprizim var!" / "I have a surprise for you!"
- **Context:** Said during a special moment like a birthday.
- **Annotation Challenge:** Deciding whether this is an expression of genuine happiness or just surprise.
- **Guideline Application:** The context and tone indicating joy suggest genuine happiness.
- **Final Sentiment:** If joy and happiness are evident, the sentiment is **Positive**; if it merely points to an unexpected situation, it could be **Neutral**.

APPENDIX B. EXAMPLES OF THE EMOTIONAL BLENDS

B.1 Example 1: Happiness vs. Sadness

- **Dialogue Snippet:** "Bugün terfi aldım, çok mutluyum!" / "I got promoted today, I'm so happy!"
  "Ama bu terfiyle birlikte daha az zaman geçireceğiz." / "But with this promotion, we'll spend less time together."
- **Context:** Said during a conversation about a recent promotion.
- **Annotation Challenge:** Distinguishing between genuine happiness about the promotion and sadness about the potential impact on their time together.
- **Guideline Application:** The first sentence clearly shows happiness due to the promotion (label as Happiness), while the second sentence reflects sadness about the decrease in time spent together (label as Sadness).
- **Final Sentiment:** If the sadness about time spent together is more deeply felt and emphasized in subsequent context, the final label should be **Sadness** (Negative). Otherwise, if the happiness outweighs the concern, it can be **Happiness** (Positive).

B.2 Example 2: Positive Surprise vs. Negative Fear

- **Dialogue Snippet:** "Bana yeni bir araba aldın mı? Bu inanılmaz!" / "Did you buy me a new car? This is amazing!"
  "Ama bu kadar para harcadığından korkuyorum." / "But I'm afraid of how much money you spent."
- **Context:** Said during a surprise reveal of a new car.
- **Annotation Challenge:** Distinguishing between the initial positive surprise and the subsequent negative fear regarding the financial implications.
- **Guideline Application:** The first sentence demonstrates positive surprise with excitement and amazement (label as Surprise), while the second sentence introduces negative fear about the financial cost (label as Fear).
- **Final Sentiment:** If the fear about the financial cost is more pressing and highlighted in the following utterances, the final label should be **Fear** (Negative). If the surprise and joy dominate the conversation, it can be **Surprise** (Positive).

B.3 Example 3: Disgust vs. Neutral

- **Dialogue Snippet:** "Bu yemek çok kötü kokuyor, bunu yiyemem!" / "This food smells awful, I can't eat this!"
  "Ama belki sadece benim damak tadıma uymuyordur." / "But maybe it just doesn't suit my taste."
- **Context:** Said during a meal where the food is found unappealing.
- **Annotation Challenge:** Differentiating between strong disgust and a neutral assessment.
- **Guideline Application:** The first sentence shows strong disgust with vivid language (label as Disgust), while the second sentence provides a neutral explanation (label as Neutral).
- **Final Sentiment:** If the disgust is more intense and repeated in the conversation, the final label should be **Disgust** (Negative). If the neutral assessment prevails and reduces the intensity of disgust, it can be **Neutral**.

APPENDIX C. PSEUDOCODE

The complete pseudocode of the conversational sentiment analysis on the Couple Dialogue dataset is given in below:

Algorithm 1 Pseudocode for Data Preparation and Preprocessing in Conversational Sentiment Analysis.

```
Input: CoupleDialogueDataset
Output: PreprocessedData

NonContextualFeatureMethods ← [BoW, TF-IDF, Word2Vec, GloVe, PretrainedTurkishBERT]
ContextualFeatureMethods ← [GloVe, PretrainedTurkishBERT, FineTunedTurkishBERT, OpenAITextEmbedding]

Define Basic Preprocessing:
BasicPreprocessing ← {"ISO encoding": True, "expanding contractions": True, "eliminating numerical values": True,
                      "removing single characters": True, "removing excessive spaces": True, "lowercasing": True}
Define Additional Preprocessing Configurations:
PreprocessingConfigs ← {
    "DetailedRootForm": BasicPreprocessing + {"remove_stopwords": True, "remove_punctuation": True, "retain_specific_punctuations": []},
    "PartialRootForm": BasicPreprocessing + {"remove_stopwords": False, "remove_punctuation": True, "retain_specific_punctuations": ["!", "..."]},
  "SurfaceForm": BasicPreprocessing + {"remove_stopwords": False, "remove_punctuation": True, "retain_specific_punctuations": ["!", "..."], "include_morphemes": True},
  "BasicPreprocessing": BasicPreprocessing
}

Initialize Nested Dictionary:
function INITIALIZE_NESTED_DICTIONARY
    return lambda: defaultdict(initialize_nested_dictionary)
```

Algorithm 2 Pseudocode for Training and Evaluating Non-Contextual Models.

```
Input: Dataset, NonContextualFeatureMethods, PreprocessingConfigs
Output: NonContextualScoresByPreprocessing

function TrainAndEvaluateNonContextualModels(Dataset, NonContextualFeatureMethods, PreprocessingConfigs)
    NonContextualScoresByPreprocessing ← initialize_nested_dictionary()
    for each ConfigName, ConfigDetails in PreprocessingConfigs do
        for each FeatureMethod in NonContextualFeatureMethods do
            for each Classifier in NonContextualClassifiers do
                if Classifier == DeepLearningModel(Keras) and (FeatureMethod == "GloVe" or FeatureMethod == "PretrainedTurkishBERT") then
                    PreprocessedDataset ← ApplyPreprocessing(Dataset, PreprocessingConfigs["BasicPreprocessing"])
                else if FeatureMethod == "PretrainedTurkishBERT" and Classifier != DeepLearningModel(Keras) then
                    PreprocessedDataset ← ApplyPreprocessing(Dataset, PreprocessingConfigs["BasicPreprocessing"])
                else
                    PreprocessedDataset ← ApplyPreprocessing(Dataset, ConfigDetails)
                FoldScores ← initialize_empty_list()
                for each Fold in KFold(PreprocessedDataset, 10) do
                    Train, Validation, Test ← Fold
                    TrainFeatures, ValidationFeatures, TestFeatures ← ExtractFeatures(Train, Validation, Test, FeatureMethod)
                    OptimizedHyperparameters ← OptimizeHyperparameters(ValidationFeatures, Classifier)
                    TestScore ← EvaluateModel(TestFeatures, Classifier, OptimizedHyperparameters)
                    FoldScores.Append(TestScore)
                AverageFoldScore ← Average(FoldScores)
                NonContextualScoresByPreprocessing[ConfigName][FeatureMethod][Classifier] ← AverageFoldScore
    return NonContextualScoresByPreprocessing
```

Algorithm 3 Pseudocode for Contextual Models, DialogueRNN, and Final Evaluation.

```
Input: CoupleDialogueDataset, ContextualFeatureMethods, ContextualClassifiers
Output: FinalResults

function EVALUATEDIALOGUERNNWITHKFOLD(Dataset, ContextualFeatureMethods)
    KFoldResultsByEmbedding ← initialize_empty_dictionary()
    for each EmbeddingType in ContextualFeatureMethods do
        KFoldResults ← initialize_empty_list()
        for each Fold in KFold(Dataset, 10) do
            Train, Validation, Test ← Fold
            FoldTrainFeatures ← ExtractDialogueFeatures(Train, EmbeddingType)
            FoldValidationFeatures ← ExtractDialogueFeatures(Validation, EmbeddingType)
            FoldTestFeatures ← ExtractDialogueFeatures(Test, EmbeddingType)
            TestScore ← EvaluateModel(FoldTestFeatures, DialogueRNN, OptimizedHyperparameters)
            KFoldResults.Append(TestScore)
        KFoldResultsByEmbedding[EmbeddingType] ← Average(KFoldResults)
    return KFoldResultsByEmbedding

function TRAINANDEVALUATECONTEXTUALMODELS(TrainData, ValidationData, TestData)
    ContextualScores ← initialize_empty_dictionary()
    for Classifier in ContextualClassifiers.PromptBased do
        ApplyPromptingTechniques(Classifier, TrainData, ValidationData)
        Score ← EvaluateModel(TestData, Classifier)
        ContextualScores[Classifier] ← Score
    for FineTunedClassifier in ContextualClassifiers.FineTuned do
        Model, Score ← FineTune(FineTunedClassifier, TrainData, ValidationData)
        Score ← EvaluateModel(TestData, Model)
        ContextualScores[FineTunedClassifier] ← Score
    return ContextualScores
DatasetSplit ← StandardSplit(CoupleDialogueDataset)
NonContextualResults ← TrainAndEvaluateNonContextualModels(CoupleDialogueDataset, NonContextualFeatureMethods, PreprocessingConfigs)
DialogueRNNResults ← EvaluateDialogueRNNWithKFold(CoupleDialogueDataset, ContextualFeatureMethods)
ContextualResults ← TrainAndEvaluateContextualModels(DatasetSplit.TrainData, DatasetSplit.ValidationData, DatasetSplit.TestData)
Print("Non-Contextual Results: ", NonContextualResults)
Print("DialogueRNN Results: ", DialogueRNNResults)
Print("Contextual Results: ", ContextualResults)
```

APPENDIX D. SOFTWARE PACKAGES

Table D.1 and Table D.2 presents information on the packages used in this study, including a short description and the versions installed.

Table D.1: *Software Packages Used in the Study (Part 1).*

Package Name	Description	Version
openai	Provides access to AI models like GPT.	0.28.0
torch	ML library for vision and NLP.	2.3.0
tensorflow	Framework for numerical computation.	2.16.1
keras	Neural networks API, runs on TensorFlow.	3.3.3
pandas	Data manipulation and analysis library.	2.2.1
numPy	Supports large, multi-dimensional arrays.	1.26.4
scikit-learn	Supports supervised and unsupervised learning.	1.4.1
transformers	Pretrained models for text tasks.	4.40.1
nltk	Toolkit for human language data.	3.8.1
spaCy	Industrial-strength NLP in Python.	3.7.4
tqdm	Progress bar for Python and CLI.	4.66.2
statsmodels	Explore data, estimate models, and perform tests.	0.14.0

Table D.2: *Software Packages Used in the Study (Part 2).*

Package Name	**Description**	**Version**
datasets	Accessing and sharing datasets in AI.	2.18.0
seaborn	Statistical graphics.	0.13.2
matplotlib	Plotting library for Python and NumPy.	3.8.3
peft	Allows for the fine-tuning of large language models without updating all of the model's parameters.	0.9.0
trl	Hugging Face's library for supervised fine-tuning (SFT) for training LLMs on instruction datasets.	0.7.12
tensorboard	Visualization tool for TensorFlow, providing metrics plot, graphs display, and more.	2.16.2
gensim	Python library for topic modeling and semantic analysis using NLP.	4.3.1
Microsoft Office Spell Checker	Provides spelling, grammar, and style corrections for Turkish language.	2016

VITA

Esma Nafiye Polat received her bachelor's degree in computer engineering in 2010, with a double major in mathematics in 2011, from Maltepe University, Istanbul, Turkey. She completed her master's degree in computer engineering at Istanbul Şehir University, Istanbul, Turkey, in 2013, specializing in text mining, information retrieval, social network analysis, and natural language processing (NLP). She is currently pursuing a Ph.D. in computer engineering at Özyeğin University, Istanbul, Turkey, where her research focuses on large language model (LLM)-based networks for conversational sentiment analysis in dyadic relationships.

She has 8 years of industry experience and 3 years in academia. She currently serves as a Senior NLP Researcher at Afiniti Inc., where she leads NLP projects. Previously, she worked as an Expert Data Scientist at Kuveyt Turk Participation Bank, leading AI initiatives related to call count estimation, cheque predictions, and credit limit scoring. Her career also includes roles as a Junior Data Scientist at Turkish Airlines and a Teaching Assistant at Marmara University, where she supported undergraduate courses and fulfilled teaching assistant duties. Her research centers on developing innovative NLP algorithms and systems using large language models, transfer learning, data science, and machine learning techniques. She has publications relevant to the fields of NLP, machine learning, and transformer models.

Ms. Polat received the Best Track Paper award at the 2013 International Conference on Electronics Computer and Computation (ICECCO) (IEEE).