Incongruence Identification in Eyewitness Narratives

A Project Report

submitted by

AKSHARA NAIR

in partial fulfilment of the requirements
for the award of the degree of

MASTER OF TECHNOLOGY

COMPUTER SCIENCE AND ARTIFICIAL INTELLIGENCE
INDRAPRASTHA INSTITUTE OF INFORMATION TECHNOLOGY DELHI

NEW DELHI- 110020

July,2024

THESIS CERTIFICATE

This is to certify that the thesis titled **Incongruence Identification in Eyewitness Narratives** being submitted by Akshara Nair, to the Indian Institute of Information Technology, Delhi, for the award of the degree of **Master of Technology**, is a bonafide record of the research work done by her under my supervision.

The contents of this thesis, in full or in parts, have not been submitted to any other Institute or University for the award of any degree or diploma.

Dr Md. Shad Akhtar
Thesis Supervisor
Department of Computer Science Engineering
IIIT Delhi, 110020

Date: July 2024

ACKNOWLEDGEMENTS

I wish to express my deepest gratitude to Dr. Md Shad Akhtar, my advisor, for providing me with the exceptional opportunity to pursue an independent research project in the fascinating field of Incongruence Detection. Dr. Akhtar's unwavering support, expert guidance, and insightful feedback have been invaluable throughout my academic journey. His profound knowledge and patient mentorship have not only directed my research but also inspired me to overcome challenges and strive for excellence. This thesis would not have been possible without his dedicated assistance and encouragement, and I am truly fortunate to have had him as my mentor.

I also extend my heartfelt thanks to the Department of Computer Science at IIITD for providing an enriching academic environment and the resources necessary for my research. Their support has been instrumental in the successful completion of this thesis.

Furthermore, I owe profound gratitude to my family and friends for their relentless moral and emotional support. Their constant encouragement and presence have provided me with the strength and motivation necessary to complete this endeavor.

ABSTRACT

Eyewitness accounts are crucial to legal and investigative proceedings, serving as foundational sources for reconstructing events. Agencies typically collect multiple statements to achieve a thorough understanding of the situation. However, during investigations, it is essential to detect any incongruences in these eyewitness accounts, such as conflicting details about timelines, actions, or identities. Comparing and identifying these inconsistencies within testimonies is critical because they often signal potential deception or manipulation of facts. Traditional methods for identifying inconsistencies often prove inadequate, as they lack access to detailed, event-specific datasets. Moreover, these conventional approaches do not pinpoint the exact incongruence between two statements, which is necessary to provide direct evidence supporting the detected answer. This thesis proposes a novel framework for identifying incongruences between two testimonies by comparing their responses within the context of shared questions.We created a Multimodal Eyewitness Deception Detection Dataset (ED3) which contains the testimonies of eyewitnesses collected through an interview process after witnessing scenarios involving different stimuli (e.g., a simulated crime) Our research is centered on two tasks: first, identifying the presence of incongruence within the statements of witnesses. We demonstrate the superior efficacy of prompt tuning techniques over traditional fine-tuning methods in this identification process. Second,detecting the exact contradiction statements within the testimonies, we utilized the reasoning ability of LLM and proposed a three step reasoning framework inspired by the Chain-of-Thought (COT) methodology.This method breaks down the statements into a step-by-step reasoning process, mirroring human problem-solving behaviors, which helps in systematically identifying, explaining the discrepancies found and facilitating the extraction of incongruent text spans. The outcomes of this thesis contribute to more accurate and dependable analysis of testimony evidence, enhancing the reliability of legal and investigative practices through the use of generative AI methods.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

> "There is almost nothing more convincing than a live human being who takes the stand, points a finger at the defendant and says, 'That's the one,'"
> *— late U.S. Supreme Court Justice William J. Brennan Jr.*

1.1 Definition

Inter-eyewitness incongruence refers to the discrepancies or contradictions between different eyewitness statements regarding the same point of reference. These inconsistencies can appear in various aspects, such as differing accounts of the sequence of events, descriptions of individuals involved, details of actions taken and, more.

1.2 Motivation

Identifying inter-eyewitness incongruence is important for several reasons. Firstly, it contributes towards ascertaining the credibility and reliability of eyewitnesses. Agreements in accounts very strongly imply the likelihood of provision of correct information. On the other hand, major inconsistencies cast doubts on the truth of the accounts; such incongruence may suggest that the person is not genuine or trying to deceive. If someone's statement is at variance with the established fact base, then they must be lying and making the statement either to mislead investigators or to further some other agenda.Identification and elimination of disparities can render the pool of overall information more effective, increasing the clear and correct picture of the incident enhancing the effectiveness of legal and investigative processes. Yet to date, no comprehensive framework effectively addresses eyewitness incongruence, highlighting the need for advanced methodologies to detect and analyze these inconsistencies accurately.

1.3 Problem Statement

Given:

- **C**: Shared context under discussion.
- **A1**: Answer related to context C by eyewitness 1.
- **A2**: Answer related to context C by eyewitness 2.

We define the triplet:

$$T = \langle C[c_1, c_2, \ldots, c_k], A_1[a_{11}, a_{12}, \ldots, a_{1l}], A_2[a_{21}, a_{22}, \ldots, a_{2j}] \rangle$$

Task 1: Identify implicit incongruence between the two answers A_1 and A_2. This involves determining whether the answers are contradictory, denoted as a subset of {Yes, No}.

Task 2: Extract the specific spans within the two answers that are contradicting. This involves identifying the segments of text $p_1, p_2, \ldots, p_m$ from A_1 and $q_1, q_2, \ldots, q_n$ from A_2 that contradict each other, denoted as:

$$S[p_1, p_2, \ldots, p_m \text{ contradicts } q_1, q_2, \ldots, q_n]$$

1.4 Contributions

1. Created a comprehensive eyewitness dataset that includes multimodal data (text, audio, video) to facilitate nuanced deception detection.
2. Addressed the novel task of inter-incongruence detection, focusing on identifying contradictions between testimonies related to a common event in the domain of deception detection.
3. Analyzed and benchmarked the performance of various pretrained language models on incongruence detection task. We designed and demonstrated the efficacy of various prompt templates in identifying contradictions.
4. Developed a multi-hop reasoning framework to detect the exact incongruent spans and provided a thorough analysis of its performance.

CHAPTER 2

RELATED WORKS

2.1 Deception Detection

Deception analysis based on visual and verbal cues have been analyzed in many genres. Early studies, dating back to 1969, focused on predicting cues to deception. Ekman and Friesen (1969)published the first influential theoretical statement on cues to deception, followed by works by Zuckerman *et al.* (1981), Buller and Burgoon (2006). In Pérez-Rosas *et al.* (2015) study, he analyzed data from real court testimonies, where deception involves high stakes. Psychological research suggests that in such contexts, signs of deception are more noticeable. Mihalcea and Strapparava (2009) conducted one of the earliest studies to distinguish between truth and lies in written text. Due to the limited availability of real-world data, Levitan *et al.* (2018) conducted research on deception by simulated interview scenarios where participants are instructed to intentionally deceive.

2.2 Eyewitness Testimony

There has been limited research in the domain of eyewitness testimonies focusing on event-based scenarios. Solà-Sales *et al.* (2023) conducted research examining the linguistic styles of simulated deception and true testimonies, with a focus on studying witness memory. Participants acted as witnesses of a crime by retelling a story they had just read. The study compared testimonies from the same participants under different conditions to analyze variations in (i) lexical and (ii) linguistic features, as well as (iii) content and speech characteristics (such as disfluencies) depending on the narrative condition. In 2024 Greenspan *et al.* (2024) assessed Verbal Eyewitness Confidence Statements Using Natural Language Processing and furnishes a new metric, confidence entropy, that measures the vagueness of witnesses' confidence statements to provide independent information about eyewitness accuracy

2.3 Incongruence Detection

Marneffe *et al.* (2008) were among the first to present empirical results for contradiction detection, focusing specifically on negation and paraphrased contradictions. de Marneffe *et al.* (2008) suggest a definition for contradictions in NLP tasks and analyze the task's performance. Wu *et al.* (2022) introduce a novel task focused on cross-document misinformation detection, aiming to identify fake information within clusters of topically related news documents. Their proposed detector enhances document-level detection by utilizing features generated from an event-level detector

2.4 Existing Dataset

Deception research relies on two types of datasets: Mock and Real-life.

Real-life Datasets: These datasets comprise data collected from real-world situations where individuals are presumed to have lied spontaneously without explicit instructions or incentives. One example is the Real Life Trial dataset introduced by Pérez-Rosas *et al.* (2015), which consists of deceptive clips obtained from suspects during guilty verdicts in courtrooms, while truthful clips were gathered from victims in the same setting. This dataset contains 121 videos, including 61 deceptive and 60 truthful trial videos, with an average length of 28 seconds. Another example is the dataset developed for the study on Fernandes and Ullah (2021) This dataset includes audio recordings of a guilty male suspect being interrogated by a law enforcement officer, with the suspect's responses to an identical set of questions recorded three times throughout the day.

Mock Datasets:

These datasets are created in controlled environments where participants are encouraged or instructed to provide deceptive information The Columbia-SRI-Colorado Corpus (CSC) Deceptive Speech University *et al.* (2013) consists of 32 hours of audio interviews with 32 native English speakers, split evenly by gender, conducted by Columbia University, SRI International, and Colorado Boulder Universities. Participants believed they were part of a project to identify top entrepreneurs and were asked to convince interviewers of their high scores. Transcriptions followed NIST EARS guidelines. The Bag-of-Lies dataset, introduced by Gupta *et al.* (2019) in 2019, includes visual, audio, EEG, and eye gaze data, capturing practical scenarios with standard mobile cameras

and microphones. EEG data was collected using a 14-Channel Emotiv EPOC+EEG headset, and Gazepoint GP3 Eye Sensor was used for gaze data collection.

2.5 Limitation of Existing Dataset

Existing Dataset Limitations and Research Gaps in Deception Detection

Lack of Span Identification: Most datasets classify entire sentences or texts as deceptive or not, without identifying the specific span or segment within the text where the deception actually occurs because of lack of ground truth

Overlooking multiple viewpoints: Existing datasets often fail to account for different viewpoints on a single issue, which could lead to a biased or inadequate knowledge of deceptive behaviors. Insufficient Examination of Eyewitness testimony: There is a limited emphasis on analyzing eyewitness testimony, which are important in some situations such as legal proceedings

Limited Range of Events: Many datasets focus on a very small number of events, and sometimes only a single event, limiting the diversity and general applicability of the data for training robust deception detection models.

Simplification of Deception Dynamics: By not addressing the nuanced ways in which information can be misrepresented (e.g., through omission, distortion, or half-truths within specific spans), there's a risk of oversimplifying the complexity of deceptive communication. In an investigation setup, no-one has worked upon unveiling the discrepancy in the narratives of multiple interrogatories or between two statements of an interrogatee.

Discrepancy Detection in Interrogations: In the investigation setup, the discrepancy in narratives between multiple interrogatories or within statements of an interrogatee remains unexplored.

CHAPTER 3

Dataset

3.1 ED3 Dataset

To overcome the limitation of the existing Deception Dataset, we introduce a multi-modal eyewitness Deception Detection (ED3). The dataset was developed through the following stages.

1. **Establishment of Data Collection Guidelines**
2. **Video Collection**
3. **Scenario Summarization and Question Development**
4. **Interrogation**
5. **Data Cleaning and Transcription**

3.2 Establishment of Data Collection Guidelines

Prior to collecting data, we meticulously developed guidelines that outline ethical standards, criteria for selecting scenarios based on various types of criminal events, and participant procedures for ensuring consistent and reliable data collection.

Event Video Collection Guideline

The collected videos must depict events that are realistic and representative of actual criminal incidents or scenarios where eyewitness testimonies are critical.

Diverse range of scenarios that capture various types of criminal activities or incidents to ensure the dataset covers a broad spectrum of potential deceptive behaviors and testimonial challenges.

Interrogation Guideline

Volunteers watch a video depicting a crime scenario, such as theft or assault.

After viewing the video, volunteers are given 5-10 minutes to reflect and recollect the witnessed scenario.

The participants are instructed to answer each question either truthfully or you can try to deceive the interviewer using the following ways of deception:

Concealment: Not providing accurate descriptions of the perpetrator's appearance, actions, or the environment, or intentionally downplaying certain details.

Fabrication: Intentionally creating or inventing false information or details. It can include making up events, experiences, or descriptions that did not occur, or providing fabricated responses to questions.

Distortion: Modifying specific details or aspects of the witnessed event Interviewers ask structured questions designed to extract detailed testimonies. Questions may include both direct queries about observed events and potential follow-ups to assess the consistency and accuracy of responses.

3.3 Video Collection

We acquired high-quality video recordings depicting different types of crimes or incidents relevant to the dataset's objectives. We ensured diversity in scenarios to encompass a range of criminal contexts.

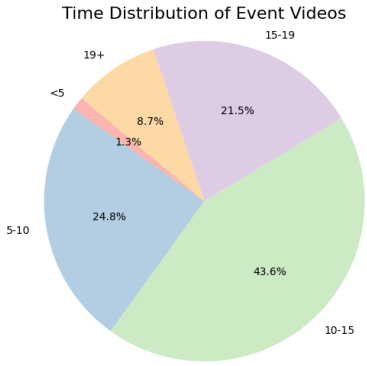

Figure 3.1: Time Distribution of Event Videos in minutes

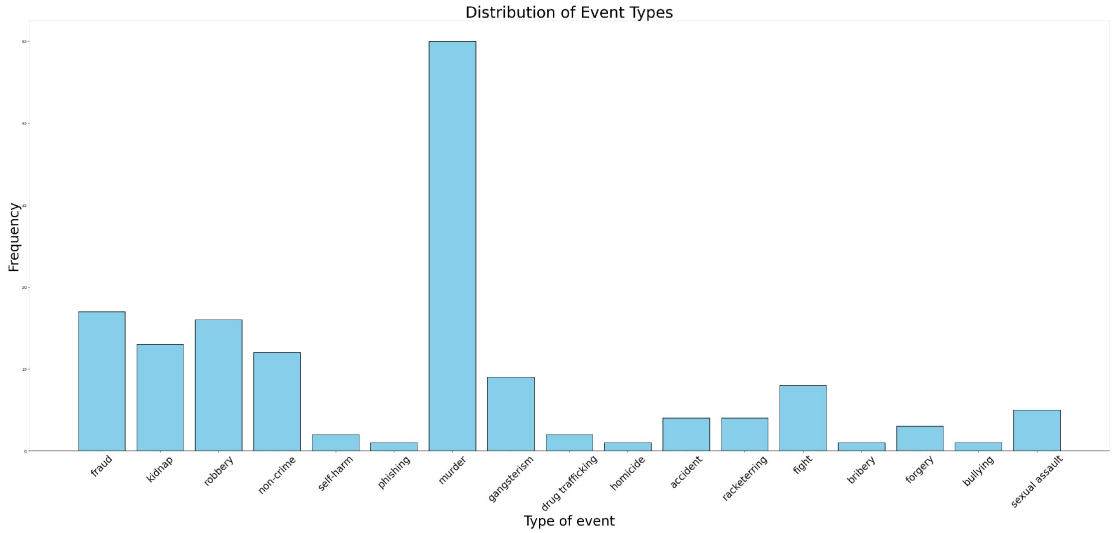

Figure 3.2: Distribution of type of Events observed by Eyewitnesses

Number of Events	149
Average length of the videos	12.14 minutes

Table 3.1: Summary of Events and Video Lengths

3.4 Scenario Summarization and Question Development

We created concise summaries of the original events depicted in the collected videos. Formulate a structured set of predetermined questions designed to extract comprehensive testimonies from eyewitnesses.

Number of Predefined questions	1332
Average number of question for each event	9

Table 3.2: Summary of predefined Questions

3.5 Interrogation

We conducted systematic interrogations following the viewing of video footage by eyewitness subjects. We employed both predetermined and cross-examination questions to elicit detailed testimonies.

Figure 3.3: Eyewitnesses involved in the interrogation process.

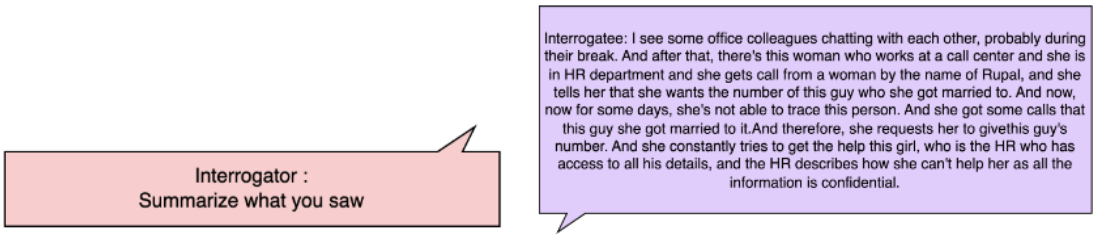

Figure 3.4: Example of an interrogation exchange.

3.6 Data Cleaning and Transcription

We processed recorded interrogation sessions by cleaning video recordings to enhance clarity, then transcribed testimonies into standardized textual formats suitable for further analysis and annotation. Transcriptions were initially obtained from Otter.ai and subsequently refined.

Overall Conversation Count	389
Total Q&A Pair Count	6013
Average Interrogation Duration	9 minutes 30 seconds
Average Summary Duration	2 minutes 50 seconds

Table 3.3: Summary statistics of the recorded interrogation sessions.

3.7 Annotation Guidelines

1. Use predetermined questions to define the context for each event.
2. Extract utterances from witness testimonies that directly respond to the established context.
3. Gather responses from different eyewitnesses for the same predetermined question.
4. Examine two responses for contradictions within similar context based on:
   - Identity Differences
   - Actions Described
   - Object Inconsistencies
   - Timing Discrepancies
   - Location Variations
   - Motivation Differences
5. Identify and mark specific phrases or segments that are logically inconsistent when the context is similar.
6. Label the context and answer triplets (responses from two different witnesses for the same question) as 1 if the responses are logically inconsistent; otherwise, label them as 0.

CHAPTER 4

Proposed Methodology

Pre-trained language models have garnered increasing attention for enhancing the effectiveness of downstream applications. These models have demonstrated exceptional capabilities in common-sense understanding and reasoning. Our study focuses on utilizing the predictive and reasoning abilities of language models to detect implicit contradictions.It encompasses two primary tasks: predicting incongruence presence and extracting incongruent spans. In our experiments, we will use 'incongruence' and 'contradiction' interchangeably.

4.1 Prompt Tuning for Incongruence Detection

Fine-tuning a Pre-trained Language Model (PLM) involves training a language model that takes an input $x = (x_0, x_1, \ldots, x_n)$, generates a probability distribution over class Y, and predicts an output y as $P(y \mid x)$.

In prompt tuning, the input x is wrapped with a prompt template. Specifically, the input x is transformed into x_p with the prompt template:

$$x_p = \text{a? [MASK] b}$$

The model M generates a probability distribution over the vocabulary V at the [MASK] position, giving the probability of each token $v \in V$ filling the [MASK] token:

$$P_M(\text{[MASK]} = v \mid x_p)$$

Additionally, a verbalizer maps from the label word set $V_y \subseteq V$ to the label space Y. For the given prompt, we define:

$$V_{\text{n\_inc}} = \{\text{"True"}\}$$

as the label word for the non-incongruent class, and

$$V_{\text{inc}} = \{\text{"False"}\}$$

as the label word for the incongruent class.

The probability of label y is then calculated as:

$$P(y \mid x_p) = g(P_M(\text{[MASK]} = v \mid x_p) \mid v \in V_y)$$

where g is a function that transforms the probability of label words into the probability of the class.

If $P(V_{\text{n\_inc}}) > P(V_{\text{inc}})$, we classify the instance as non-incongruent.

4.1.1 Prompt Construction

In the context of implicit semantic understanding tasks such as contradiction detection, the model needs to compare the semantics of one text with another to identify contradictions. The goal of prompt construction is to provide clear guidance to the model about the task to enhance its ability to detect these contradictions. Specifically, we define two kinds of prompts, including Question prompts and 6W prompts.

Question prompt

These prompts guide the model to detect inconsistencies by posing specific questions. Building on the summary of prompt design for various tasks from Liu *et al.* (2021), we propose two question prompt templates for recognizing incongruence: `Tq1(x)` and `Tq2(x)`.

- `Tq1(x)`: [X] Is there a direct contradiction between the statements made by Witness A and Witness B? [MASK]
- `Tq2(x)`: [X] Can event A and B occur simultaneously? [MASK]"

6W Prompts

The 6W prompt inspired by Mott (1942) is a structured method to compare two testimonies by examining specific aspects (the "6 Ws": Who, What (action), What (object), Where, When, and Why) and aims to identify consistencies, contradictions, or absences in the details provided by the witnesses. The goal of prompt construction is to narrow the focus to particular aspects of the testimonies.

- `Tw1`: Witness A's identification of the person is [MASK] with Witness B's identification.
- `Tw2`: Witness A's described action is [MASK] with Witness B's description.
- `Tw3`: Witness A's described object is [MASK] in Witness B's described object.
- `Tw4`: Witness A's reported timeline is [MASK] with Witness B's timeline.
- `Tw5`: Witness A's location of the event is [MASK] with Witness B's location.
- `Tw6`: Witness A's explanation of the motive is [MASK] with Witness B's explanation.

This elicit the model to compare on each aspect individually

4.1.2 Verbalizer Engineering and Label Words

Language models generate a probability distribution over their vocabulary set, making it crucial to find suitable words that fit within this vocabulary and are appropriate for the masked position. Verbalizer engineering involves selecting label words that accurately convey the intended meaning, enabling the model to produce precise and relevant responses.

For Question prompts, *Yes* or *No* is used to answer the question.

For the 6W prompts, the following labels will be instructed and used to fill the masks during instruction tuning:

- **complement**: Use when both testimonies provide consistent or additional details that align with each other.
- **contradict**: Use when testimonies directly conflict with each other.
- **absent**: Use when one testimony does not provide information on a detail covered by the other.

Verbalizer Mapping

Verbalizer mapping is the process of associating specific output words from a language model with predefined categories or classes. In the case of a question prompt, the word "Yes" is mapped to 1, and "No" is mapped to 0. Similarly, for a 6W prompt, the word "Contradict" is mapped to 1, while both "Consistent" and "Absent" are mapped to 0.

We designed two types of prompts for prompt tuning and analyzed the performance of four MLM models and four LLMs. Specifically, we targeted the following research questions:

RQ1: How does prompt tuning for contradiction detection perform under standard supervised settings using MLMs

RQ2: How does prompt tuning for contradiction detection perform under low-resource settings for MLMs ?

RQ3: How does LLMs perform under Instruction Tuned Setting

RQ4: Which prompt tuning technique performs best and why?

4.2 Extracting Incongruent Span

We focus on the extraction of Incongruent spans using two primary techniques: Few-shot Learning and a Multi-hop Reasoning Framework.

Few-shot Learning

Few-shot learning operates by presenting K examples of context and their completions, followed by an example of context where the model is expected to provide the completion. In this study, we set K = 4 examples. The primary advantages of few-shot learning include significantly reducing the need for task-specific data and minimizing the risk of learning from a narrowly focused fine-tuning dataset.

Selection Criteria for K Examples:

The selection of examples is critical for effective few-shot learning. We used examples covering the following scenarios:

1. **No Contradiction Spans:** Both accounts provide consistent/unrelated information with no contradictions.
2. **Unanswered Response (No Contradiction):** One account provides information while the other does not answer, resulting in no contradictions.
3. **Single Discrepancy:** The two accounts contain a single piece of contradictory information.
4. **Multiple Discrepancies:** The two accounts contain multiple pieces of contradictory information.

4.2.1 Input Transformation and Prompt Template

Each input consists of a question (the context) and two texts in the following format:

> **Question**: {context1}
> **Account A**: {text1}
> **Account B**: {text2}

For contradictory spans, the following prompt is used:

> Identify and extract specific text segments that contradict between Witness A and Witness B's testimonies. Format your findings as follows:
>
> > Contradiction 1: ["Span1" from Witness A] contradicts ["Span" from Witness B]
> > Contradiction 2: ["Span2" from Witness A] contradicts ["Span" from Witness B]
>
> Ensure your output strictly adheres to this format. If no contradictions are found, output "No contradiction."

We leverage the generalization capability of large pre-trained language models. The models are presented with 4 examples of context and their corresponding contradictory spans, followed by a new context where the model is expected to identify contradictory spans.

4.2.2 Multi Hop Reasoning:

Witness testimonies often involve intricate details and subtle contradictions that may not be adequately represented in a small training set typically used for few-shot learning. Fine-tuning requires a substantial amount of labeled data, which may be costly and time consuming to collect, especially for the diverse and nuanced incongruencies present in testimonies.

This can be addressed by integrating information from multiple segments of text by iteratively linking and reasoning across these sources which allows the model to build a comprehensive understanding of the context surrounding contradictions.

Three Hop Reasoning

Inspired by the work of Fei *et al.* (2023) we developed a multi-step reasoning framework using three distinct prompts to enhance the extraction of contradictory spans. This approach incrementally builds on the model's reasoning capabilities, addressing the complexities of the task in a structured manner:

Identifying Fine-Grained Key Details: The first prompt focuses on extracting fine-grained key details from the two accounts in relation to the given question. This ensures that the model accurately comprehends the specific information provided in each testimony.

Incongruence Reasoning: The second prompt involves using common sense to reason about the presence of contradictions. By applying logical and contextual reasoning, the model can detect potential contradictions between the accounts.

Extracting Contradictory Spans: The third prompt is dedicated to extracting the specific text spans that contain contradictions, based on the reasoning conducted in the previous step. This step involves identifying the exact segments that contradict each other, guided by the earlier inferences.

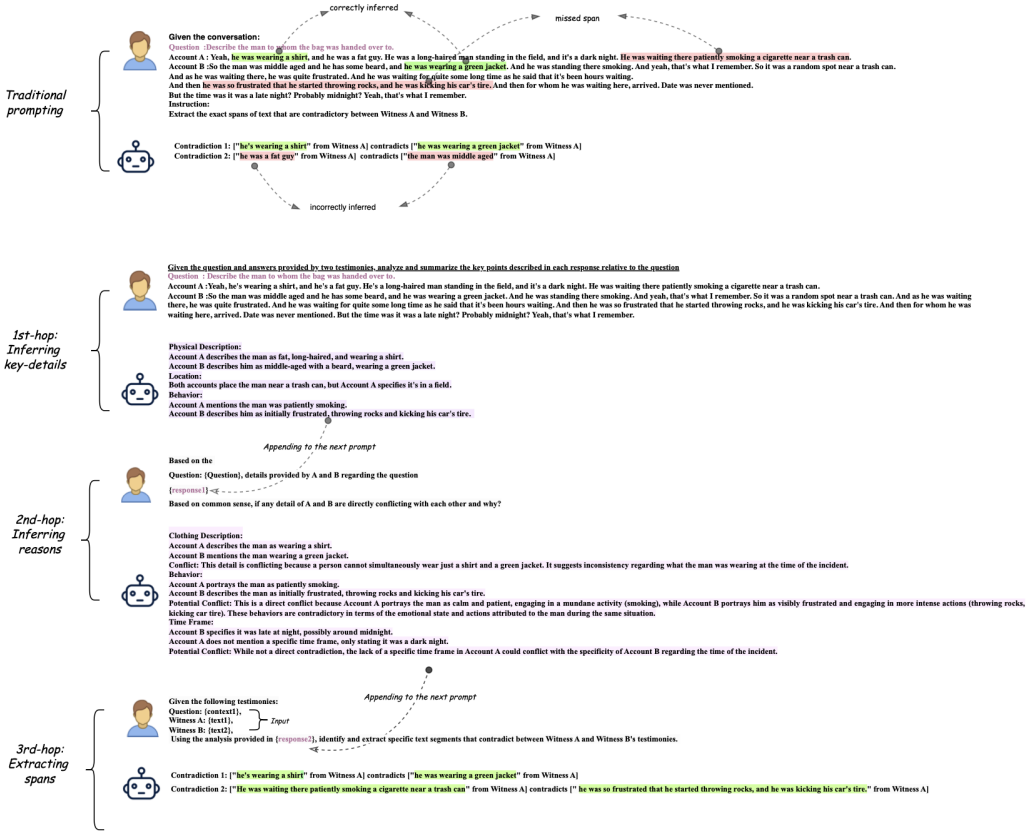

Figure 4.1: Illustration of multi hop reasoning framework

This incremental reasoning approach moves from simpler to more complex tasks, enabling the model to uncover hidden contexts progressively. By breaking the task into these three steps, the framework simplifies the process of span extraction, effectively addressing the challenges associated with direct contradiction prediction. This method leverages the model's ability to integrate and reason through multiple layers of information, resulting in a more precise and reliable extraction of contradictory spans from witness testimonies.

CHAPTER 5

Results

5.1 Experimental Setup

Incongruence Detection: We conducted experiments on our dataset and partitioned it into 605% for validation and 35% for testing . To evaluate the outputs, we utilized Standard metrics including precision, recall and F1-score.
Incongruence Span: We evaluated our untrained models on a few shot and multi hop settings utilizing 500 samples of the data. The evaluation of the outputs are done using metrics including (INC-1 precision,INC-1 recall,INC-1 F1-score, INC-2 precision, INC-1 recall, INC-1 F1-score, INC precision, INC recall, INC F1-score).

5.2 Models

Incongruence Detection:
In our research on Incongruence detection, we explored the capabilities of these Masked Language Models (MLMs): Longformer,, BigBird and BigBird fine tuned on MNLI dataset.
Large Language Models: LLAMA-7b, Qwen-7b, Mixtral-7b, Gemma-7b, CHAT-GPT, CHAT GPT-4o
Incongruence Span: We experimented with variations of LLAMA, Gwen, Mixtral, Gemma Large Language Models.

5.3 Hyperparameters

Incongruence Detection

We used the MLM tokenizer with a maximum token length of 1024. Our setup included the Adam optimizer with a learning rate of 1e-5, a batch size of 2, and we employed binary cross entropy as the loss function. For Large Language Models, we used Unsloth AI instruction tuning method The hyperparameters for instruction tuning are crucial for optimizing the training process of a model. The per device train batch size is set to 2, warmup steps value of 5 while number of epochs is limited to 7. The learning rate is set to 2e-4, The optimizer chosen is adamw 8bit, and a weight decay of 0.01

Incongruence span We adjusted parameters such as temperature to 7 , top-k sampling as 9 , on number Dataset of samples were 429 and number of few-shot samples were 5 .

5.4 Baselines

Incongruence Detection: For this task, we selected fine-tuned pretrained language models as our baseline and we used the regular prompt tuning discussed in section 4.1 as our baseline. Incongruence span: We chose models configured for few-shot learning as our baseline approach for the extraction task. We compared these models with a multi-hop framework that was evaluated without any prior training or fine-tuning.

5.5 Evaluation Metrics

Incongruence Detection

We employ precision,recall ,F1-score, accuracy to measure the model's capability to find the contradicted pairs

Incongruence span

We evaluated three variations of precision, recall, and F1-score. INC1 Precision evaluates how accurately the model identifies the contradicting span of tokens in Text 1. INC2 Precision assesses the accuracy of the predicted span in Text 2 that contradicts Text 1. INC Precision evaluates the alignment between the predicted spans in both texts and reflects the overall precision of the model in predicting both spans accurately. Similarly, INC1 Recall measures the model's ability to identify all relevant contradicting spans in Text 1, INC2 Recall assesses its capability in Text 2, and INC Recall evaluates the alignment of predicted spans and finally F1-scores are also calculated in the same manner

- A: Total number of answers.
- C: Total number of gold label contradictory spans in an answer.
- c_a: Gold label contradictory span of answer a.
- $\text{pred}_{a,j}$: Predicted j-th contradictory span of answer a.
- $\text{precision}(c_a, \text{pred}_{a,j})$: Precision between the gold label c_a and the predicted span $\text{pred}_{a,j}$.
- $\text{recall}(c_a, \text{pred}_{a,j})$: Recall between the gold label c_a and the predicted span $\text{pred}_{a,j}$.
- $\text{F1-Score}(c_a, \text{pred}_{a,j})$: F1-Score calculated from precision and recall.
- |Predicted Span$\cap$Actual Entire Answer|: Number of tokens in the predicted span that intersect with the tokens in the actual entire answer.
- |Predicted Span|: Total number of tokens in the predicted span.

$$Precision_{avg} = \frac{1}{A}\sum_{a=1}^{A}\frac{1}{C}\sum_{c=1}^{C}\max_{j}\left(\text{precision}(c_a, \text{pred}_{a,j})\right)$$

$$Recall_{avg} = \frac{1}{A}\sum_{a=1}^{A}\frac{1}{C}\sum_{c=1}^{C}\max_{j}\left(\text{recall}(c_a, \text{pred}_{a,j})\right)$$

$$F1_{avg} = \frac{1}{A}\sum_{a=1}^{A}\frac{1}{C}\sum_{c=1}^{C}\max_{j}\left(2 \times \frac{\text{precision}(c_a, \text{pred}_{a,j}) \times \text{recall}(c_a, \text{pred}_{a,j})}{\text{precision}(c_a, \text{pred}_{a,j}) + \text{recall}(c_a, \text{pred}_{a,j})}\right)$$

$$\text{Coverage} = \frac{|\text{Predicted Span} \cap \text{Actual Entire Answer}|}{|\text{Predicted Span}|}$$

5.6 Experimental Result on Incongruence Detection

We will answer the research questions addressed in section 4.1,

5.6.1 Results of MLMs under Standard Supervised Fine-tuning Settings

Fine-tuning established a strong baseline by adapting models like LONGFORMER, BIG-BIRD, and BIRD-MNLI to the task. Among the fine-tuned models, BIG-BIRD performs the best with an F1-Score of 0.655, followed by LONGFORMER and BIRD-MNLI.The performance of the models with regular prompt tuning shows a slight improvement for BIRD-MNLI, which achieves the highest F1-Score of 0.645. LONGFORMER and BIG-BIRD have comparable performance, slightly improving from their fine-tuned versions.Question tuning method yields mixed results. LONGFORMER-TQ1 and BIRD-MNLI perform well with F1-Scores of 0.645 and 0.65, respectively. However, BIG-BIRD's performance drops to an F1-Score of 0.62. BIG-BIRD generally shows strong performance with fine-tuning and regular prompt tuning but underperforms with question prompt tuning. BIRD-MNLI performs consistently across all tuning methods, with a notable improvement in question prompt tuning. LONGFORMER shows the best results with question prompt tuning compared to other methods.

Pre-trained Models with fine-tuning			
Model	PRECISION	RECALL	F1-SCORE
LONGFORMER	0.638	0.63	0.638
BIG-BIRD	0.655	0.65	0.655
BIRD-MNLI	0.645	0.64	0.64
Pre-trained Models with regular prompt tuning			
LONGFORMER	0.643	0.64	0.64
BIG-BIRD	0.644	0.643	0.64
BIRD-MNLI	0.644	0.65	0.645
Pre-trained Models with Question prompt tuning			
LONGFORMER-TQ1	**0.65**	**0.64**	**0.645**
BIG-BIRD	0.646	0.62	0.63
BIRD-MNLI	0.64	**0.65**	0.64

Table 5.1: Performance metrics of different pre-trained models with various tuning methods.

5.6.2 Results of MLMs under low Resource settings

In real-world scenarios, collecting annotated data to finetune classification models is often time-consuming and labor-intensive. To address RQ2, we selected LONGFORMER and BIG BIRD, both fine-tuned, as our baseline models. For low-resource settings, we randomly sampled $r\%$ of instances from each class in the initial training and validation sets to create the few-shot training and validation sets, with r ranging from {1, 5, 10, 20}.

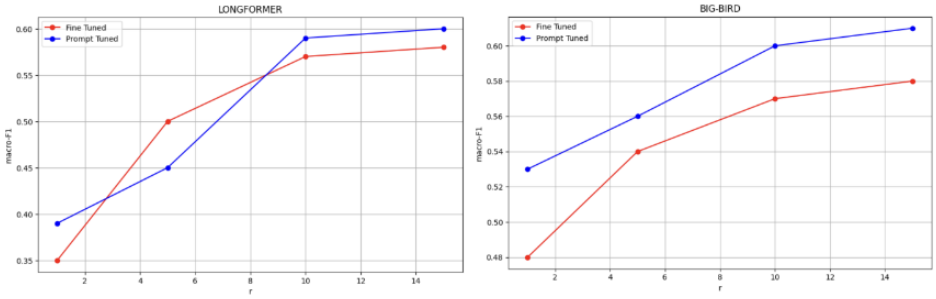

Figure 5.1: Macro F1 scores under low-resource settings, using 1%, 5%, 10%, and 15% instances

Our observations indicate that the Question prompt performed better in nearly all cases. Therefore, the effectiveness of the Question prompt is evident even in low-resource settings.

5.6.3 Results of LLMs under Instruction Tuned setting

Model	Precision	Recall	F1-Score
Gemma_7b	0.90	0.609	0.757
Mistral-7b	0.91	0.600	0.740
LLama-3	0.90	0.613	0.732
Qwen-7B	0.89	0.612	0.740

Table 5.2: Performance of LLM on Question prompt

All the large language models (LLMs) were tuned using the same question prompts, and the results are presented in the table. Additionally, we annotated 150 samples using the 6W prompts outlined in Section 4.1, instruction-tuning the LLMs and evaluating them on the test set. We observed that the LLMs achieved a very high precision rate; however, recall decreased by 2-3% compared to fine-tuned or prompt-tuned pretrained language models (PLMs). We believe this decline may result from the extensive con-

textual knowledge of the LLMs, leading to more conservative predictions. The models may prioritize stronger contextual cues indicating clear contradictions, causing them to overlook subtler contradictions that require a broader perspective or different contextual framing.The best F1-score for Question prompts achieved 75.2%

Model	**Precision**	**Recall**	**F1-Score**
Llama-3	0.7903	0.6878	0.7355
Gemma-7B	0.6800	0.7100	0.6970
Mistral-7B	0.8936	0.6992	0.7845
Qwen-7B	0.8480	0.6800	0.7550

Table 5.3: Performance metrics for various models using 6W prompts

In 6W setting, we achieved a staggering precision of 89% with best recall value of 69 and an F1-score of 78 by Mistral model. Overall all the models improved on these parameters

5.6.4 Which prompt tuning technique performs best and why?

Mistral-7b-6W achieves the highest F1-Score of 0.78, followed by Qwen-7b-6W with an F1-Score of 0.755. These scores are significantly higher than those of the other models and tuning methods.The performance of the Gemma-7b model is consistently high across TQ1 and TQ2 prompts, indicating robustness in those tuning methods, but still slightly lower than the top 6W methods. The results indicate that the 6W prompt tuning method significantly outperforms other prompt configurations, with mistral-7b-6W and Qwen-7b-6W models achieving the highest F1-Scores. Traditional TQ prompts (TQ1 and TQ2) generally perform worse compared to the 6W methods.

MODELS	F1-SCORE
LONGFORMER-TQ1	0.645
LONGFORMER-TQ2	0.623
BIG-BIRD-TQ1	0.62
BIG-BIRD-TQ2	0.59
BIRD-MNLI-TQ1	0.65
BIRD-MNLI-TQ2	0.615
Gemma_7b-TQ1	0.757
Gemma_7b-TQ2	0.756
mistral-7b-TQ1	0.719
mistral-7b-TQ2	0.74
LLama-3-TQ1	0.732
LLama-3-TQ2	0.68
Qwen-7B-TQ1	0.74
Qwen-7B-TQ2	0.578
llama-3-6W	0.73
gemma-7b-6W	0.697
mistral-7b-6W	**0.78**
Qwen-7b-6W	**0.755**

Table 5.4: F1-Scores of different models with various tuning methods.

5.7 Experimental Result on Incongruence Span Detection

We conducted experiments on 429 question-and-two-answer triplets, using Large Language Models (LLMs) with 7 and 14 billion parameters as our backbone. Additionally, we tested with GPT-4o and CHAT GPT 3.5. We compared the results of the reasoning model with few-shot learning results of the same LLMs. The evaluation was performed using precision, recall, and F1-score metrics, and the experiments were run on V100 PCIe Tesla GPUs.

5.7.1 Baselines Results on Few-shot Learning

The performance metrics for the models LLAMA-3-8B, GWEN-2 7B, GWEN-1.5 7B, MISTRAL - 7B, and GEMMA - 7B in detecting incongruence between texts indicated varied values for precision, recall, and F1-score. LLAMA-3-8B exhibits an approach with precision (0.37) and recall (0.403), achieving an F1-score (0.355). GWEN-2 7B and GWEN-1.5 7B shows higher precision (0.428) and(0.436) but lower recall (0.327),(0.335) indicating it is more accurate in identifying contradictions but may miss some.MISTRAL - 7B provides the best recall (0.464) while maintaining a balanced precision (0.387), resulting in a F1-score (0.389). GEMMA - 7B presents consistent results with precision (0.366) and recall (0.396), similar to LLAMA-3-8B.

Model	INC1				INC2				INC		
	PRECISION	RECALL	F1-SCORE	coverage	PRECISION	RECALL	F1-SCORE	coverage	PRECISION	RECALL	F1-SCORE
LLAMA-3-8B	0.396	0.412	0.367	0.952	0.385	0.412	0.357	0.951	0.37	0.403	0.355
GWEN-2 7B	0.469	0.351	0.359	0.88	0.487	0.356	0.372	0.88	0.428	0.327	0.336
GWEN-1.5 7B	0.482	0.31	0.341	0.92	0.464	0.373	0.373	0.913	0.436	0.335	0.347
MISTRAL-7B	0.416	0.456	0.392	0.89	0.412	0.5	0.405	0.928	0.387	0.464	0.389
GEMMA-7B	0.357	0.39	0.34	90.2	0.37	0.384	0.345	90.3	0.366	0.396	0.353

Table 5.5: Performance metrics for various models on Few -shot prompting

5.7.2 Results using multi-hop reasoning

The multi-hop reasoning technique demonstrates superior performance over previous few-shot techniques across most models. Notable improvements are seen in models such as Llama-3-8B, Gwen-2-7B, and Mixtral-7B. While Gemma-7B shows moderate performance improvements with this technique, its gains in precision, recall, and F1-scores are lower compared to other models.

Model	INC1				INC2				INC		
	PRECISION	RECALL	F1-SCORE	coverage	PRECISION	RECALL	F1-SCORE	coverage	PRECISION	RECALL	F1-SCORE
LLAMA	0.45	0.445	0.411	0.9	0.442	0.438	0.399	0.92	0.404	0.4	0.372
GWEN-2 7B	0.542	0.436	0.44	0.934	0.534	0.406	0.418	0.925	0.496	0.391	0.403
GWEN-1.5 7B	0.429	0.429	0.389	0.924	0.395	0.471	0.383	0.932	0.379	0.43	0.369
Gwen-14B	0.47	0.47	0.419	0.95	0.493	0.455	0.435	0.96	0.455	0.434	0.413
MISTRAL-7B	0.51	0.54	0.485	0.964	0.523	0.511	0.474	0.969	0.494	0.511	0.471
GEMMA-7B	0.384	0.399	0.356	0.943	0.401	0.397	0.364	0.945	0.365	0.395	0.344
chatGPT	0.544	0.598	0.522	0.996	0.52	0.56	0.496	0.987	0.499	0.592	0.49
GPT 4o	0.612	0.708	0.626	0.998	0.566	0.674	0.565	0.999	0.526	0.695	0.555

Table 5.6: Performance metrics for various models on multi-hop reason prompting

5.7.3 Performance with CHAT-GPT and GPT4o

We compared the performance of the multi-hop reasoning method on ChatGPT 3.5 and GPT-4o. Figure 5.1 illustrates the results from testing on 100 instances. Both models show a slight improvement in performance when using the multi-hop reasoning method.

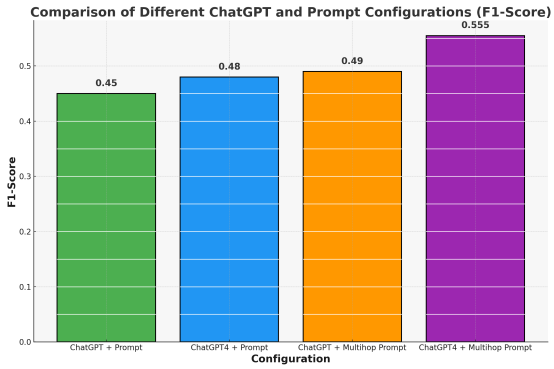

Figure 5.2: Comparison between CHAT GPT 3.5 AND CHAT GPT 4o on 100 random instances

5.7.4 Failure Analysis

We performed a failure analysis on the outputs of three models: Llama-7B, Mixtral-7B, and GPT-4o. Our analysis focused on 30 erroneous cases, each of which had an F1-score below 0.5.

We categorized the errors into three types:

- **Reasoning Errors**: These occur when the models fail to reason and identify contradictions.
- **Boundary Errors**: These occur when the models reason correctly but provide only partial answers.
- **Inference Errors**: These occur when the models fail to infer correctly based on instructions, even though the reasoning itself is correct. An example is when the inferred output does not follow the provided format.

In our analysis, Llama-7B failed to analyze and reason out the contradictions in 48% of the cases. This failure rate decreases to 38% for Mixtral-7B and 34% for GPT-4o. The majority of failures for GPT-4o and Mixtral-7B are due to boundary errors. This indicates that while these models can identify the location of contradictions, they either fail to fully address the contradiction or provide more information than necessary.

Lastly, the percentage of inference errors for each model is relatively low compared to other errors. These errors occur when the models do not adhere to the format specified in the instructions and fail to extract the exact span. Instead of providing the precise span, the models continue reasoning in the third iteration, carrying forward the reasoning from the second iteration, which results in inference errors.

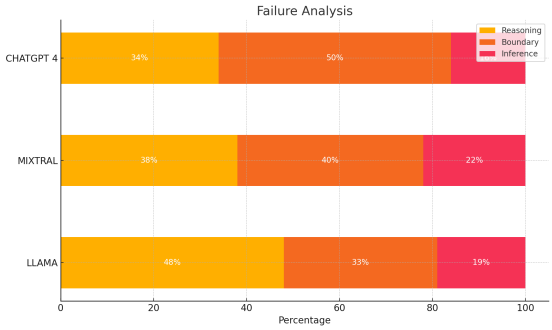

Figure 5.3: Failure Analysis

5.7.5 Human Evaluation

We introduced four metrics for human evaluation and evaluated our GPT 4o + multi-hop prompt output on these metrics:

1. **Contradiction Clarity**: Evaluates whether the statements clearly and directly contradict each other.
2. **Logical Exclusivity**: Assesses whether the statements are logically exclusive, presenting mutually exclusive facts or states that cannot both be true simultaneously.
3. **Context Relevance**: Determines whether the contradiction is highly relevant to the context or question at hand. This metric checks if the statements address the same topic or detail.
4. **Coverage**: Measures whether the statements comprehensively cover all necessary aspects. This includes addressing all key elements to provide a complete understanding of the contradiction.

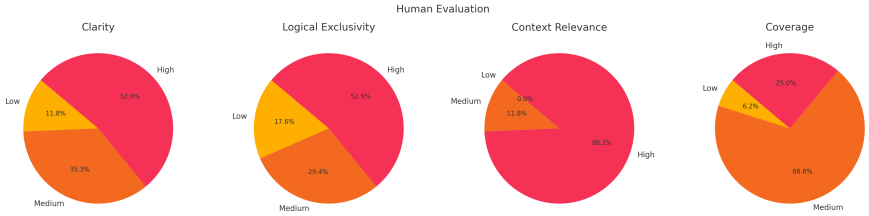

Figure 5.4: Human Evaluation

5.8 Conclusion

In this thesis, we successfully created a novel multimodal deception detection dataset, which facilitated the introduction of two significant tasks: Incongruence detection and Incongruence span detection. By establishing a robust baseline using machine learning models (MLM) and large language models (LLM), we demonstrated the efficacy of various prompt templates in uncovering implicit incongruence within testimonial statements. Additionally, we explored and assessed the effectiveness of few-shot and multi-hop reasoning methods and providing a comprehensive analysis of the performance of multi-hop reasoning frameowrk.

Future work can propose several advancements in the areas of deception detection using our dataset. Significant improvements can be made in the span detection model, enhancing its accuracy and reliability. Additionally, introducing explainability by achieving a better semantic understanding of statements and incorporating further reasoning steps will be crucial. These enhancements will not only refine the current methodologies but also provide deeper insights and more transparent results in the field of deception detection.

Our findings contribute valuable insights and methodologies to the field of deception detection, paving the way for future advancements and applications.

REFERENCES

1. **Buller, D.** and **J. Burgoon** (2006). Interpersonal deception theory. *Communication Theory*, **6**, 203 – 242.

2. **de Marneffe, M.-C.**, **A. N. Rafferty**, and **C. D. Manning**, Finding contradictions in text. *In* **J. D. Moore**, **S. Teufel**, **J. Allan**, and **S. Furui** (eds.), *Proceedings of ACL-08: HLT*. Association for Computational Linguistics, Columbus, Ohio, 2008. URL https://aclanthology.org/P08-1118.

3. **Ekman, P.** and **W. V. Friesen** (1969). Nonverbal leakage and clues to deception†. *Psychiatry*, **32**(1), 88–106. URL https://doi.org/10.1080/00332747.1969.11023575. PMID: 27785970.

4. **Fei, H.**, **B. Li**, **Q. Liu**, **L. Bing**, **F. Li**, and **T.-S. Chua**, Reasoning implicit sentiment with chain-of-thought prompting. *In* **A. Rogers**, **J. Boyd-Graber**, and **N. Okazaki** (eds.), *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. Association for Computational Linguistics, Toronto, Canada, 2023. URL https://aclanthology.org/2023.acl-short.101.

5. **Fernandes, S. V.** and **M. S. Ullah**, Development of spectral speech features for deception detection using neural networks. *In 2021 IEEE 12th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*. 2021.

6. **Greenspan, R. L.**, **A. Lyman**, and **P. Heaton** (2024). Assessing verbal eyewitness confidence statements using natural language processing. *Psychological Science*, **35**(3), 277–287. URL https://doi.org/10.1177/09567976241229028. PMID: 38376954.

7. **Gupta, V.**, **M. Agarwal**, **M. Arora**, **T. Chakraborty**, **R. Singh**, and **M. Vatsa**, Bag-of-lies: A multimodal dataset for deception detection. *In 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. 2019.

8. **Levitan, S.**, **A. Maredia**, and **J. Hirschberg**, Linguistic cues to deception and perceived deception in interview dialogues. 2018.

9. **Liu, P.**, **W. Yuan**, **J. Fu**, **Z. Jiang**, **H. Hayashi**, and **G. Neubig** (2021). Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *CoRR*, **abs/2107.13586**. URL https://arxiv.org/abs/2107.13586.

10. **Liu, Y.**, **R. Zhang**, **Y. Fan**, **J. Guo**, and **X. Cheng**, Prompt tuning with contradictory intentions for sarcasm recognition. *In* **A. Vlachos** and **I. Augenstein** (eds.), *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics, Dubrovnik, Croatia, 2023. URL https://aclanthology.org/2023.eacl-main.25.

11. **Marneffe, M.-C.**, **A. Rafferty**, and **C. Manning**, Finding contradictions in text. 2008.

12. **Mihalcea, R.** and **C. Strapparava**, The lie detector: Explorations in the automatic recognition of deceptive language. *In* **K.-Y. Su**, **J. Su**, **J. Wiebe**, and **H. Li** (eds.), *Proceedings of the ACL-IJCNLP 2009 Conference Short Papers*. Association for Computational Linguistics, Suntec, Singapore, 2009. URL https://aclanthology.org/P09-2078.

13. **Mott, F. L.** (1942). Trends in newspaper content. *The Annals of the American Academy of Political and Social Science*, **219**, 60–65. Accessed on Jan 10, 2023.

14. **Pérez-Rosas, V.**, **M. Abouelenien**, **R. Mihalcea**, and **M. Burzo**, Deception detection using real-life trial data. *In Proceedings of the 2015 ACM on International Conference on Multimodal Interaction*, ICMI '15. Association for Computing Machinery, New York, NY, USA, 2015. ISBN 9781450339124. URL https://doi.org/10.1145/2818346.2820758.

15. **Pérez-Rosas, V.**, **M. Abouelenien**, **R. Mihalcea**, **Y. Xiao**, **C. Linton**, and **M. Burzo**, Verbal and nonverbal clues for real-life deception detection. *In* **L. Màrquez**, **C. Callison-Burch**, and **J. Su** (eds.), *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Lisbon, Portugal, 2015. URL https://aclanthology.org/D15-1281.

16. **Solà-Sales, S.**, **C. Alzetta**, **C. Moret-Tatay**, and **F. Dell'Orletta** (2023). Analysing deception in witness memory through linguistic styles in spontaneous language. *Brain Sciences*, **13**(2). ISSN 2076-3425. URL https://www.mdpi.com/2076-3425/13/2/317.

17. **University, C.**, **S. International**, and **U. of Colorado Boulder** (2013). Csc deceptive speech ldc2013s09. Web Download. Web Download. Philadelphia: Linguistic Data Consortium.

18. **Wei, J.**, **X. Wang**, **D. Schuurmans**, **M. Bosma**, **E. H. Chi**, **Q. Le**, and **D. Zhou** (2022). Chain of thought prompting elicits reasoning in large language models. *CoRR*, **abs/2201.11903**. URL https://arxiv.org/abs/2201.11903.

19. **Wu, X.**, **K.-H. Huang**, **Y. Fung**, and **H. Ji**, Cross-document misinformation detection based on event graph reasoning. *In* **M. Carpuat**, **M.-C. de Marneffe**, and **I. V. Meza Ruiz** (eds.), *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, Seattle, United States, 2022. URL https://aclanthology.org/2022.naacl-main.40.

20. **Zuckerman, M.**, **B. M. DePaulo**, and **R. Rosenthal**, Verbal and nonverbal communication of deception11preparation of this article was supported in part by a grant from the office of naval research (n00014-76-c-0001), the national science foundation, the university of virginia research council, and the national institute of mental health. this article is dedicated to all those researchers who, in response to our requests, went back to their old data files, their old computer outputs, and the backs of their old envelopes in order to provide us with the necessary data for the quantitative summaries that are reported. volume 14 of *Advances in Experimental Social Psychology*. Academic Press, 1981, 1–59. URL https://www.sciencedirect.com/science/article/pii/S006526010860369X.