

Claim Identification and Verification by Leveraging Prompts and Deep Learning Techniques

submitted by

Anjani Ahir

Class Roll No: 002210503032

Exam Roll: MCA244031

Master of Computer Application

under the supervision of

Dr Dipankar Das

in the

Department of Computer Science and Engineering

Faculty of Engineering and Technology

Jadavpur University

2024

Faculty of Engineering and Technology
Jadavpur University

CERTIFICATE OF RECOMMENDATION

To Whom It May Concern

I hereby recommend that the project report entitled "Claim Identification and Verification by Leveraging Prompts and Deep Learning Techniques" has been carried out by Anjani Ahir (Reg. No.: 163624 of 2022-23, Roll No.: 002210503032, Exam Roll: MCA244031), under my guidance and supervision and be accepted in partial fulfilment of the requirement for the degree of Master of COMPUTER APPLICATION in DEPARTMENT of COMPUTER SCIENCE and ENGINEERING, JADAVPUR UNIVERSITY during the academic year 2023-2024.

Dr. Dipankar Das
Project Supervisor
Dept. of Computer Science and
Engineering
Jadavpur University

Prof. (Dr.) Nandini Mukhopadhyay
Head of the Department
Dept. of Computer Science & Engineering
Jadavpur University

Prof. Dipak Laha
Dean, Faculty Council
of Engineering & Technology
Jadavpur University, Kolkata-032

Faculty of Engineering and Technology
Jadavpur University

CERTIFICATE OF APPROVAL

This is to certify that the project report entitled “**Claim Identification and Verification by Leveraging Prompts and Deep Learning Techniques**” is a bonafide record of work carried out by ANJANI AHIR, in fulfillment of the requirements for the award of the degree of Master of Computer Application in the Department of Computer Science and Engineering, Jadavpur University. It is understood that by this approval the undersigned do not necessarily endorse or approve any statement made, opinion expressed or conclusion drawn therein but approve the project report only for the purpose for which it has been submitted.

Signature of Examiner 1

Date :

Signature of Examiner 2

Date :

Faculty of Engineering and Technology
Jadavpur University

**DECLARATION OF COMPLIANCE WITH
ACADEMIC ETHICS**

I hereby declare that this project report entitled "Claim Identification and Verification by Leveraging Prompts and Deep Learning Techniques" is an outcome of my genuine and independent work, as part of my Master of Computer Application program.

All information in this document has been obtained and presented in accordance with academic rules and ethical conduct.

I also declare that, as required by these rules and conduct, all materials and results that are not original to this work have been fully cited and referenced.

Anjani Ahir

Registration No.: 163624 of 2022-23

Examination Roll No.: MCA244031

Thesis Title :- Claim Identification and Verification by Leveraging Prompts and Deep Learning Techniques

Signature with Date

ACKNOWLEDGEMENT

With my most sincere and gratitude, I would like to thank Dr.Dipankar Das, Department of Computer Science & Engineering, my supervisor, for his overwhelming support throughout the duration of the project. His motivation always gave me the required inputs and momentum to continue with my work, without which the project work would not have taken its current shape. His valuable suggestions and numerous discussions have always inspired new ways of thinking. I feel deeply honored that I got this opportunity to work under him.

I would like to express my sincere thanks to all my teachers for providing sound knowledge base and cooperation.

I would like to thank all the faculty members of the Department of Computer Science & Engineering of Jadavpur University for their continuous support.

Last, but not the least, I would like to thank my batch mates for staying by my side when I needed them.

Anjani Ahir

Class Roll No.: 002210503032

Examination Roll No.: MCA244031

Department of Computer Science & Engineering

Jadavpur University

TABLE OF CONTENTS

List of Figures	I
List of Tables.....	I
Abstract	II
Chapter 1: Introduction.....	1
1.1 Overview.....	1
1.2 Background and Motivation	2
1.3 Objective.....	3
1.4 Contributions.....	3
1.5 Outline of the Report	3
 Chapter 2: Literature Survey.....	 5
 Chapter 3: Methodology.....	 7
3.1 Problem Statement.....	7
3.2 Dataset Preparation	7
3.2.1 Prompt Engineering	7
3.2.2 Few-Shot Prompting	11
3.2.3 Dataset used in Experiment.....	14
3.2.4 Dataset Statistics	16
3.3 Proposed Technique/Approach.....	19
3.3.1 BiLSTM (Bidirectional Long Short-Term Memory).....	19
3.3.2 BERT(Bidirectional Encoder Representations from Transformers).....	21
 Chapter 4: Experiments and Results.....	 23
4.1 Experimental Setup.....	23
4.1.1 Prompt Engineering	23
4.1.2 BiLSTM Model	25
4.1.3 BERT Model	27
4.2 Results	29
4.2.1 Results for Claim Classification	29
4.2.2 Results for Claim Verifiability.....	30

4.2.3 Comparing the performance of BiLSTM and BERT	31
4.2.4 Results on Balanced Dataset.....	32
4.3 Error Analysis.....	33
4.4 Discussion	37
Chapter 5: Conclusion and Future Work.....	38
References.....	39

List Of Figures

Figure Name	Page No.
Fig1: BiLSTM Architecture ¹	20
Fig2: BERT Architecture ²	22

List Of Tables

Table Name	Page No.
Table 1: Snapshot of the Dataset	16
Table 2: Claim Distribution by Label	17
Table 3: Claim Distribution by Topic	17
Table 4: Claim Distribution by Topic and Label	17
Table 5: Claim Distribution by Verifiability and Topic	18
Table 6: Dataset Features	18
Table 7: Claim Classification results using BiLSTM	29
Table 8 : Claim Classification results using BERT	30
Table 9: Claim Verifiability results using BiLSTM	30
Table 10: Claim Verifiability results using BERT	31
Table 11: Result Comparison of BiLSTM and BERT	31
Table 12: Claim Classification results using BiLSTM on balanced dataset	32
Table 13: Example of misclassified claims by BiLSTM	34
Table 14: Example of misclassified claims by BERT	35
Table 15: Example of misclassified claim verifiability by BiLSTM	36
Table 16: Example of misclassified claim verifiability by BERT	37

¹ <https://www.baeldung.com/wp-content/uploads/sites/4/2022/01/bilstm-1.png>

² <https://www.researchgate.net/profile/Phillip-Schoenfelder/publication/355938108/figure/fig2/AS:1086781453807616@1636120298039/Simplified-high-level-view-of-the-BERT-architecture-7-8-The-word-tokens-are.png>

ABSTRACT

The intersection of prompt engineering and deep learning represents a promising frontier in the field of identification and verification, particularly in the context of claim verification. This research focuses on creating a comprehensive claim verification dataset using advanced prompt engineering techniques. The goal is to harness the power of prompt engineering to enhance the initial stages of data preparation, which is crucial for training effective deep learning models. By leveraging prompt engineering, the research aims to streamline the process of claim generation and annotation, reducing the time and effort required while improving the quality and diversity of the dataset.

Subsequently, deep learning models are employed to classify claims and identify supporting or refuting text spans. Evaluating these models against the prompt-engineered dataset, the study demonstrates how prompt engineering improves model accuracy, efficiency, and interpretability. This integrated approach offers a robust solution to misinformation, ensuring the dissemination of reliable information and showcasing the synergistic potential of prompt engineering and deep learning in claim verification.

CHAPTER 1: INTRODUCTION

1.1. Overview

The proliferation of misinformation and the need for reliable information sources have underscored the importance of robust identification and verification systems. This paper investigates a novel approach to claim verification by leveraging prompt engineering and deep learning techniques. The proposed methodology involves creating a claim verification dataset using prompt engineering, which is subsequently used to determine the class of the claim and identify relevant text spans. This dual approach is then compared with traditional deep learning models to evaluate performance and efficacy.

In the first phase, prompt engineering is employed to construct a comprehensive dataset for claim verification. Prompts are designed to elicit specific responses that categorize claims and highlight critical text spans within the documents. These prompts guide the data annotation process, ensuring consistency and relevance in the dataset.

Next, the same task is approached using advanced deep learning models. These models are trained on the dataset generated through prompt engineering. The deep learning models aim to classify the claims into categories such as SUPPORTS, REFUTS or NOT ENOUGH INFO.

The comparative analysis reveals that prompt engineering can significantly enhance the performance of deep learning models. Prompts serve as a preprocessing step that simplifies the complexity of the input data, enabling the models to focus on pertinent information. The deep learning models, when trained on the prompt-engineered dataset, exhibit improved accuracy in claim classification and text span identification compared to training on raw data.

1.2. Background and Motivation

The exponential growth of digital information has led to an increase in misinformation and disinformation, which can have significant social, political, and economic impacts. In response, the field of claim verification has gained substantial attention, aiming to develop systems that can automatically verify the veracity of claims made in various texts.

Traditional approaches to claim verification have predominantly relied on rule-based systems and manual fact-checking, which are labour-intensive and not scalable. Recent advancements in artificial intelligence, particularly deep learning, have introduced new methodologies for automating the verification process. Deep learning models, such as those based on transformer architectures, have demonstrated remarkable capabilities in natural language understanding and generation. However, these models often require large amounts of annotated data and significant computational resources.

Additionally, their black-box nature makes it challenging to interpret their decisions, which is a critical requirement in high-stakes applications. The motivation behind this research stems from the need to improve the efficiency and effectiveness of claim verification systems. Key drivers include:

1. **Combating Misinformation:** With the proliferation of fake news and misleading information, there is an urgent need for automated systems that can quickly and accurately verify claims. Effective claim verification can mitigate the spread of misinformation and its potential harm.
2. **Enhancing Deep Learning Models:** While deep learning models are powerful, they often struggle with understanding the context and nuances in natural language. Prompt engineering can guide these models, improving their performance by focusing on relevant aspects of the input data.
3. **Improving Dataset Quality:** High-quality, annotated datasets are crucial for training effective deep learning models. Prompt engineering offers a systematic way to generate such datasets, ensuring consistency and relevance in the data.
4. **Interpretable AI:** There is a growing demand for AI systems that provide interpretable and explainable outputs. By integrating prompt engineering with deep learning, it is possible to create systems that not only perform well but also offer insights into their decision-making processes.

1.3 Objective

This report aims to design an effective system for claim identification and verification by integrating prompt engineering and deep learning techniques. The primary goal is to generate a diverse and comprehensive dataset through sophisticated prompt engineering, which will be used to classify claims and extract pertinent evidence. By training deep learning models on this dataset, the study seeks to evaluate their effectiveness in accurately verifying claims. The objective is to improve the precision, efficiency, and clarity of claim verification methods, thereby tackling the issues of misinformation and ensuring the dissemination of trustworthy information.

1.4 Contributions

This work makes significant contributions to the field of claim verification.

First, a custom claim verification dataset was developed using advanced prompt engineering techniques. This dataset serves as a critical resource for training and evaluating machine learning models in this domain.

Second, state-of-the-art deep learning techniques, specifically BERT and BiLSTM, were applied to this dataset. These models were selected for their proven effectiveness in natural language processing tasks.

Finally, a comprehensive comparative analysis of the results obtained from these models was conducted, providing valuable insights into their performance and identifying potential areas for improvement.

1.5 Outline of the Report

The project consists of five main chapters outlining a comprehensive study on claim verification leveraging prompt engineering and deep learning techniques.

The literature survey in Chapter 2 reviews existing research on prompt engineering, claim verification, and deep learning, highlighting seminal works such as Petroni et al. (2020), Thorne et al. (2018), and Vaswani et al. (2017).

Chapter 3, Methodology, delineates the problem statement, dataset preparation methods including prompt engineering and few-shot prompting, and introduces the proposed techniques such as BiLSTM and BERT models.

Chapter 4, Experiments and Results, Detailed explanation of the experimental setup, including prompt engineering and model configurations has been discussed in this section of the report. It also presents the results for claim classification and

verifiability tasks, along with a comparative analysis of BiLSTM and BERT models. Furthermore, error analysis and discussion of results are provided.

Lastly, Chapter 5 concludes the study, summarizing key findings and suggesting future research directions, followed by a comprehensive list of references.

CHAPTER 2: LITERATURE SURVEY

1. Prompt Engineering in NLP:

Prompt engineering has emerged as a powerful technique in natural language processing (NLP), enabling fine-tuning of large pre-trained language models (PLMs) for specific downstream tasks.

- Petroni et al. (2020) pioneered the concept by demonstrating the effectiveness of prompt-based fine-tuning in few-shot learning scenarios.
- Chen et al. (2023) explores the transformative potential of prompt engineering within the realm of large language models (LLMs). Various advanced techniques like Few-Shot Prompting has been discussed here.

2. Claim Verification:

Claim verification, a crucial task in combating misinformation, has gained significant attention in recent years.

- Thorne et al. (2018) introduced the FEVER dataset, laying the groundwork for automated fact extraction and verification tasks.
- Wang et al. (2019) extended this work to scientific literature with the SciFact dataset, fostering research in claim verification within specialized domains.

3. State-of-the-Art Approaches:

- The transformative architecture of the transformer based model was introduced here [Vaswani et al. \(2017\)](#) that made a significant impact.
- Recent advancements in claim verification have seen the adoption of transformer-based models like BERT (Devlin et al., 2019). Hanselowski et al. (2020) demonstrated the effectiveness of BERT in claim verification tasks, emphasizing the importance of evidence selection and model fine-tuning.

5. Contributions of the Proposed Paper:

The paper "Claim Identification and Verification by Leveraging Prompts and Deep Learning Techniques" builds upon this existing literature by proposing a

novel approach that leverages prompt engineering techniques to enhance claim identification and verification tasks. By integrating prompts with deep learning models, the paper aims to improve accuracy, efficiency, and interpretability in claim verification, addressing the challenges posed by misinformation and the need for reliable information.

Overall, the integration of prompt engineering and deep learning techniques represents a promising frontier in claim verification research, offering a robust solution to combat misinformation and promote the dissemination of accurate information.

CHAPTER 3: METHODOLOGY

3.1 Problem Statement

The proliferation of misinformation in digital media has created an urgent need for effective claim verification systems. Traditional methods of claim verification are often time-consuming and labour-intensive, making it difficult to keep pace with the rapid dissemination of information. Furthermore, existing automated systems may lack accuracy and interpretability, leading to unreliable results. This report addresses the challenge of improving the efficiency, accuracy, and interpretability of claim verification by integrating prompt engineering with deep learning techniques. The goal is to develop a robust system capable of generating and verifying claims efficiently while providing clear evidence for its classifications.

3.2 Dataset Preparation

This report proposes a novel approach that involves prompt engineering for the creation of a dataset specifically designed for claim verification and claim check worthiness. By leveraging several techniques of prompting, the dataset was meticulously constructed to encompass a wide range of claim types and scenarios, ensuring comprehensive coverage and robustness.

3.2.1 Prompt Engineering

A diverse and comprehensive claim verification dataset has been prepared using advanced prompt engineering techniques, including Few-Shot Prompting. This methodology allows for the creation of a structured collection of claim instances across five diverse topics: Finance, Healthcare, Technology, Indian Parliament, and Indian Politics.

The dataset is meticulously crafted to ensure a variety of unique claims, each accompanied by relevant evidence. Claims are categorized into three distinct classes: "SUPPORTS," "REFUTES," and "NOT ENOUGH INFO." Each claim is further annotated

with verifiability status and evidence sources, ensuring the dataset's robustness and reliability.

Key characteristics of the dataset include:

- **Diverse Topics:** The inclusion of claims across diverse topics like Finance, Healthcare, Technology, Indian Parliament, and Indian Politics ensures a broad spectrum of verification instances.
- **Structured Classification:** Claims are systematically classified into "SUPPORTS," "REFUTES," and "NOT ENOUGH INFO," facilitating clear and precise claim verification.
- **Annotated Evidence:** Each claim is supported or refuted by meticulously annotated evidence, enhancing the dataset's credibility and utility for training deep learning models.
- **Verifiability Status:** Every claim in the dataset is annotated with verifiability status that indicates whether the claim is verifiable or not verifiable.

Guide to Crafting Prompts for Preparation of Claim Verification

Dataset: The dataset creation for claim verification involves a structured methodology that ensures the generation of unique, verifiable claims across various topics, accompanied by corresponding evidence and labels. The following steps outline the comprehensive approach used for crafting the prompt for dataset preparation:

1. **Task Specification:** Providing clear and specific instructions of the task within the prompt helps guide the model towards generating the desired output. Ambiguity in the prompt can lead to responses that may not fulfil the desired expectations.
2. **Define the Dataset Structure and Fields:** The dataset is designed to include the following fields:
 - **Verifiable:** Indicates whether the claim can be verified, with possible values being "Verifiable," "Not Verifiable," or "Not Enough Information."
 - **Claim:** The statement that requires verification.
 - **Evidence:** Supporting text, links, or documents that provide context and information about the claim.
 - **Label:** The categorization of the claim based on the evidence, with possible values being "SUPPORTS," "REFUTES," or "NOT ENOUGH INFO."
 - **Source:** The source from which the evidence is derived.

3. **Topic Selection:** Select diverse topics to ensure the dataset covers a wide range of subject areas. Topics can include technology, health, politics, science, and more. Each topic serves as a basis for generating claims.
4. **Ask for Structured Output:** Mention the format of the output to generate the dataset. Here in this report the dataset is organized in a structured JSON format with the specified fields: verifiable, claim, evidence, label, and source.
5. **Provide Examples:** Providing a few examples of the desired output enable in-context learning where we provide demonstrations in the prompt to steer the model to better performance. These examples demonstrate the expected behaviour, output, or task before presenting the model with new, related input. This approach, known as Few-Shot Prompting, allows the model to build an understanding of what is expected even with limited context.

Example Prompt:

```
prompt = f"""
```

Task: Dataset Creation for Claim Verification

Goal: Generate a dataset for claim verification tasks across various topics, ensuring unique claims with corresponding evidence and labels.

Fields:

verifiable: Either the claim is verifiable or not verifiable or not enough information

Claim: The statement to be verified.

Evidence: Supporting evidence (text, links, documents) for the claim.

Label: either SUPPORTS, REFUTES or NOT ENOUGH INFO

Instructions:

Topic: [Topic]

verifiable: Either the claim is verifiable or not verifiable or not enough information

Generate Claims: For each topic, craft unique claims that can be either true or false. Ensure claims cover a spectrum of complexity and relevance.

Gather Evidence: Find or create supporting evidence (articles, research papers, official statements, etc.) for each claim.

The evidence should clearly support or refute the claim.

Assign Labels: Based on the provided evidence, assign labels to each claim.

Ensure Uniqueness: Verify that all claims within the dataset are distinct and do not overlap in content or meaning.

Format: Organize the dataset in a structured JSON format with the specified fields (Claim, Evidence, Label).

Example: [provide a few examples of the exact output]

```
"""
```

Guide to Crafting Prompts for Text Span Identification of Claim:

1. **Instruction for Classification:** Provide clear instructions on how to classify the given claim. Emphasize the importance of considering the evidence provided to make an informed decision.
2. **Step-by-Step Approach:** Encourage the user to think step by step and analyze the claim and evidence carefully. Break down the process into manageable steps to guide them through the task effectively.
3. **Claim and Evidence Presentation:** Present the claim, evidence, and the source of the claim in a clear and structured format. Ensure that these elements are easily identifiable and distinguishable within the prompt.
4. **Output Format:** Specify the format in which the output should be provided. Mention that it should include a justification for the classification decision and the corresponding text span that supports or refutes the claim, or indicates insufficient information.
5. **Prompt Example:** Provide a template prompt example that incorporates all the aforementioned elements. Use placeholders like {claim}, {evidence}, and {source} to indicate where the actual claim, evidence, and source should be inserted.
6. **Encouragement for Critical Thinking:** Encourage critical thinking and thorough analysis of the claim and evidence. Remind the user to consider various perspectives and evaluate the credibility of the information provided.
7. **Conclusion:** Conclude by reiterating the importance of careful examination and critical analysis in claim verification. Encourage users to provide thorough justifications supported by relevant text spans.

By following this guide, you can craft prompts that effectively guide users through the process of identifying text spans for claim classification while promoting critical thinking and thorough analysis.

Example Prompt:

```
prompt = f"""
Now Can you fact-check a claim for me? Classify the given claim into 3 labels: "Supports",
"Refutes"
and in case there is very less information provided about a claim classify it as "Not Enough
Info". Let's think step by step.
First print the classification and then Provide justification as well as text span for which it is
supports or refutes or not enough info.
Given claim:
```${claim}```
Evidence of that claim : ```${evidence}```
Source of the Claim : ```${source}```
Now analyze the claim and evidence carefully and provide the output in the given format.
Think step by step.
Output Format :
Justification :
Text Span for that justification :
"""
```

### 3.2.2 Few-Shot Prompting

Few-shot prompting is a technique that leverages the capabilities of Large Language Models (LLMs) to perform specific tasks. By providing a few examples, known as "shots," you can condition the model to generate desired outputs, be it text, code, or images.

While LLMs are impressive in their zero-shot capabilities, they often struggle with more complex tasks. Few-shot prompting serves as in-context learning, guiding the model towards better performance by offering demonstrations within the prompt. The effectiveness of few-shot prompting can vary depending on the number of shots provided—1-shot, 3-shot, 5-shot, etc. However, it's worth noting that this technique has limitations in handling specific reasoning tasks, making advanced prompt engineering essential for optimal results.

Few-shot prompting has many applications in educational publishing, including:

- **Natural Language Understanding:** To enhance sentiment analysis in e.g. product review.
- **Question Answering:** For improving the generation of Q&A sections in textbooks.
- **Summarisation:** To guide the model in producing concise and compelling chapter summaries.

## Example of few-shot prompt used in dataset preparation.

### Topic: HealthCare

**verifiable:** "Verifiable",

**claim:** " Eating a diet rich in fruits and vegetables lowers the risk of developing chronic diseases.",

**evidence:** " Numerous studies show that a diet high in fruits and vegetables is associated with a reduced risk of chronic diseases.",

**label:** "SUPPORTS",

**source:** " Harvard School of Public Health"

**verifiable:** "Not Verifiable",

**claim:** "All mental illnesses will be treated with personalized brain stimulation therapies by 2040. ",

**evidence:** "The widespread use of personalized brain stimulation therapies for all mental illnesses is speculative.",

**label:** "NOT ENOUGH INFO",

**source:** " Nature Neuroscience"

### Topic: Technology

**verifiable:** "Verifiable",

**claim:** " Edge computing introduces new security risks and compliance challenges, including data privacy concerns, unauthorized access, and regulatory compliance issues.",

**evidence:** " Edge computing platforms leverage local computing resources to process sensor data and execute applications at the network edge, enhancing responsiveness and scalability for latency-sensitive IoT deployments.",

**label:** "REFUTES",

**source:** "Forbes"

**verifiable:** "Verifiable",

**claim:** " The development of quantum-resistant encryption standards ensures the long-term security of digital communication and protects sensitive information from decryption by quantum computers, safeguarding data privacy and integrity in the quantum era. ",

**evidence:** " Quantum-resistant cryptography algorithms, such as lattice-based cryptography and hash-based signatures, are designed to withstand quantum attacks and provide robust security guarantees for cryptographic protocols, ensuring resilience against emerging threats to digital security.",

**label:** "SUPPORTS",

**source:** " European Telecommunications Standards Institute (ETSI)"

## **Topic: Finance**

**verifiable:** "Verifiable",

**claim:** " Exchange-traded funds (ETFs) generally have lower fees than mutual funds.",

**evidence:** " The Efficient Market Hypothesis posits that it is impossible to consistently achieve higher returns than average market returns on a risk-adjusted basis.",

**label:** "SUPPORTS",

**source:** " Journal of Economic Perspectives"

**verifiable:** "Not Verifiable",

**claim:** " The next global financial crisis will be caused by a major cryptocurrency collapse. ",

**evidence:** " Predicting the exact cause of the next financial crisis is speculative.",

**label:** "NOT ENOUGH INFO",

**source:** "Bloomberg"

## **Topic: Indian Politics**

**verifiable:** "Verifiable",

**claim:** " India has the largest number of official languages in the world.",

**evidence:** " India recognizes 22 official languages under the Eighth Schedule of the Indian Constitution, but this is not the largest number in the world.",

**label:** "REFUTES",

**source:** " Constitution of India"

**verifiable:** "Not Enough Info",

**claim:** " India's defence budget is larger than that of Japan. ",

**evidence:** "Detailed and current comparative data on defence budgets is necessary to confirm this claim and is not readily available.",

**label:** "NOT ENOUGH INFO",

**source:** " SIPRI"

## **Topic: Indian Parliamentary**

**verifiable:** "Verifiable",

**claim:** " The Indian Parliament consists of two houses: the Lok Sabha and the Rajya Sabha.",

**evidence:** " The Indian Constitution defines the Parliament as consisting of the President, the Lok Sabha, and the Rajya Sabha.",

**label:** "SUPPORTS",

**source:** "Indian Constitution"

**verifiable:** "Not Verifiable",  
**claim:** " Indian Parliamentarians are immune to external pressures. ",  
**evidence:** "Indian Parliamentarians may face external pressures that influence their decisions.",  
**label:** "REFUTES",  
**source:** "The Week"

### 3.2.3 Dataset Used in Experiment

The following dataset that has been prepared through prompt engineering techniques. This dataset has been used to train the deep learning models discussed in this report. The snapshot of the dataset is given in Table 1.

Topic	verifiable	claim	evidence	label	Source
HealthCare	Verifiable	"Regular physical activity reduces the risk of developing cardiovascular diseases."	<i>Multiple studies confirm that regular exercise lowers the risk of heart disease and stroke.</i>	SUPPORTS	American Heart Association
	Verifiable	"All vaccines provide 100% immunity from the diseases they target."	<i>No vaccine provides 100% immunity; their effectiveness varies.</i>	REFUTES	CDC
	Not Verifiable	"Medical tricorders will be in common use for diagnosing illnesses by 2035".	<i>The future use of medical tricorders for common diagnostics is speculative.</i>	NOT ENOUGH INFO	IEEE Spectrum
Finance	Verifiable	"Treasury bonds are considered risk-free investments."	<i>Treasury bonds, also known as T-bonds or T-bills, are issued by the U.S. government and are backed by its full faith and credit.</i>	SUPPORTS	TreasuryDirect
	Verifiable	Inflation has consistently been above 5% in the United States over the past decade.	<i>U.S. inflation rates have generally been below 5% over the past decade, with few exceptions.</i>	REFUTES	Bureau of Labor Statistics (BLS)

	Not Verifiable	The rise of fintech companies will lead to the disappearance of all traditional banks by 2040.	<i>While fintech is growing, predicting the complete disappearance of traditional banks is speculative.</i>	NOT ENOUGH INFO	Forbes
	Not Verifiable	No Indian government policy has ever benefited the economy.	<i>Various policies have had differing impacts on the economy.</i>	REFUTES	General Knowledge
<b>Tcehnology</b>	Verifiable	5G networks offer faster data speeds and lower latency, enabling advancements in autonomous vehicles and remote surgeries.	<i>5G technology promises faster data transmission and reduced latency, which are crucial for real-time applications like remote surgeries and autonomous driving.</i>	SUPPORTS	BBC News
	Not Verifiable	The rise of automation and robotics will lead to mass unemployment globally.	<i>While automation may disrupt certain job sectors, it can also create new job opportunities and increase productivity in others, leading to mixed economic impacts.</i>	NOT ENOUGH INFO	World Economic Forum
	Verifiable	Machine learning algorithms are susceptible to biases and errors, leading to inaccurate predictions and potential harm in automated decision-making systems.	<i>Biases in training data and algorithmic design can lead to discriminatory outcomes and errors in machine learning models, raising concerns about fairness and accountability in AI applications.</i>	REFUTES	Harvard Business Review
<b>Indian Politics</b>	Verifiable	The Indian Supreme Court has the final authority on the interpretation of the Constitution.	<i>The Supreme Court of India has the final authority on the interpretation of the Constitution.</i>	SUPPORTS	Supreme Court of India
	Not Verifiable	The 2024 Indian general elections will have the highest	<i>Future events such as the voter turnout for the 2024 elections are speculative and not verifiable at present.</i>	NOT ENOUGH INFO	Election Commission of India

		voter turnout in history.			
	Verifiable	The capital of India is Mumbai.	<i>The capital of India is New Delhi, not Mumbai.</i>	REFUTES	Wikipedia
	Not Verifiable	The Indian Parliament is the largest parliamentary body in the world.	<i>The size of parliamentary bodies can vary and are difficult to compare, making this claim not verifiable.</i>	NOT ENOUGH INFO	Various Sources
<b>Indian Parliamentary</b>	Verifiable	Increasing the number of Lok Sabha seats will improve representation.	<i>Evidence suggests a correlation between increased seats and better representation.</i>	SUPPORTS	Election Commission of India
	Not Verifiable	Indian Parliament will become paperless in the next decade.	<i>The transition to a paperless Parliament within a specific timeframe is speculative.</i>	NOT ENOUGH INFO	Business Standard
	Verifiable	St. John's Wort is an effective treatment for severe depression.	<i>St. John's Wort is not recommended for severe depression due to inconsistent evidence of effectiveness.</i>	REFUTES	National Institutes of Health
	Not Verifiable	All laws passed by Indian Parliament are implemented effectively.	<i>Implementation of laws by Indian Parliament is hindered by various factors, leading to inefficiencies.</i>	REFUTES	The Hindu

Table 1: Snapshot of the Dataset

### 3.2.4 Dataset Statistics

The dataset prepared through prompt engineering techniques has over 1300 unique claims with their respective evidence, label and source across 5 diverse topics: Healthcare, Technology, Indian Politics, Finance and Indian Parliamentary.

## Claim Distribution by Label

The dataset for claim verification tasks includes a total of 1,320 claims, categorized into three labels: SUPPORTS, REFUTES, and NOT ENOUGH INFO. Table 2 represents the distribution of claim by label.

Label	No. of Claim/Evidence Pair
SUPPORTS	615
REFUTES	352
NOT ENOUGH INFO	353
TOTAL	1320

Table 2: Claim Distribution by Label

## Claim Distribution by Topic

The distribution of claims across different topics is shown in Table 3.

Topic	No. of Claims/Evidence Pair
Healthcare	400
Technology	320
Indian Politics	210
Finance	220
Indian Parliamentary	170
Total	1320

Table 3: Claim Distribution by Topic

## Claim Distribution by Topic and Label

The table below presents the distribution of claims across different topics, categorized by the labels SUPPORTS, REFUTES, and NOT ENOUGH INFO. This detailed breakdown shown in table 4 provides a comprehensive view of how claims are allocated within each topic.

Topic	SUPPORTS	REFUTES	NOT ENOUGH INFO
Healthcare	203	94	103
Technology	131	95	94
Indian Politics	100	50	60
Finance	101	56	63
Indian Parliamentary	80	57	33
Total	615	352	353

Table 4: Claim Distribution by Topic and Label

## Claim Distribution by Verifiability and Topic

The Table 5 provides a detailed distribution of claims based on their verifiability status across different topics. The claims are categorized into Verifiable, Not Verifiable, and Not Enough Info.

Topic	Verifiable	Not Verifiable	Not Enough Info
Healthcare	287	86	27
Technology	219	62	39
Indian Politics	159	57	14
Finance	147	62	12
Indian Parliamentary	111	59	0
Total	923	326	92

Table 5: Claim Distribution by Verifiability and Topic

Some Other Dataset Statistics like Highest, Lowest Length of Claim and Evidence is shown in Table 6 .

Feature	Length
Highest Claim Length(by Words)	34
Lowest Claim Length(by Words)	3
Average Claim Length(by Words)	10.98
Median Claim Length(by Words)	10
Highest Evidence Length(by Words)	126
Lowest Evidence Length(by Words)	3
Average Evidence Length(by Words)	25.04
Median Evidence Length(by Words)	17

Table 6: Dataset Features

### 3.3 Proposed Technique/Approach

This section of the report outlines the proposed technique and approach used in developing and evaluating models for claim verification. The approach leverages deep learning techniques such as BiLSTM and BERT, applied to a meticulously curated claim verification dataset.

Two state-of-the-art deep learning models were utilized for the claim verification task: BiLSTM (Bidirectional Long Short-Term Memory) and BERT (Bidirectional Encoder Representations from Transformers).

#### 3.3.1 BiLSTM(Bidirectional Long Short-Term Memory)

BiLSTM (Bidirectional Long Short-Term Memory) is a deep learning model specifically designed for sequence data analysis. It enhances the traditional LSTM (Long Short-Term Memory) by incorporating bidirectional information flow. Unlike standard LSTM, the input flows in both directions, and it's capable of utilizing information from both sides. This means that it considers both past and future context when processing each element in a sequence, making it highly effective for tasks like sentiment analysis in text data. BiLSTM is capable of capturing intricate dependencies within sequences, making it a valuable tool for claim verification task.

**Working Principle:** BiLSTM (Bidirectional Long Short-Term Memory) extends the capabilities of the traditional LSTM (Long Short-Term Memory) by introducing bidirectional processing. The fundamental principle behind BiLSTM lies in its ability to capture contextual information from both the past and future of a given word or element in a sequence.

Traditional LSTM processes sequences from left to right, considering only the past context. In contrast, BiLSTM introduces two separate layers of hidden states: one that reads the input sequence from left to right (forward LSTM) and another that reads it from right to left (backward LSTM). These two layers capture sequential dependencies in both directions, allowing the model to understand the context surrounding each element.

During training, BiLSTM learns to update its hidden states based on the input sequence. The final hidden states of both the forward and backward LSTMs contain comprehensive information about the sequence context. This bidirectional information flow is particularly valuable in tasks like sentiment analysis because it ensures that the model comprehensively understands the sentiment context of each word or element. In the context of claim verification, BiLSTM leverages its bidirectional

processing to capture nuances and dependencies in the claims and evidence, enhancing its ability to accurately classify the claims as supported, refuted, or having insufficient information. Additionally, it has been further utilized to determine whether a claim is verifiable or not, adding another layer of analysis to the verification process.

To illustrate, the unrolled BiLSTM is presented in the Fig1 below:

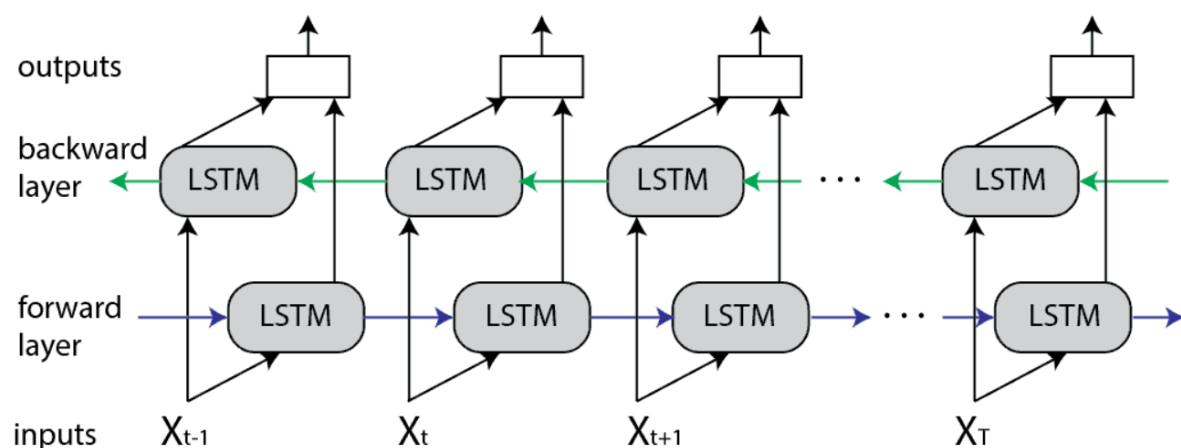


Fig1: BiLSTM Architecture<sup>3</sup>

### Advantages of BiLSTM for Claim Verification:

1. **Contextual Understanding:** BiLSTM captures context from both past and future elements in a sequence, enhancing its ability to understand the nuances and dependencies within claims and supporting evidence.
2. **Long-Range Dependencies:** It excels at capturing complex dependencies within sequences, which is critical for accurately determining the relationships between claims and their evidence.
3. **Flexible Length Handling:** BiLSTM handles variable-length sequences effectively, accommodating the varying lengths of claims and evidence without extensive preprocessing.
4. **High Performance:** Demonstrates state-of-the-art performance in claim verification and various other natural language processing tasks, ensuring reliable and accurate results.
5. **Versatility:** Suitable for a wide range of text analysis tasks, making it a versatile choice for verifying claims across different topics and domains.

<sup>3</sup> <https://www.baeldung.com/wp-content/uploads/sites/4/2022/01/bilstm-1.png>

### 3.3.2 BERT (Bidirectional Encoder Representations from Transformers)

BERT is a state-of-the-art natural language processing model introduced by Google in a recent paper published by researchers at Google AI Language. It's designed to understand the context of words in a sentence by considering both the left and right context, making it bidirectional. BERT achieves impressive performance on a wide range of NLP tasks due to its deep architecture and pretraining on massive text corpora. The architecture and effectiveness of BERT build upon the transformer model introduced in the paper "Attention Is All You Need" by Vaswani et al. (2017).

BERT's key technical innovation is applying the bidirectional training of Transformer, a popular attention model, to language modelling. This is in contrast to previous efforts which looked at a text sequence either from left to right or combined left-to-right and right-to-left training. The paper's results show that a language model which is bidirectionally trained can have a deeper sense of language context and flow than single-direction language models. In the paper, the researchers detail a novel technique named Masked LM (MLM) which allows bidirectional training in models in which it was previously impossible.

**Working Principle :** BERT makes use of Transformer, an attention mechanism that learns contextual relations between words (or sub-words) in a text. In its vanilla form, Transformer includes two separate mechanisms — an encoder that reads the text input and a decoder that produces a prediction for the task. Since BERT's goal is to generate a language model, only the encoder mechanism is necessary.

As opposed to directional models, which read the text input sequentially (left-to-right or right-to-left), the Transformer encoder reads the entire sequence of words at once. Therefore it is considered bidirectional, though it would be more accurate to say that it's non-directional. This characteristic allows the model to learn the context of a word based on all of its surroundings (left and right of the word).

The input is a sequence of tokens, which are first embedded into vectors and then processed in the neural network. The output is a sequence of vectors of size  $H$ , in which each vector corresponds to an input token with the same index. The architecture of the model is shown in Fig2.

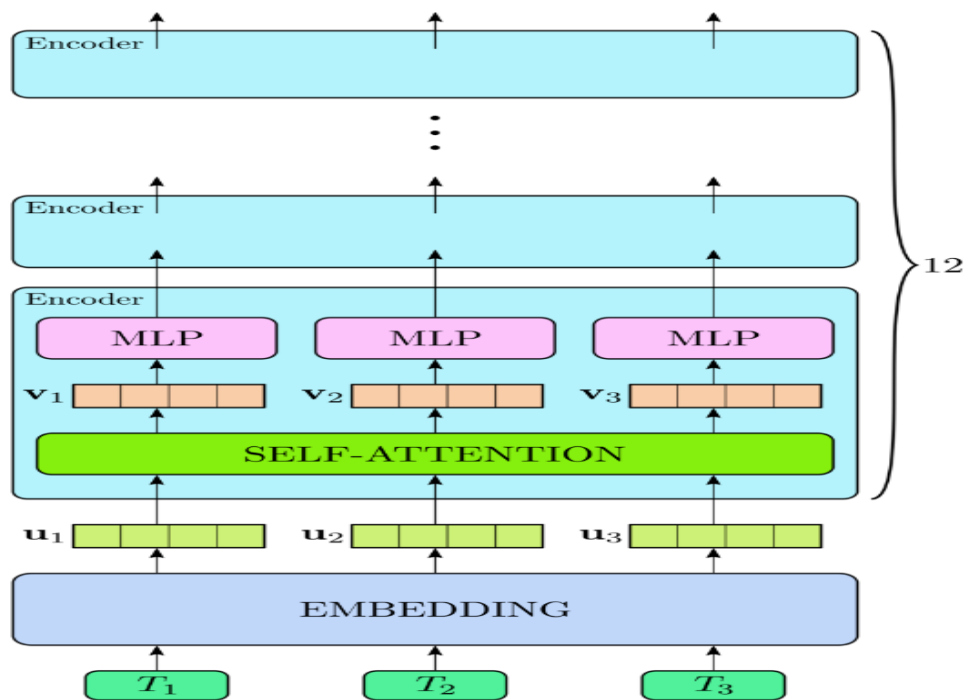


Fig2: BERT Architecture<sup>4</sup>

### Advantages of BERT:

- 1. Contextual Understanding:** BERT captures the contextual meaning of words by considering both their left and right context in a sentence. This contextualization allows it to understand sentiment nuances in movie reviews effectively.
- 2. Pretrained Knowledge:** BERT is pretrained on large text corpora, making it a powerful language model with extensive knowledge about language and semantics.
- 3. Transfer Learning:** BERT's pretrained weights can be fine-tuned for various NLP tasks with minimal task-specific data, reducing the need for extensive dataset collection and training time.
- 4. State-of-the-Art Performance:** BERT has set new benchmarks on a wide range of NLP tasks, demonstrating its ability to achieve state-of-the-art performance.
- 5. Open Source:** BERT is open-source and available to the research and developer community, fostering innovation in NLP applications.

<sup>4</sup> <https://www.researchgate.net/profile/Phillip-Schoenfelder/publication/355938108/figure/fig2/AS:1086781453807616@1636120298039/Simplified-high-level-view-of-the-BERT-architecture-7-8-The-word-tokens-are.png>

# CHAPTER 4: EXPERIMENTS AND RESULTS

---

## 4.1 Experimental Setup

In this project, we embark on three distinct experimental setups tailored to harness the capabilities of state-of-the-art natural language processing (NLP) models for different tasks.

Firstly, we delve into prompt engineering with GPT-3.5, focusing on secure API integration and efficient text completion generation.

Next, we explore a BiLSTM model for text classification, covering data preprocessing, model architecture, training, and evaluation.

Finally, we delve into utilizing the BERT model for text classification, showcasing its prowess in categorizing textual data into predefined classes. Each setup is meticulously designed to address specific objectives, highlighting the versatility and effectiveness of advanced NLP techniques in diverse applications.

### 4.1.1 Prompt Engineering

In this experimental setup, we aim to utilize the OpenAI GPT-3.5 model to generate text completions based on input prompts. The setup focuses on secure handling of environment variables, proper configuration of the OpenAI API, and the definition of a function to interact with the GPT-3.5 model.

The objective is to ensure a seamless and secure integration of the OpenAI API into our application, allowing for efficient and reliable generation of text completions. Below, we outline the key components and steps involved in preparing the environment, configuring the API, and defining the function for generating completions.

#### **Environment Preparation:**

To securely manage environment variables, we use a `.env` file. This approach ensures that sensitive information, such as API keys, is not hardcoded into the source code. The `dotenv` library is employed to read the `.env` file, and the `load_dotenv()` and `find_dotenv()` methods are utilized to locate and load these environment variables into the application.

- **Environment Variables:** Use a .env file to securely store and load environment variables.
  - Library: The dotenv library is used to read the .env file.
  - Method: load\_dotenv() and find\_dotenv() methods are used to locate and load the .env file.

### API Configuration:

The OpenAI API key, essential for accessing the GPT-3.5 model, is retrieved from the environment variables. By using `openai.api_key = "YOUR_API_KEY"`, we assign the API key directly from the environment variables, maintaining a secure and scalable configuration.

- **API Key:** The OpenAI API key is retrieved from the environment variables.
  - Direct Assignment: The API key is directly assigned using `openai.api_key = "YOUR_API_KEY"`.

### Function Definition:

- **Chat Completion Function:**
  - **Function Name:** `get_completion`
  - **Parameters:**
    - `prompt`: The input prompt for the GPT-3.5 model.
    - `model`: The model name, defaulting to "gpt-3.5-turbo".
  - **Messages Format:** The prompt is formatted as a message with the role "user".
  - **API Call:** `openai.ChatCompletion.create` is used to call the OpenAI API with the specified model and messages.
  - **Temperature Parameter:** Set to 0 to minimize randomness in the model's output, ensuring more deterministic responses.
  - **Return Value:** The function returns the content of the first message choice from the API response.

### Execution:

- **Output:** The function `get_completion` returns the model's output based on the provided prompt and specified parameters.

This setup ensures secure handling of API keys, proper configuration of the OpenAI API, and a structured function to interact with the GPT-3.5 model for generating completions based on input prompts.

### 4.1.2 BiLSTM Model

This environment setup outlines the process of building, training, and evaluating a deep learning model for text classification using TensorFlow and Keras. The setup encompasses various stages, including data loading and preprocessing, label encoding, text tokenization, model architecture design, compilation, training, and evaluation.

#### Environment Preparation:

To begin, we import essential libraries for data manipulation, model building, and evaluation, including NumPy and Pandas for data handling, Scikit-learn for preprocessing and evaluation, and TensorFlow with Keras for building and training the deep learning model.

- **Libraries:** Import necessary libraries for data manipulation, model building, and evaluation.
  - NumPy and Pandas: For data handling.
  - Scikit-learn: For data preprocessing, model evaluation, and metrics.
  - TensorFlow and Keras: For building and training the deep learning model.

#### Data Loading and Preprocessing:

- **Dataset:** Load the dataset using Pandas.
  - Combination: Combine claim and evidence columns into a single text column for input to the model.

#### Label Encoding:

- LabelEncoder: Encode categorical labels into numerical values.

#### Text Tokenization and Sequence Padding:

- **Tokenizer:** Initialize a tokenizer with a maximum of 20,000 words.
  - Fit: Fit the tokenizer on the combined text data.
  - Text to Sequences: Convert text data into sequences of integers.
  - Pad Sequences: Ensure all sequences have the same length (100).

#### Label Encoding (One-hot):

- One-hot Encoding: Convert numerical labels into a one-hot encoded format using Keras utility.

### **Data Splitting:**

- **Train-Test Split:** Split the data into training and testing sets using an 80-20 split.

### **Model Building:**

- **Architecture:**
  - **Embedding Layer:** Embedding layer with an input dimension of 20,000 and output dimension of 100.
  - **Bidirectional LSTM:** LSTM layer with 64 units, wrapped in a Bidirectional layer.
  - **Dense Layer:** Fully connected layer with 3 units (for 3 classes) and softmax activation function.

### **Model Compilation:**

- **Loss Function:** Categorical crossentropy for multi-class classification.
- **Optimizer:** Adam optimizer.
- **Metrics:** Accuracy.

### **Model Training:**

- **Training Parameters:**
  - **Epochs:** Train for 10 epochs.
  - **Batch Size:** Batch size of 32.
  - **Validation Split:** 20% of the training data used for validation.

### **Model Evaluation:**

- **Predictions:** Generate predictions on the test set.
- **Metrics Calculation:**
  - **Precision, Recall, F1 Score:** Calculate weighted precision, recall, and F1 score.
  - **Classification Report:** Generate a detailed classification report including precision, recall, and F1 score for each class.

### 4.1.3 BERT Model

In this experiment, we aimed to classify textual data into three categories: 'REFUTES', 'SUPPORTS', and 'NOT ENOUGH INFO'. We leveraged the BERT (Bidirectional Encoder Representations from Transformers) model to create a classifier for this task. The setup includes data preprocessing, model architecture, training, and evaluation. Below is a detailed description of the experimental setup, including all adjustments and technicalities.

#### Dataset Preparation:

- Data Source: The dataset used is the one prepared through prompt engineering techniques
- Text Processing: Claims and corresponding evidence are combined into a single text column.
- Label Encoding: The labels 'REFUTES', 'SUPPORTS', and 'NOT ENOUGH INFO' are encoded to integers 0, 1, and 2, respectively.

#### Data Splitting:

- The dataset is split into training (70%), validation (15%), and test (15%) sets using stratified sampling to maintain label proportions.

#### Tokenization and Encoding:

- Tokenizer: BERT's tokenizer (bert-base-uncased) is used to tokenize and encode the text sequences.
- Sequence Padding: All sequences are padded to a maximum length of 25 tokens.

#### Data Loading:

- Batch Size: A batch size of 16 is used.
- DataLoader: Training, validation, and test data are loaded into PyTorch DataLoaders for efficient batching.

#### Model Architecture:

- Base Model: The base model used is bert-base-uncased.
- Custom Layers:
  - A dropout layer with a dropout rate of 0.3.
  - A ReLU activation function.
  - Two fully connected layers: the first with 512 units and the second with 3 units to match the number of classes.

- A LogSoftmax activation function for the output layer.

### **Model Training:**

- Freezing Parameters: All BERT parameters are frozen initially, except for the last 4 layers which are unfrozen for fine-tuning.
- Optimizer: AdamW optimizer is used with a learning rate of  $1e-5$ .
- Class Weights: Class weights are computed to handle class imbalance and are applied to the loss function.
- Loss Function: Weighted Negative Log-Likelihood Loss (NLLLoss) is used.
- Learning Rate Scheduler: A linear learning rate scheduler with warmup is used to adjust the learning rate during training.

### **Training Loop:**

- Epochs: The model is trained for 4 epochs.
- Gradient Clipping: Gradients are clipped to a maximum norm of 1.0 to prevent exploding gradients.
- Validation: After each epoch, the model's performance on the validation set is evaluated.
- Model Checkpointing: The model with the lowest validation loss is saved during training.

### **Evaluation:**

- Best Model Loading: The best model (based on validation loss) is loaded for final evaluation.
- Test Set Performance: The model's predictions on the test set are evaluated using the classification report, providing precision, recall, and F1-score for each class.
- Confusion Matrix: The confusion matrix was computed to analyze the misclassifications

This experimental setup ensures that the model is properly trained, validated, and tested with techniques to handle data imbalance, optimize learning, and prevent overfitting.

## 4.2 Results

After performing experiments on claim verification dataset prepared through prompt engineering techniques with two state-of-the-art deep learning models like BiLSTM and BERT the results are as follows:

### 4.2.1 Results for Claim Classification

#### 1. BiLSTM Model

The BiLSTM model achieved an overall accuracy of 82% in the label classification task for claim verification. The values for other metrics like precision is 0.82, recall is 0.81 and f1-score is 0.82.

The macro average F1-score was 0.81, and the weighted average F1-score was 0.82, demonstrating consistent performance across different labels. Overall, the BiLSTM model shows robust performance in claim verification, particularly excelling in the "SUPPORTS" and "NOT ENOUGH INFO" categories. The results for BiLSTM model for the task of claim classification is shown in Table7.

	precision	recall	f1-score
SUPPORTS	0.87	0.86	0.87
REFUTES	0.68	0.79	0.73
NOT ENOUGH INFO	0.91	0.78	0.84
accuracy			0.82
macro avg	0.82	0.81	0.81
weighted avg	0.83	0.82	0.82

Table 7: Claim Classification results using BiLSTM

#### 2. BERT Model

The BERT model achieved an overall accuracy of 75% in the label classification task for claim verification. It performed exceptionally well in the "NOT ENOUGH INFO" category with a precision, recall, and F1-score of 0.88. The model also demonstrated high recall (0.90) in the "SUPPORTS" category. The macro and weighted averages of the F1-scores were both 0.73, indicating balanced performance across different labels. Overall, BERT shows strong potential for claim verification tasks, particularly in identifying claims with insufficient information.

The results for BERT model for the task of claim classification is shown in Table 8.

	<b>precision</b>	<b>recall</b>	<b>f1-score</b>
SUPPORTS	0.65	0.9	0.75
REFUTES	0.79	0.43	0.56
NOT ENOUGH INFO	0.88	0.87	0.88
accuracy			0.75
macro avg	0.78	0.73	0.73
weighted avg	0.77	0.75	0.73

Table 8 : Claim Classification results using BERT

## 4.2.2 Results for Claim Verifiability

### 1. BiLSTM Model

The BiLSTM model achieved an overall accuracy of 82% in verifying whether a given claim is Verifiable or Not Verifiable or Not Enough Info. It performed exceptionally well in the "verifiable" category with a precision of 0.88, recall of 0.91, and F1-score of 0.89. The results for BiLSTM model for the task of claim Verifiability is shown in Table 9.

	<b>precision</b>	<b>recall</b>	<b>f1-score</b>
verifiable	0.88	0.91	0.89
not verifiable	0.69	0.69	0.69
not enough info	0.25	0.12	0.17
accuracy			0.82
macro avg	0.60	0.57	0.58
weighted avg	0.80	0.82	0.81

Table 9: Claim Verifiability results using BiLSTM

### 2. BERT Model

The BERT model achieved an overall accuracy of 82% in verifying whether a given claim is Verifiable or Not Verifiable or Not Enough Info. The results for BERT model for the task of claim classification is shown in Table 10.

	precision	recall	f1-score
verifiable	0.88	0.9	0.89
not verifiable	0.79	0.69	0.74
not enough info	0.47	0.57	0.52
accuracy			0.82
macro avg	0.71	0.72	0.71
weighted avg	0.82	0.82	0.82

Table 10: Claim Verifiability results using BERT

### 4.2.3 Comparing the performance of BiLSTM and BERT

In comparing the performance of BiLSTM and BERT models across label classification and verifiability tasks, several key distinctions emerge. For label classification, BiLSTM demonstrates superior precision and recall, achieving a precision of 0.82 and a recall of 0.81 compared to BERT's precision of 0.78 and recall of 0.73. BiLSTM also maintains a higher F1-score at 0.82, whereas BERT lags behind at 0.73.

However, in the verifiability task, BERT shines with a precision of 0.82, surpassing BiLSTM's precision of 0.79. Moreover, BERT maintains a comparable recall of 0.82, slightly outperforming BiLSTM's recall of 0.81. While both models achieve similar F1-scores, with BiLSTM at 0.80 and BERT at 0.82, BERT's precision in verifiability is notably higher.

Overall, BERT performs better in the verifiability task, showing higher precision and comparable recall and F1-score compared to BiLSTM. However, for label classification, BiLSTM exhibits higher precision, recall, and F1-score. Table 11 shows a detailed comparison between the results obtained by BiLSTM and BERT for both the tasks.

	Label Classification		Verifiability	
	BiLSTM	BERT	BiLSTM	BERT
precision	0.82	0.78	0.79	0.82
recall	0.81	0.73	0.81	0.82
f1-score	0.82	0.73	0.80	0.82
accuracy	0.82	0.75	0.82	0.82

Table 11: Result Comparison of BiLSTM and BERT

#### 4.2.4 Results on Balanced Dataset

After balancing the dataset, we applied the BiLSTM model for the claim classification task due to its previously demonstrated superior performance. The results show a notable improvement in precision, recall, and F1-score for the class "REFUTES." Specifically, the precision for "REFUTES" increased to 0.79, recall to 0.77, and F1-score to 0.78.

For the class "SUPPORTS," the precision, recall, and F1-score remain stable at 0.83. Similarly, for the class "NOT ENOUGH INFO," precision remains at 0.89, recall at 0.93, and F1-score at 0.91. The overall accuracy of the classification also remains steady at 83%.

In summary, there is a balanced improvement across precision, recall, and F1-score metrics, particularly notable for the "REFUTES" class, indicating the efficacy of the BiLSTM model in classifying claims, even after dataset balancing. Table 12 shows the detailed result of the same.

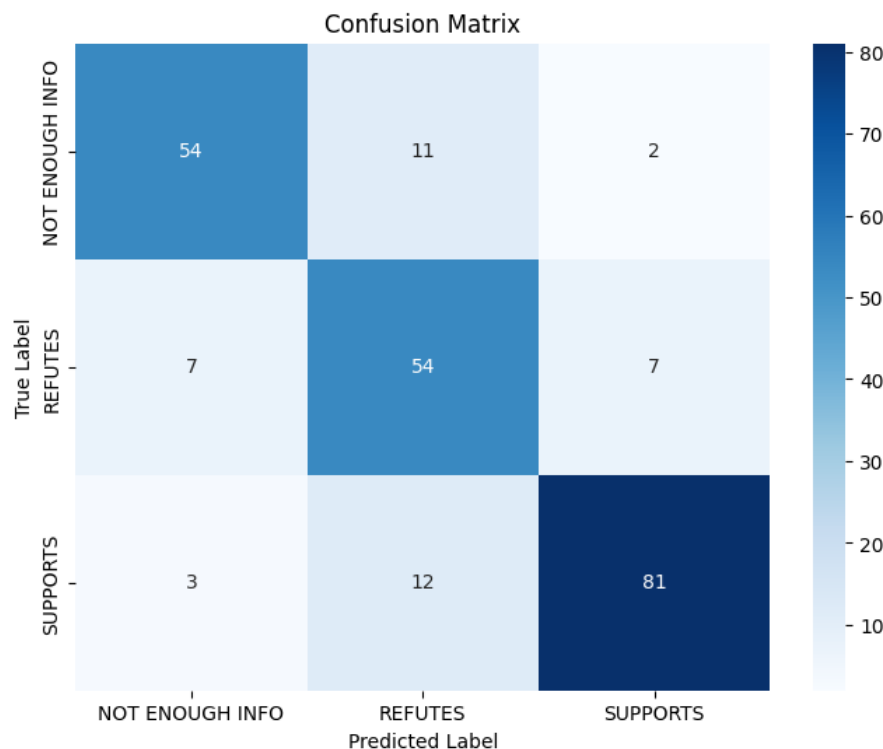
	<b>precision</b>	<b>recall</b>	<b>f1-score</b>
SUPPORTS	0.83	0.83	0.83
REFUTES	0.79	0.77	0.78
NOT ENOUGH INFO	0.89	0.93	0.91
accuracy			0.83
macro avg	0.84	0.84	0.84
weighted avg	0.83	0.83	0.83

Table 12: Claim Classification results using BiLSTM on balanced dataset

## 4.3 Error Analysis

The confusion matrix provides a comprehensive overview of the performance of a classification model by summarizing the counts of true positive (TP), true negative (TN), false positive (FP), and false negative (FN) predictions across all classes.

### Confusion Matrix for Claim Classification Using BiLSTM Model



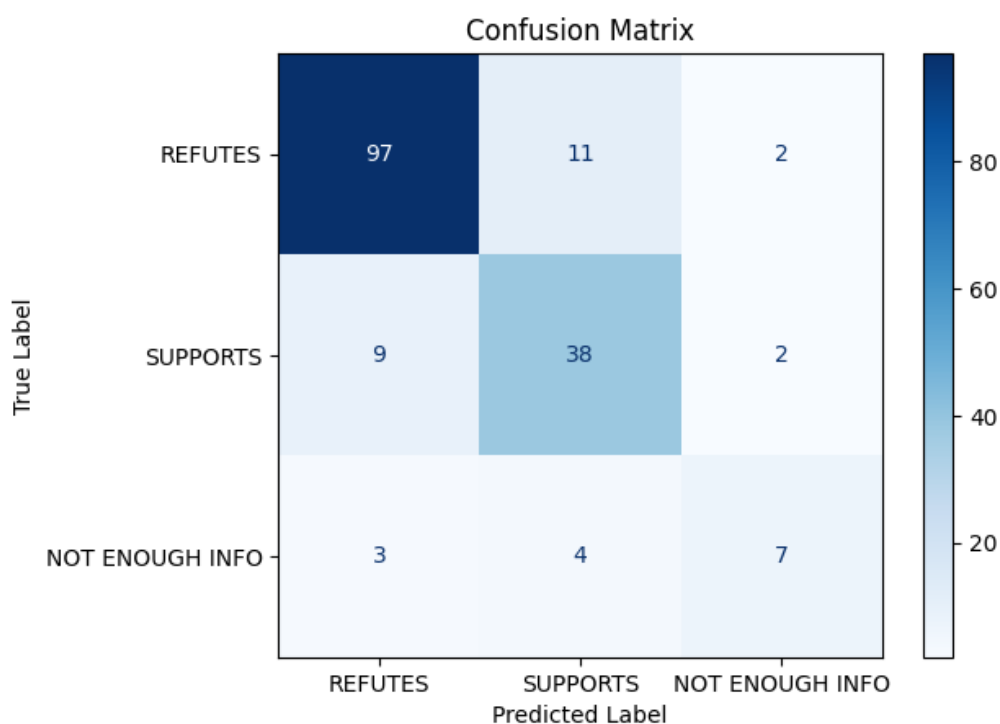
- **Class 0 (NOT ENOUGH INFO):** 54 instances were correctly classified, 7 instances were misclassified as class 1 (REFUTES) and 6 instances were misclassified as class 2 (SUPPORTS).
- **Class 1 (REFUTES):** 45 instances were correctly classified, 7 instances were misclassified as class 0 (NOT ENOUGH INFO) and 16 instances were misclassified as class 2 (SUPPORTS).
- **Class 2 (SUPPORTS):** 86 instances were correctly classified, 3 instances were misclassified as class 0 (NOT ENOUGH INFO) and 7 instances were misclassified as class 1 (REFUTES).

Example of Claims that were misclassified are shown in table 13

Claim	True Label	Predicted Label
<i>Investing in real estate investment trusts (REITs) provides diversification and income.</i>	REFUTES	SUPPORTS
<i>Treasury bonds are considered one of the safest investments.</i>	REFUTES	NOT ENOUGH INFO
<i>Bitcoin will replace traditional currencies by 2030</i>	NOT ENOUGH INFO	REFUTES

Table 13: Example of misclassified claims by BiLSTM

### Confusion Matrix for Claim Classification Using BERT Model



Interpreting the matrix, we can see that:

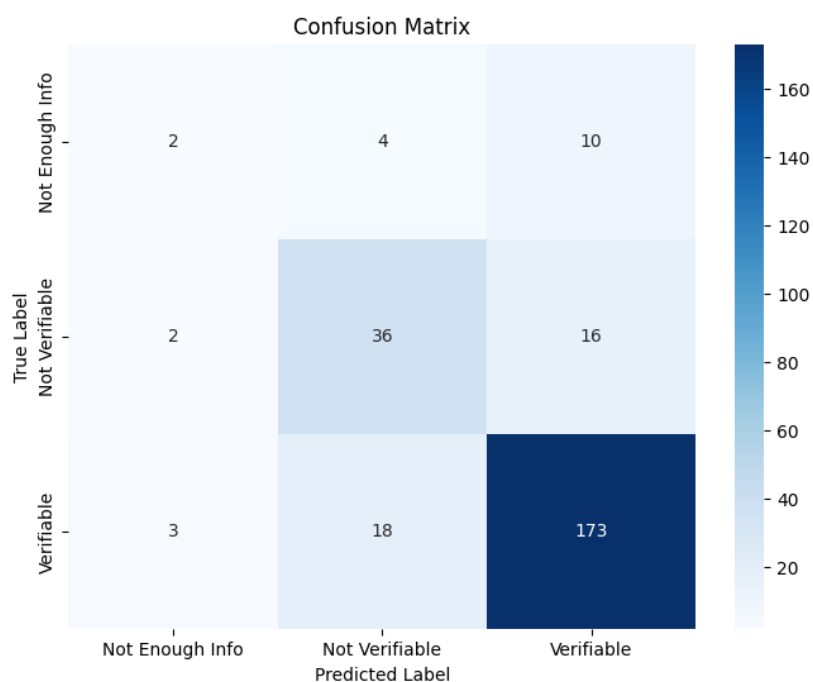
- **Class 0 (REFUTES):** 42 instances were correctly classified, 10 were misclassified as class 1, and 1 was misclassified as class 2.
- **Class 1 (SUPPORTS):** 43 instances were correctly classified, 16 were misclassified as class 0, and 8 were misclassified as class 2.
- **Class 2 (NOT ENOUGH INFO):** 45 instances were correctly classified, 2 were misclassified as class 0, and 6 were misclassified as class 1.

Example of Claims that were misclassified by BERT are shown in Table 14.

Claim	True Label	Predicted Label
<i>Hormone replacement therapy can alleviate menopausal symptoms</i>	SUPPORTS	NOT ENOUGH INFO
<i>The deployment of autonomous drones for delivery services will lead to job displacement and economic disruptions in the logistics industry.</i>	NOT ENOUGH INFO	REFUTES
<i>Index funds outperform actively managed funds</i>	NOT ENOUGH INFO	REFUTES

Table 14: Example of misclassified claims by BERT

### Confusion Matrix for Claim Verifiability Using BiLSTM Model



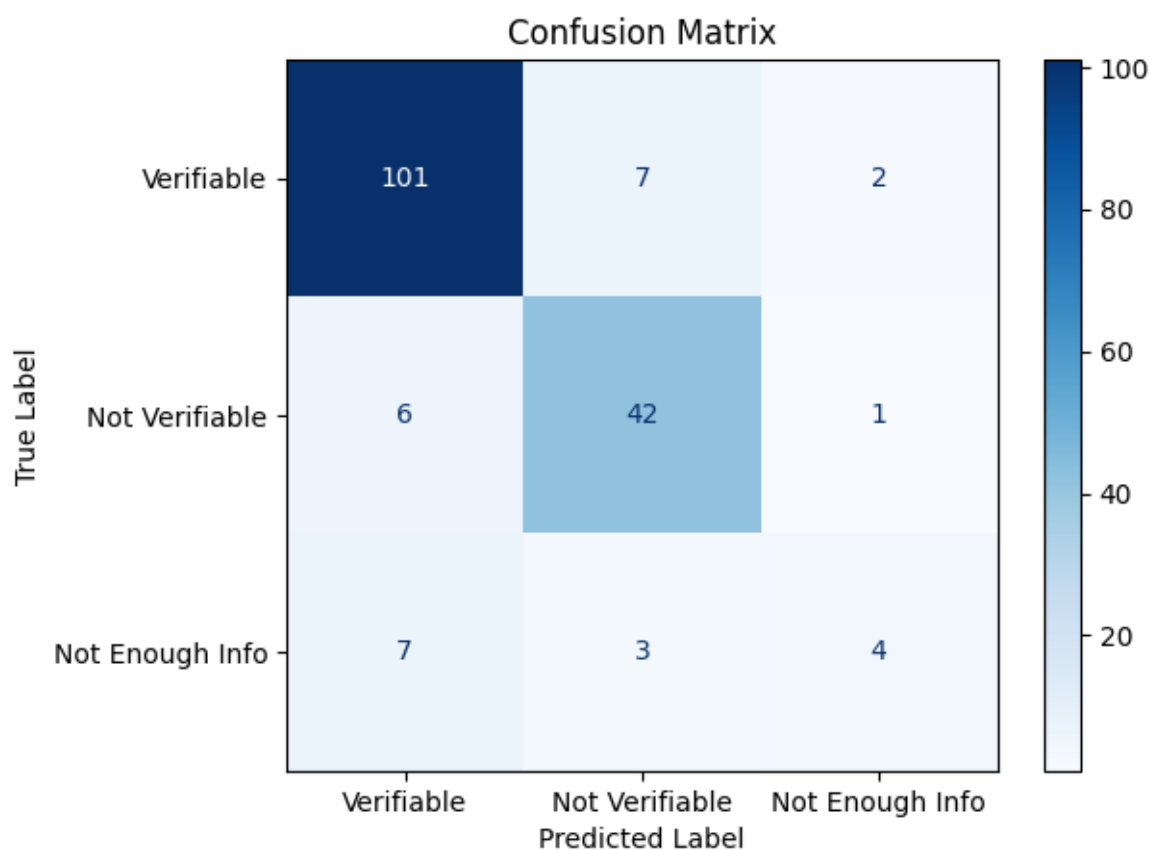
- **Class 0 (Not Enough Info):** 2 instances were correctly classified, 4 instances were misclassified as class 1 (Not Verifiable) and 10 instances were misclassified as class 2 (Verifiable).
- **Class 1 (Not Verifiable):** 36 instances were correctly classified, 2 instances were misclassified as class 0 (Not Enough Info) and 16 instances were misclassified as class 2 (Verifiable).
- **Class 2 (Verifiable):** 173 instances were correctly classified, 3 instances were misclassified as class 0 (Not Enough Info) and 18 instances were misclassified as class 1 (Not Verifiable).

Table 15 shows some of the examples of the claims that were misclassified by BiLSTM.

Claim	True Label	Predicted Label
<i>All developed countries will adopt a universal digital currency within the next 15 years.</i>	Verifiable	Not Verifiable
<i>The next global financial crisis will be caused by a major cryptocurrency collapse.</i>	Verifiable	Not Verifiable
<i>Mutual funds offer investors access to a diversified portfolio of securities managed by professional fund managers.</i>	Verifiable	Not Enough Info

Table 15: Example of misclassified claim verifiability by BiLSTM

### Confusion Matrix for Claim Verifiability Using BERT Model



Here's the interpretation of the confusion matrix provided:

- **Verifiable** (Class 0): 101 instances were correctly classified as Verifiable (True Positives), 7 instances of Verifiable claims were incorrectly classified as Not

Verifiable (False Negatives) and 2 instances of Verifiable claims were incorrectly classified as Not Enough Info (False Negatives).

- **Not Verifiable** (Class 1): 42 instances were correctly classified as Not Verifiable (True Positives), 6 instances of Not Verifiable claims were incorrectly classified as Verifiable (False Negatives) and 1 instances of Not Verifiable claims were incorrectly classified as Not Enough Info (False Negatives).
- **Not Enough Info** (Class 2): 4 instance was correctly classified as Not Enough Info (True Positive), 7 instance of Not Enough Info was incorrectly classified as Verifiable (False Positives) and 3 instance of Not Enough Info was incorrectly classified as Not Verifiable (False Positives).

Table 16 shows some of the examples of the claims that were misclassified by BERT.

Claim	True Label	Predicted Label
<i>The deployment of AI-powered chatbots in mental health services improves accessibility and affordability of therapy, providing timely support and resources to individuals in need.</i>	Not Verifiable	Not Enough Info
<i>India's literacy rate is higher than that of Pakistan.</i>	Not Enough Info	Verifiable

Table 16: Example of misclassified claim verifiability by BERT

## 4.4 Discussion

Overall, the results indicate that while the BiLSTM model is more effective for the label classification task, the BERT model excels in the verifiability task. Each model has its strengths: BiLSTM performs better in tasks requiring understanding of sequential text patterns, whereas BERT's strong contextual understanding makes it superior in verifying claims. This highlights the importance of selecting the appropriate model based on the specific requirements of the task at hand. Future work could explore hybrid approaches that leverage the strengths of both models to achieve even better performance across diverse tasks.

## CHAPTER 5: CONCLUSION AND FUTURE WORK

---

In this report, we demonstrated the effectiveness of integrating prompt engineering with deep learning models to enhance claim verification accuracy. Prompt engineering facilitated the creation of a structured dataset that significantly improved the performance of deep learning models in classifying claims and identifying relevant text spans. The results highlighted the advantages of this approach over traditional methods using raw data.

Future work will focus on expanding the dataset by increasing the diversity and size of the dataset by incorporating a wider range of sources and claim types can enhance the robustness and generalizability of the models, exploring additional deep learning architectures, investigating alternative methodologies, and developing techniques to establish relationships between claim sources and evidence. Developing more sophisticated and context-aware prompts can further improve the quality of the dataset. Experimenting with different prompt structures and exploring automatic prompt generation techniques could yield even better annotation results. These efforts aim to further improve the robustness and efficacy of claim verification systems, addressing the growing challenge of misinformation.

## References

Chen, Banghao, et al. Unleashing the Potential of Prompt Engineering in Large Language Models: A Comprehensive Review. arXiv:2310.14735, arXiv, 27 Oct. 2023

Devlin Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. [BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding](#). In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Thorne James, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018. [FEVER: a Large-scale Dataset for Fact Extraction and VERification](#). In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), pages 809–819, New Orleans, Louisiana. Association for Computational Linguistics.

Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. "Attention Is All You Need." ArXiv:1706.03762 [Cs], June 12. <https://arxiv.org/abs/1706.03762>.

Wadden David, Shanchuan Lin, Kyle Lo, Lucy Lu Wang, Madeleine van Zuylen, Arman Cohan, and Hannaneh Hajishirzi. 2020. [Fact or Fiction: Verifying Scientific Claims](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7534–7550, Online. Association for Computational Linguistics.