

# Detecting Logical Fallacies with LLMs

**Adrian Imfeld**

Lucerne University  
of Applied Sciences  
and Arts (HSLU)

Master of Science in Applied  
Information and Data Science  
Autumn Semester 2024

Degree dissertation submitted as part of the  
requirements for the MSc in Applied Information  
and Data Science at the School of Business,  
Lucerne University of Applied Sciences and Arts

December, 2024

# Detecting Logical Fallacies with LLMs

## Master's Thesis

Lucerne University  
of Applied Sciences  
and Arts (HSLU)

Master of Science in Applied  
Information and Data Science  
Autumn Semester 2024

Author: Adrian Imfeld  
E-Mail: aimfeld@aimfeld.ch

Supervisor: Marcel Blattner, PhD  
E-Mail: marcel.blattner@hslu.ch

December, 2024

# Abstract

This thesis explores the application of large language models (LLMs) for detecting logical fallacies in natural language text. Logical fallacies (errors in reasoning) are pervasive in online discourse and social media, undermining critical thinking and informed decision-making. Improving on previous benchmark results, this research demonstrates that state-of-the-art LLMs can effectively identify and classify both formal and informal logical fallacies, and therefore help address the critical challenge of combating misinformation and manipulative communication. A significant outcome is the development of the Fallacy Search web application, an interactive tool for detecting fallacious reasoning in long textual input, showcasing the practical utility of these methods.

Central to this study is the FALLACIES dataset, which served as the primary benchmark for evaluating LLM performance. This dataset comprises valid and invalid arguments across 232 fallacy types, categorized into formal and informal fallacy subcategories. The current flagship models of OpenAI, Anthropic, and Google performed very well on the binary fallacy identification task of distinguishing between valid and invalid reasoning, with accuracies in a close range between 83% and 88%. Meta's open source models are catching up, while Mistral AI models are trailing behind. Notably, many of the tested models exhibited a systematic false alarm bias, judging valid reasoning as invalid. Fine-tuning experiments using this dataset revealed substantial performance improvements, although these gains may be attributed to overfitting to task-specific patterns. Furthermore, this study used the MAFALDA dataset, a unified benchmark consolidating multiple sources of argumentative content, annotated with 23 distinct fallacy types. The MAFALDA disjunctive annotation scheme allows multiple fallacy labels per text span, reflecting the complexity of real-world arguments. Using a scalable single-prompt method with structured output to analyze long textual input, GPT-4o achieved significant advancements over previously reported performance scores.

This thesis contributes to the intersection of artificial intelligence and critical thinking by advancing automated methods for logical fallacy detection. It underscores the importance and feasibility of robust AI tools in mitigating misinformation and fostering informed public discourse.

*Keywords:* large language model, LLM, logical fallacy, fallacy detection, critical thinking

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Problem Statement and Motivation . . . . .	1
1.2	Objectives . . . . .	3
<b>2</b>	<b>Research Questions</b>	<b>4</b>
2.1	Fallacy Identification . . . . .	4
2.2	Fallacy Classification . . . . .	5
2.3	Fallacy Search . . . . .	5
<b>3</b>	<b>Theoretical Background</b>	<b>7</b>
3.1	Arguments . . . . .	7
3.2	Logical Fallacies . . . . .	9
3.3	Large Language Models (LLMs) . . . . .	10
<b>4</b>	<b>Literature Review</b>	<b>14</b>
4.1	Logical Fallacy Datasets . . . . .	14
4.2	Logical Reasoning Capabilities of LLMs . . . . .	16
<b>5</b>	<b>Datasets</b>	<b>18</b>
5.1	FALLACIES Dataset . . . . .	18
5.2	MAFALDA Dataset . . . . .	19
<b>6</b>	<b>Methods</b>	<b>22</b>
6.1	Model Selection . . . . .	22
6.2	Fallacy Identification . . . . .	24
6.3	Fallacy Classification . . . . .	28
6.4	Fallacy Search . . . . .	29
6.5	Technical Implementation . . . . .	31
6.6	Performance Metrics . . . . .	31
6.7	Sanity Checks . . . . .	34
<b>7</b>	<b>Results</b>	<b>35</b>
7.1	Fallacy Identification . . . . .	35
7.2	Fallacy Classification . . . . .	45
7.3	Fine-Tuning Performance . . . . .	47
7.4	Fallacy Type Performance . . . . .	49
7.5	Fallacy Search . . . . .	51

<b>8 Practical Implementation</b>	<b>53</b>
8.1 From Research to Application . . . . .	53
8.2 Fallacy Search Web Application User Interface . . . . .	53
8.3 Use Cases . . . . .	55
8.4 Real-World Implications . . . . .	58
<b>9 Discussion</b>	<b>59</b>
9.1 Context and Significance . . . . .	59
9.2 Summary of Key Findings . . . . .	59
9.3 Interpretation of Results . . . . .	61
9.4 Limitations of the Study . . . . .	63
9.5 Future Research Directions . . . . .	64
9.6 Conclusion . . . . .	65
<b>References</b>	<b>67</b>
<b>Glossary</b>	<b>73</b>
<b>Acronyms</b>	<b>77</b>
<b>List of Tables</b>	<b>79</b>
<b>List of Figures</b>	<b>80</b>
<b>List of Data Sources</b>	<b>82</b>
<b>Appendices</b>	<b>83</b>
Appendix A: Rephrased Prompt (Experiment 1.5) . . . . .	84
Appendix B: Reliability Test (Experiment 1.6) . . . . .	85
Appendix C: Out-of-Distribution (OOD) Evaluation of Fine-Tuned Models . . . . .	86
Appendix D: Fallacy Search F1-Scores with Uncovered Text Spans . . . . .	87
<b>Declarations</b>	<b>88</b>
Declaration of Originality . . . . .	88
Declaration of the use of Generative AI . . . . .	89

# 1 Introduction

## 1.1 Problem Statement and Motivation

In today's digital age, pseudoscience, unsound reasoning, and highly polarized political statements abound on social media. With the advent of generative AI created text, the problem of misinformation, propaganda, and other manipulative content is only becoming bigger. This surge in AI-generated content could further blur the lines between reality and fabrication, as these tools become ever more sophisticated and accessible. In a recent randomized controlled trial, Salvi et al. (2024) found that when debating randomly assigned topics, humans were more likely to be persuaded by GPT-4 than human opponents, especially when basic personal information was provided about the debaters. The authors warn that by crafting micro-targeted messages based on personal information disclosed on social media, [large language models \(LLMs\)](#) could therefore be used for mass persuasion at scale. As Sourati, Ilievski, et al. (2023) point out, misinformation and propaganda are thorny issues for social media platforms and are under the scrutiny of different organizations and governmental bodies, including the United Nations and the European Commission, as the spread of harmful and incorrect arguments can influence the population and lead to political shifts and civil unrest.

Critical thinking plays a crucial role in refuting false information, manipulative content, and bad arguments on social media. It can be defined as the analysis of available facts, evidence, observations, and arguments to form a judgment by applying rational, skeptical, and unbiased analyses and evaluation ("Critical thinking", 2024). A central aspect of critical thinking is an understanding of logical fallacies which are systematic errors in reasoning that distort our thinking. Logical fallacies can be intentionally used to manipulate or persuade by deception, or happen unintentionally because of human limitations such as carelessness, cognitive or social biases, and ignorance, or possibly due to the limitations of language and the understanding of language ("Fallacy", 2024). Unfortunately, the general public is often poorly equipped with critical thinking skills, including the detection of fallacies, as these skills are not taught systematically in school (Bouygues, 2022). Even at university level, instruction on critical thinking and logical fallacies often does not extend further than stating the principle that "correlation does

## 1 Introduction

not imply causation”, except in specialized courses like introductory logic in philosophy or the study of cognitive biases in psychology.

Although LLMs can be used to generate manipulative content, they also represent a promising avenue as a tool to identify faulty reasoning. These models have shown remarkable capabilities in understanding text, performing at the level of humans or even surpassing them in challenging tasks, including theory-of-mind abilities, understanding false beliefs and misdirection, indirect requests, and recognizing irony (Strachan et al., 2024). However, the task of detecting logical fallacies is not trivial, as it involves not only identifying the fallacious content but also understanding the context in which it appears. This requires deep semantic analysis, a challenge that brings together the fields of artificial intelligence, linguistics, logic, and philosophy.

Early LLMs such as GPT-2 and GPT-3 performed poorly in the task of logical fallacy detection, barely exceeding random guessing (Jin et al., 2022). In more recent work, GPT-4 has shown promising results, achieving an overall average accuracy of 87.7%, surpassing other models by a substantial margin (Hong et al., 2024). However, these results were achieved on a binary fallacy identification task, given an argument and a Yes/No choice to indicate whether the argument is valid or fallacious. The accuracy for classifying a statement into 232 given fallacy categories was much lower, with GPT-4 achieving an overall accuracy of 35.0% and most models performing at less than 10%. This means that there is substantial room for improvement for newer models before reaching ceiling effects. The authors also found that the fallacy identification performance of most LLMs was much lower for formal fallacies, which are errors in the logical structure of an argument, than for informal fallacies, which involve errors in reasoning due to content or context. In a related but smaller study, Lim and Perrault (2024) found that GPT-4 performed well overall at accuracies of 79% and 90%, but showed specific weaknesses in identifying instances of the [Appeal to Emotion](#) fallacy, failing to do so nearly 40% of the time.

Although recent results based on labeled benchmark datasets are encouraging, it is unclear whether these LLM capabilities will fall short when applied to longer text, where sparse occurrences of logical fallacies need to be detected and classified. As Hong et al. (2024) emphasized, for a long argument comprising multiple reasoning steps, overall performance could decrease exponentially with the number of steps in it. Adding to the complexity, more than 300 types of logical fallacies have been distinguished in the literature (Bennet, 2021). There are many synonymous categories, and classification is not always straightforward in practice.

However, the goal of automatic detection and classification of the most important and

prevalent fallacy types has become realistic with recent progress in LLM capabilities, although certain weaknesses are to be expected. Tools leveraging these models could prove to be very useful in counteracting the misinformation issues described above. Specifically, they could potentially be valuable assets for quality control in journalism, highlighting manipulative content and propaganda, and analyzing blog posts and content on social media. In addition, they may be applied to identify weaknesses in legal arguments and court decisions, flagging pseudoscience and unsound marketing, and many other applications.

## 1.2 Objectives

In short, this thesis has three main objectives:

1. **Fallacy Identification:** Find out how well the current flagship LLMs can identify fallacies, that is, *distinguish between valid arguments and logical fallacies*. To this end, different methods of prompting will be used and compared, such as zero-shot, few-shot, and chain-of-thought.
2. **Fallacy Classification:** Evaluate these LLMs on the more challenging task of *classifying logical fallacies* into a large number of categories. At this stage, specific strengths and weaknesses in detecting different categories of fallacies will become apparent.
3. **Fallacy Search**
  - a) **Research:** Develop and evaluate a method for *detecting and analyzing logical fallacies in real-world text*. The method should have a quantitative performance metric, be reliable and cost-effective, handle long input text, use a single prompt with structured output, and minimize false alarms.
  - b) **Practical Implementation:** Leverage this method to *build an LLM-powered tool for detecting and analyzing logical fallacies in real-world text*. The practical implementation of this tool is detailed in Chapter 8.

The research questions, methodology, and benchmark datasets for addressing these three objectives will be described in more detail in the following chapters.

## 2 Research Questions

This chapter defines the research questions associated with the three main objectives listed in Section 1.2.

### 2.1 Fallacy Identification

The first group of questions relates to the first objective of fallacy identification, which means finding out how well current flagship LLMs can distinguish between valid arguments and logical fallacies. For this binary fallacy identification task, the FALLACIES dataset will be used, which includes both valid and invalid arguments (Hong et al., 2024). As described in detail in Section 5.1, the FALLACIES dataset is a substantial improvement over the LOGIC dataset used in earlier studies (Jin et al., 2022; Lim & Perrault, 2024; Sourati, Ilievski, et al., 2023).

Key research questions:

1. **Model Performance:** Can the current flagship LLMs surpass the GPT-4 fallacy identification accuracy of 87.7% reported by Hong et al. (2024)? Will the models perform worse at identifying formal fallacies, compared to informal ones, as reported by the authors? Which models exhibit superior performance and which lag behind? Different models will be compared, including flagship commercial and open-source LLMs from OpenAI, Anthropic, Google, Meta, and Mistral AI.
2. **Model Size:** How does the number of parameters impact performance? Small models have performed poorly in previous studies (see e.g. Helwe et al., 2024; Jin et al., 2022), but more recent efficient models released by the mentioned providers are more powerful and could be more promising. To answer this question, LLMs will be divided into two groups, large and small models, and group performance will be compared.
3. **Prompt Method Comparison:** What are the most effective prompt methods for fallacy identification using LLMs? The research will explore various approaches,

## 2 Research Questions

such as zero-shot, few-shot, and chain-of-thought, to understand which techniques best leverage the models' capabilities for this specific task.

4. **Fine-Tuning Performance:** To what extent can targeted fine-tuning enhance fallacy identification? The research will assess both absolute performance gains and generalization to broader logical reasoning tasks to evaluate potential overfitting effects.

### 2.2 Fallacy Classification

The second group of questions relates to the second objective of evaluating the chosen LLMs on the more challenging task of classifying logical fallacies into a large number of specific fallacy types. Again, the FALLACIES dataset will be used for this task.

Key research questions:

1. **Model Performance:** Can the current flagship LLMs surpass the GPT-4 fallacy classification accuracy of 35.0% reported by Hong et al. (2024)? Note that there is much more room for improvement in this challenging task, since models other than GPT-4, GPT-3.5, and Flan-T5-xxl have scored below 10%.
2. **Fine-Tuning Performance:** Can classification accuracy be improved by fine-tuning a model on a part of the dataset? Fine-tuning has been used in a previous study (Sourati, Ilievski, et al., 2023), using the LOGIC dataset which has far fewer categories than the FALLACIES dataset.
3. **Fallacy Type Performance:** Which types of fallacies are most effectively classified by LLMs, and which types pose challenges? The study will assess the strengths and weaknesses of LLMs in identifying and classifying different fallacies, from more straightforward formal fallacies to nuanced informal fallacies that depend heavily on context and subtlety. Will the models exhibit worse classification performance for formal logical fallacies compared to informal ones?

### 2.3 Fallacy Search

The third group of questions is concerned with the third objective, that is, applying these models to detect and analyze logical fallacies in real-world text. This investigation will utilize the Multi-level Annotated Fallacy Dataset (MAFALDA) created by Helwe et al.

## 2 Research Questions

(2024), which aggregates textual content from diverse real-world sources, including U.S. presidential debates, social media interactions, and climate change discussions. The MAFALDA dataset is particularly valuable for this analysis because it covers naturalistic argumentative contexts rather than using artificially constructed examples. A detailed description of this dataset and its characteristics will be presented in Section 5.2.

Key research questions:

1. **Structured Output:** Can LLMs be effectively prompted to produce structured responses that not only detect several fallacies at once, but also classify their type and provide reasoning for the classification? This involves testing the model's ability to dissect arguments and generate coherent, contextually aware feedback.
2. **Reasoning Problems:** When explaining why a certain argument contains a particular logical fallacy, are there any patterns of reasoning problems exhibited by the LLMs? For example, do the models take into account the difference between appealing to the authority of an individual scientist, a small group of scientists, or a solid consensus of the scientific community in a particular field? Are the models prone to committing so-called fallacy fallacies, i.e., the invalid assumption that the conclusion of a fallacy must necessarily be false (Aberdein, 2023)?
3. **False Alarms:** As Maarten Boudry (2017) has pointed out, philosophy students instructed to look for logical fallacies in real life tend to become overeager in fallacy detection, leading to false positives. Do LLMs have the same tendency? What mechanisms can be implemented to reduce false positives in fallacy detection? This involves refining the sensitivity of the models to differentiate between actual fallacious reasoning and closely related but logically sound arguments by careful prompt engineering.
4. **Principle of Charity:** How can LLMs be instructed to critically analyze arguments while adhering to the principle of charity, meaning that they do not unfairly construct and argue against misrepresentations of the original arguments?

## 3 Theoretical Background

This chapter introduces the foundational concepts of arguments and logical fallacies, followed by a brief introduction to large language models (LLMs). The thesis integrates these topics, bridging the disciplines of philosophy, logic, and artificial intelligence.

### 3.1 Arguments

An argument is a complex symbolic structure in which premises support a conclusion, marked by indicators such as "hence" or "therefore". The study of arguments is closely tied to reasoning, which involves reaching conclusions through careful consideration of information. According to Dutilh Novaes (2022), the following main categories of arguments can be distinguished:

1. **Deductive Arguments:** These arguments are characterized by the necessity that if the premises are true, the conclusion must also be true. *Valid* deductive arguments are those where the truth of the premises necessitates the truth of the conclusion: the conclusion must be true *if* the premises are true. A *valid* deductive argument whose premises are *actually* true is said to be *sound*. Validity depends solely on the argument's form, not the actual truth of its premises. For example, the argument "All animals are birds; my pet is an animal; therefore, my pet is a bird" is valid because the conclusion logically follows from the premises, even though the premises are factually false. In contrast, an argument is *sound* when it is valid and its premises are actually true. This guarantees that the conclusion is also true. For instance, "In some countries, foreigners are not allowed to vote; in those countries, some athletes are foreigners; therefore, in some countries, some athletes are not allowed to vote" is both valid and sound because its premises and conclusion align with reality. Understanding this distinction helps distinguish between flawed reasoning and robust arguments when analyzing logical fallacies. Deductive arguments are often associated with mathematical proof and are known for their certainty and indubitability.

### *3 Theoretical Background*

2. **Inductive Arguments:** Inductive arguments involve drawing conclusions about future instances and general principles based on observations of past instances and regularities, such as predicting the sun will rise in the east because it always has. These arguments rely on statistical frequencies to generalize from observed cases to unobserved ones, with the premises providing some support for the conclusion, though never fully guaranteeing it. Historically, induction was recognized by Aristotle as moving from particulars to universals, but it gained prominence during the scientific revolution with figures like Francis Bacon advocating for empirical observation. Despite being widely used and generally reliable, inductive reasoning faces challenges, notably David Hume's "problem of induction," which questions the justification of inductive inferences without circular reasoning. Induction is central to scientific practice and everyday reasoning.
3. **Abductive Arguments:** Abduction involves forming explanatory hypotheses from observed facts, often described as inferring to the best explanation. While not necessarily ensuring that the conclusion follows with certainty, abduction is crucial in scientific discovery and everyday reasoning. It is distinct from induction as it seeks to explain rather than predict. A prosecutor's closing argument is a prime example of abduction, where they review the evidence and argue that the most plausible explanation is that the defendant committed the crime.
4. **Analogies:** These arguments rely on similarities between two entities to infer that what is true for one is likely true for the other. They are prevalent in various fields, including law and philosophy, but are often critiqued based on the strength of the similarity between compared entities. A simple example of an analogical argument is comparing the heart to a pump, suggesting that understanding how a pump works can provide insights into how the heart functions. The strength of this analogy depends on how closely the mechanics of pumping water and circulating blood are comparable.
5. **Fallacies:** Fallacious arguments, including logical fallacies, are those that appear to be valid but are not. They are crucial to identify and avoid in order to argue effectively. Philosophical debates about fallacies explore what makes an argument fallacious, their causes, and whether focusing on fallacies is a productive approach to studying argumentation. Despite criticisms, fallacies remain central to understanding and teaching argumentative skills.

### 3.2 Logical Fallacies

The concept of fallacy was introduced into the Western intellectual tradition by Aristotle's study of fallacious arguments in *On Sophistical Refutations*, in which he identified thirteen types of fallacies (Smith, 2000). Fallacies have since been studied by many well-known philosophers, including Francis Bacon, John Locke, Jeremy Bentham, John Stuart Mill, and others (Hansen, 2023). In more recent times, logical fallacies were discussed and incorporated in the framework of scientific skepticism by the American astronomer, planetary scientist, and science communicator Carl Sagan (Sagan, 1995).

In modern fallacy studies, it is common to distinguish between *formal* and *informal* fallacies (Hansen, 2023). *Formal* logical fallacies occur in the context of deductive reasoning when an argument violates the rules of logic in *structure*, making the conclusion *invalid* regardless of the truth of its premises. These fallacies are inherent in the deductive structure of an argument where the premises are intended to provide definitive support for the conclusion. A typical example is the fallacy of [Affirming the Consequent](#), where the logic incorrectly assumes that if "If A, then B" is true, then the presence of B must necessarily mean the presence of A. This error in reasoning fails to acknowledge that B could arise from causes other than A, thereby making the argument logically *invalid*.

In contrast, *informal* logical fallacies are not limited to deductive arguments and arise from errors in reasoning that involve the *content* of the argument rather than its logical structure. These fallacies often depend on the relevance of the premises or the context, making the argument *weak* rather than strictly invalid. Examples include [Ad Hominem](#) attacks, where a rebuttal is directed against the person rather than addressing the person's arguments, or an [Appeal to Authority](#), where the truth of a claim is erroneously justified based solely on the authority of the claimant. Informal fallacies can be more subtle and context-dependent, making them harder to detect than formal fallacies. Figure 1 shows some examples of common informal logical fallacies.

### 3 Theoretical Background

Logical Fallacy ( <i>in Latin</i> )	Definition
Against the Person ( <i>Argumentum Ad Hominem</i> )	Attacking the person or some aspect of the person making the argument instead of addressing the argument directly.
Appeal to Authority ( <i>Argumentum Ad Verecundiam</i> )	Using an alleged authority who is not really an authority on the facts relevant to the argument as evidence.
Appeal to Popularity ( <i>Argumentum Ad Populum</i> )	Affirming that something is real or better because the majority in general or of a particular group thinks so.
Appeal to Emotion ( <i>Argumentum Ad Passiones</i> )	Manipulating the reader's emotions in order to win the argument in place of a valid reason.
Hasty Generalization ( <i>Secundum Quid</i> )	Drawing a conclusion about all or many instances of a phenomenon on the basis of one or a few instances of that phenomenon.
Questionable Cause ( <i>Non Causa Pro Causa</i> )	Concluding that one thing caused another simply because they are regularly associated.
Red Herring ( <i>Ignoratio Elenchi</i> )	Attempting to divert the reader's attention from the original argument by offering a different point.

Figure 1: Common Informal Logical Fallacies (Lim & Perrault, 2024).

It is often not straightforward to categorize a particular fallacy, as more than one category may apply. While different authors mostly agree on the definition of fallacious arguments as well as the broad categorizations of fallacies, there are differences in the fine-grained categorizations (Sourati, Prasanna Venkatesh, et al., 2023). A comprehensive taxonomy and analysis of logical fallacies can be found on the [Fallacy Files](#) website, including aliases, etymology, history, examples, and sub-fallacies for every category (Curtis, 2001).

### 3.3 Large Language Models (LLMs)

Large language models (LLMs) have become a cornerstone in artificial intelligence, particularly in [natural language processing \(NLP\)](#), due to their ability to understand and generate human-like text. This short introduction explores their foundation in neural networks and deep learning, the architecture and training of LLMs, important milestones, and current reasoning capabilities and limitations.

#### 3.3.1 Neural Networks and Deep Learning

LLMs are based on neural networks which are computational models inspired by the human brain, consisting of interconnected nodes or neurons (Taye, 2023). These networks can learn complex patterns from data, making them ideal for tasks involving large datasets. Deep learning, a subset of machine learning, involves training neural networks

### 3 Theoretical Background

with many layers to automatically learn hierarchical representations of data. This capability allows deep learning models to capture intricate patterns and dependencies in data, which is crucial for understanding and generating human language.

#### 3.3.2 Architecture and Training

At the core of LLM architecture is the transformer model, introduced by the seminal paper "Attention is All You Need" (Vaswani et al., 2017). This architecture leverages the attention mechanism to process entire sequences simultaneously, greatly improving efficiency and performance over previous models like [recurrent neural networks \(RNNs\)](#) and [long short-term memory networks \(LSTMs\)](#). Key components include:

- Input Layer: Tokenization breaks down input text into smaller units called tokens, which are then converted into numerical representations.
- Embedding Layer: Each token is mapped to a dense vector representing its semantic meaning.
- Transformer Blocks: These consist of multi-head self-attention mechanisms and feedforward neural networks, allowing the model to learn complex hierarchical representations.
- Output Layer: The model predicts the next token in a sequence or fills in a masked token, using a probability distribution over the vocabulary.

Training an LLM is a complex process and can be broadly divided into three main phases: self-supervised learning, supervised learning, and reinforcement learning (Bach, 2024). In the self-supervised learning phase, the model is exposed to large amounts of unannotated text data. The primary task of the model is to predict missing parts of the text, such as the next word in a sentence. This task helps the model learn the structure and nuances of language by identifying patterns and relationships between words and phrases. The supervised learning phase involves the use of labeled data in which the model learns to perform tasks based on given instructions. This phase enhances the model's ability to generalize across new tasks and respond accurately to user input. The final phase is based on pivotal work by Christiano et al. (2017) and often involves reinforcement learning on human feedback (RLHF), which aims to refine the model's behavior by encouraging desirable outputs and discouraging undesirable ones.

### 3 Theoretical Background

#### 3.3.3 Important Milestones

The development of LLMs has been marked by several key milestones:

- Word Embeddings and Word2vec (2013): The introduction of word embeddings was a significant advancement in natural language processing, allowing efficient learning of word representations from raw text (Mikolov et al., 2013).
- Transformers (2017): The introduction of transformers revolutionized LLMs by enabling models to focus on different parts of an input sequence when making predictions (Vaswani et al., 2017).
- BERT (2018): Developed by Google, [bidirectional encoder representations from transformers \(BERT\)](#) advanced natural language understanding tasks by providing contextual understanding through bidirectional encoding (Devlin et al., 2019).
- GPT Series (2020): OpenAI's [generative pretrained transformer \(GPT\)](#) models have progressively increased in size and capability, with GPT-3 boasting 175 billion parameters (Brown et al., 2020).
- Vision Capabilities (2021): Integration of vision capabilities allows for multimodal processing, expanding the range of applications (Radford et al., 2021).
- Open-Source Models (2023): The rise of open-source models such as Meta's LLaMA 3 offers transparency and innovation opportunities (Touvron et al., 2023).

To put the current state of LLMs in a broader context using a speculative five-step roadmap (Cook, 2024), OpenAI and other leading providers may be approaching the second level (reasoning AI) and some are venturing into autonomous AI (e.g. Anthropic, 2024):

1. Conversational AI: This stage involves AI systems that can engage in conversational language with humans, exemplified by tools like ChatGPT and AI customer service agents.
2. Reasoning AI: Future AI systems, known as "reasoners," will be capable of performing problem-solving tasks comparable to those of a highly educated human.
3. Autonomous AI: At this level, AI "agents" can operate independently for extended periods, managing tasks without constant human oversight and potentially self-correcting when issues arise.

### *3 Theoretical Background*

4. Innovating AI: These "innovators" can independently develop and improve processes, offering critical thinking and innovative solutions beyond executing predefined tasks.
5. Organizational AI: The ultimate stage where AI systems, termed "organizations," can perform all functions of a human-run organization, collaborating to enhance efficiency and effectiveness without human intervention.

## 4 Literature Review

In addition to the literature mentioned in the [Theoretical Background](#) chapter, this literature review will discuss logical fallacy datasets, since the study of logical fallacies in recent years has focused on dataset construction and fallacy classification (Li et al., [2024](#)). The second part addresses the logical reasoning capabilities and limitations of large language models (LLMs).

### 4.1 Logical Fallacy Datasets

The study of detecting logical fallacies has been supported by a variety of datasets, created to improve our understanding of fallacy identification across different domains. These datasets vary in size, scope, and type of annotated fallacies, revealing both advancements and challenges in this research area.

A significant number of datasets focus on specific logical fallacies or sets of fallacies within specific contexts. For instance, Habernal et al. ([2017](#)) presented *Argotario*, a serious multilingual and educational game that deals with fallacies in everyday argumentation. They later expanded on their work by annotating 430 arguments with six classes (Habernal, Pauli, & Gurevych, [2018](#)), and created a new dataset from argumentation data in the *Change my View* subreddit (Habernal, Wachsmuth, et al., [2018](#)), focusing on ad hominem attacks. Similarly, Sahai et al. ([2021](#)) compiled a corpus from Reddit, covering eight fallacy categories. In the domain of online news, Da San Martino et al. ([2019](#)) contributed a dataset of 451 annotated articles featuring 18 propaganda techniques, many aligning with logical fallacies. Expanding to political discourse, Goffredo et al. ([2022](#)) built on work by Haddadan et al. ([2019](#)) by introducing a dataset sourced from American political debates with 6 main and 14 sub-categories.

More recently, datasets including misguided climate change arguments and the COVID-19 pandemic were compiled. For example, Jin et al. ([2022](#)) introduced the LOGIC dataset with 2449 text samples categorized into 13 fallacy types, together with an additional challenge set with climate change claims, using an annotation approach inspired by first-order logic. Musi et al. ([2022](#)) and Alhindi et al. ([2022](#)) focused on misinformation contexts, annotating COVID-19 and climate change articles with ten fallacy types. Also

#### 4 Literature Review

focusing on the pandemic, Bonial et al. (2022) applied the annotation schema of Habernal et al. (2017) to 226 articles on COVID-19, highlighting logical fallacies in different mediums.

With the advent of LLMs, researchers have begun exploring the role of these models in dataset creation and benchmark evaluation. Payandeh et al. (2023) introduced LOGICOM, a dataset with over 5000 pairs of logical vs. fallacious arguments designed to evaluate the robustness of LLMs against logical fallacies in debate contexts where a persuader and a debater agent engage in a multi-round debate on a controversial topic. Research by Veselovsky et al. (2023) and Li et al. (2024) explored how LLMs can assist in generating datasets, either directly through the generation of synthetic examples or by using LLMs in the annotation workflow. Integrating LLMs into crowdsourcing platforms has shown promise in helping workers tackle complex annotation tasks, while also improving transparency by tracking prompts and LLM-generated output (Bartolo et al., 2022; Veselovsky et al., 2023). Very recently, Yeh et al. (2024) introduced CO-COLOFA, the currently largest logical fallacy dataset containing 7706 comments for 648 news articles, with each comment labeled for fallacy presence and type. The noteworthy size of the dataset was achieved by employing crowd workers and providing them with an LLM-powered assistant, replacing the data annotation process with data generation. However, the dataset covers only eight common logical fallacies, which limits its use for a systematic study of a wide range of fallacy types.

The particularly comprehensive FALLACIES dataset with 4640 arguments, chosen as a data source for this study, was created by Hong et al., 2024 and proposes a taxonomy encompassing 232 fallacy types, including both formal and informal fallacies. Although the applicability for fallacy classification is limited due to the overwhelming number of fallacy types, the dataset contains both valid and invalid arguments, ideal for fine-tuning models and benchmarking performance on binary fallacy detection. In addition, the MAFALDA dataset introduced by Helwe et al. (2024) was chosen as another data source due to its complementary characteristics to the FALLACIES dataset: it features a smaller number of 200 longer annotated texts with 23 more distinct fallacy types and allows for multiple overlapping annotations and fallacy interpretations. In contrast to the artificially constructed arguments in the FALLACIES dataset, the MAFALDA dataset contains real-world examples sourced from four other datasets, rather than artificially constructed one-sentence arguments. These two datasets will be described in more detail in Chapter 5.

Despite these advances, existing datasets exhibit notable limitations. The definitions and typologies of fallacies often diverge across datasets, leading to incompatible sets of

fallacy types. Studies such as Helwe et al. (2024) strive to unify these efforts, however, a truly standardized and comprehensive taxonomy remains elusive. Moreover, limited dataset sizes restrict the task-specific fine-tuning of LLMs. The integration of these models into dataset creation workflows represents a promising direction to address these challenges, enabling the generation of large-scale, diverse, and well-annotated text corpora.

In summary, the field of logical fallacy detection has benefited from a wide range of datasets tailored to specific domains and purposes. However, issues such as inconsistent taxonomies and scalable data annotation remain, offering opportunities for future research and innovation.

## 4.2 Logical Reasoning Capabilities of LLMs

Logical reasoning is a crucial component of human intelligence, and integrating it into natural language understanding systems has been a focus since the inception of artificial intelligence. Researchers have pursued various methods, including rule-based systems, symbolic approaches, and the fine-tuning of LLMs. Traditional approaches involved symbolic systems using first-order logic or natural logic to address inference tasks (MacCartney & Manning, 2007). However, rule-based models often struggle due to their reliance on hand-crafted rules and theorem provers. These methods face difficulties with ambiguity and scalability, making them less effective with real-world natural language data (Liu et al., 2023).

With the rise of large neural network models, large-scale [natural language inference \(NLI\)](#) datasets such as SNLI (Bowman et al., 2015) and MNLI (Williams et al., 2018) have become crucial benchmarks in the development of LLMs. The introduction of transformer-based models led to a major breakthrough in [natural language processing \(NLP\)](#). These models now outperform humans on many [NLI](#) and [machine reading comprehension \(MRC\)](#) tasks, but even the best performing LLMs including GPT-4 still have difficulty handling [out-of-distribution \(OOD\)](#) datasets (Liu et al., 2023).

LLMs have demonstrated significant advancements in reasoning capabilities in various domains. Recent approaches like the Consistency-driven Rationale Evaluation for Self-Training (CREST) have improved LLM reasoning abilities by evaluating self-generated rationales (explanations) through follow-up questions. This method improves the logical robustness and correctness of rationales, leading to better reasoning performance compared to previous self-training approaches (Lee et al., 2024). LLMs have shown emergent abilities in cross-modal reasoning, where they integrate multiple modalities such as

#### *4 Literature Review*

images, audio, and video with language. Frameworks like X-InstructBLIP take advantage of these capabilities by aligning multi-modal instruction-aware representations to LLMs, achieving significant advances in visual reasoning tasks (Panagopoulou et al., 2024). LLMs seem to be able of self-critique, iteratively improving their solutions. Although this capability is not fully realized, iterative prompting has shown potential in enhancing reasoning and planning tasks when combined with external verification mechanisms (Stechly et al., 2024).

However, LLMs also face notable limitations that affect their performance. For example, they struggle with compositional relation reasoning, which involves understanding complex relationships between different components of a problem. Evaluations in multiple languages have highlighted these limitations, indicating that LLMs often fail to adapt to diverse linguistic contexts (Zhao & Zhang, 2024). Despite improvements on benchmarks like GSM8K, LLMs also exhibit fragility in mathematical reasoning. Their performance declines significantly when faced with variations in numerical values or increased complexity in question clauses, suggesting limitations in genuine logical reasoning capabilities (Mirzadeh et al., 2024). And finally, multimodal LLMs face challenges in abstract visual reasoning tasks such as Bongard problems, which require combining visual and verbal descriptions. These models often struggle with using new information effectively and maintaining context in dialogues, particularly with synthetic visual input (Małkiński et al., 2024).

In summary, while LLMs have made impressive strides in reasoning capabilities, they continue to face challenges in areas that require deep compositional understanding and abstract reasoning. The ongoing research aims to address these limitations through innovative training frameworks and cross-modal integration strategies.

## 5 Datasets

### 5.1 FALLACIES Dataset

The FALLACIES dataset compiled by Hong et al. (2024) contains 4640 samples of logical fallacies, divided into 232 different types (data source: Hong, 2024). The fallacies are categorized into a fine-grained hierarchical taxonomy. This dataset includes both valid and fallacious arguments, referred to as “reasoning steps” by the authors: ten fallacious and ten correct steps for each of the 232 types of fallacies. The fallacies are divided into four formal and five informal fallacy categories shown in Figure 2.

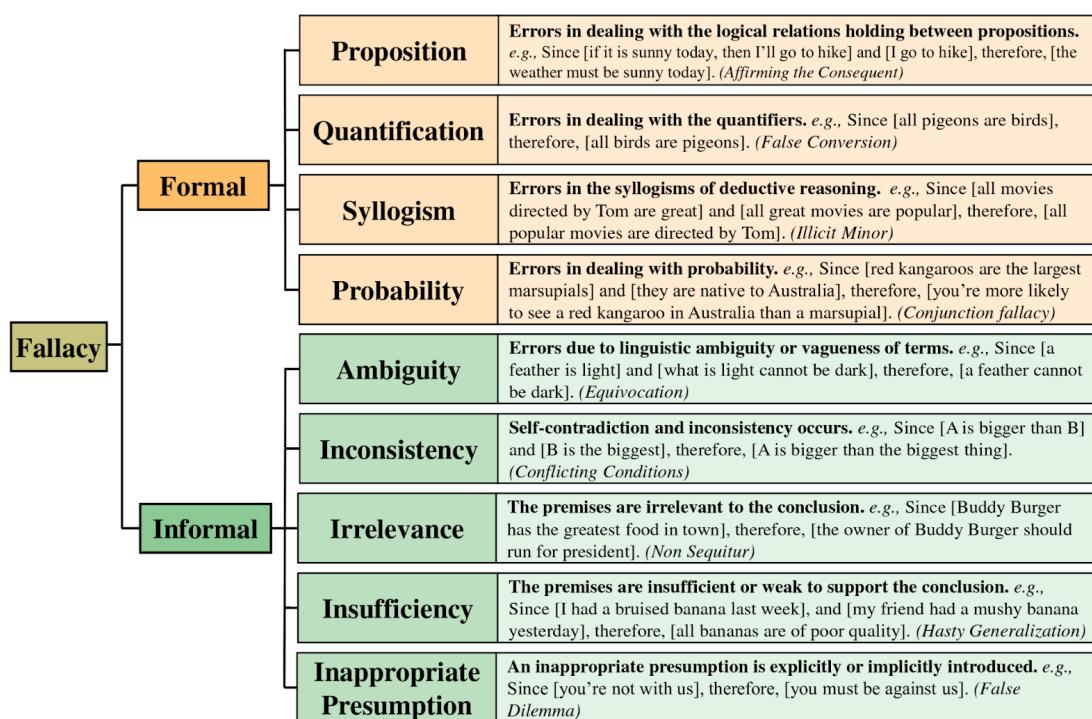


Figure 2: Taxonomy of Fallacies in the FALLACIES Dataset. Square brackets indicate the premises and conclusions (Hong et al., 2024).

This dataset stands out due to its comprehensive coverage of fallacy types, clear definitions, and detailed hierarchical categorization. It was constructed by human experts,

## 5 Datasets

with GPT-4 providing candidate examples for each fallacy type. An additional team of three human experts re-annotated randomly chosen samples, achieving high interrater reliability (Cohen’s Kappa: 0.856) and thus ensuring high data quality.

As shown in Figure 3, the FALLACIES dataset compares favorably with older datasets due to the larger number of samples and the more fine-grained taxonomy. It constitutes a substantial improvement over the LOGIC dataset and is therefore a suitable choice for fallacy benchmark tests. However, the large number of different fallacy types results in considerable semantic overlap between types, which makes classification challenging.

Dataset	Number of Fallacies	Number of Steps	Taxonomy of Fallacy	Granularity of Reasoning	Explicit Premises and Conclusions	Identifying Fallacy from Correct Reasoning
Stab and Gurevych (2017)	1	1,029	No	Coarse	No	Yes
Habernal et al. (2018)	1	2,085	No	Coarse	No	Yes
Jin et al. (2022)	13	2,449	Coarse	Coarse	No	No
FALLACIES (Ours)	232	4,640	Fine & Hierarchical	Fine	Yes	Yes

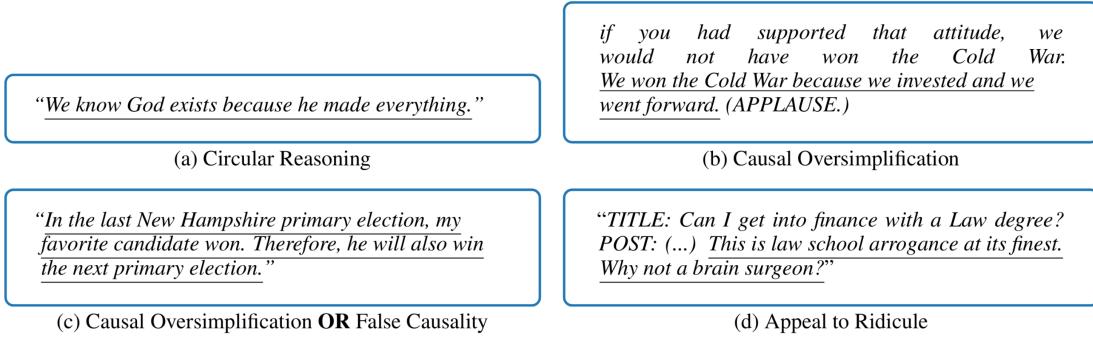
*Figure 3: Comparison of Logical Fallacy Datasets (Hong et al., 2024).* The FALLACIES dataset represents a significant improvement over datasets used in previous studies, making it a suitable choice for benchmarking fallacy detection tasks.

## 5.2 MAFALDA Dataset

The Multi-level Annotated Fallacy Dataset (MAFALDA) introduced by Helwe et al. (2024) comprises 200 manually annotated texts containing 260 instances of logical fallacies, each with justifications and explanations (data source: Helwe, 2024). After careful evaluation, 63 of the 200 texts were considered non-fallacious. The annotators include five human experts, who are also the authors of the paper. No LLMs or crowd workers were involved in the annotation process, as crowd workers often use ChatGPT (Veselovsky et al., 2023). Additionally, the dataset contains 9545 non-annotated texts.

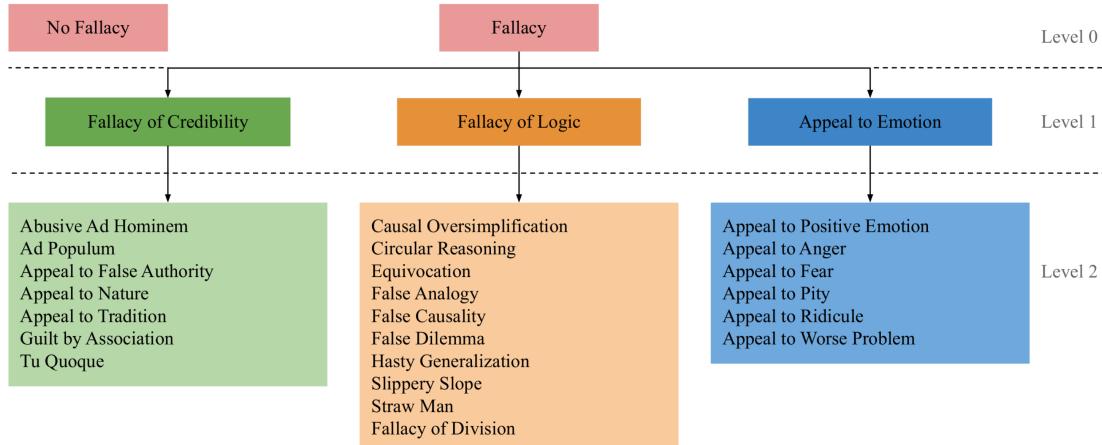
The dataset was created by consolidating four existing datasets, including U.S. presidential debates, social media interactions, and climate change discussions (Da San Martino et al., 2019; Goffredo et al., 2022; Jin et al., 2022; Sahai et al., 2021), offering a unified framework for fallacy analysis. Figure 4 illustrates some concrete text examples with annotations.

## 5 Datasets



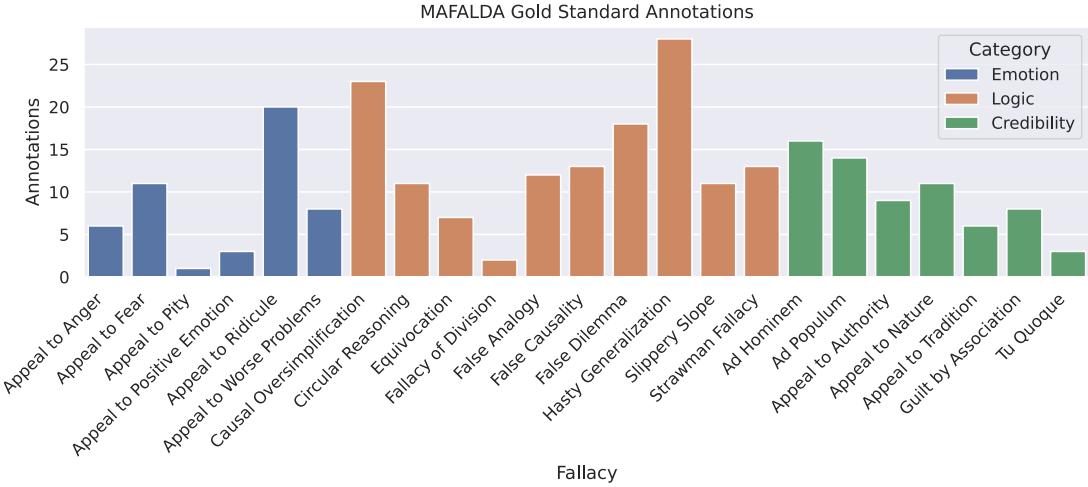
**Figure 4: Examples of Fallacies in the MAFALDA Dataset (Helwe et al., 2024).** The MAFALDA dataset consists of 200 manually annotated texts with 260 logical fallacies, justifications, and explanations. It consolidates four existing datasets to provide a unified framework for fallacy analysis. Example 1a is from Jin et al. (2022), 1b from Goffredo et al. (2022), and 1d from Sahai et al. (2021).

As shown in Figure 5, fallacies are categorized into 23 distinct types (level 2), which are organized under three main Aristotelian categories (level 1): Emotion (Pathos), Logic (Logos), and Credibility (Ethos). Finally, arguments are classified as either fallacious or non-fallacious (level 0). As Helwe et al. (2024) plausibly argue, the smaller number of fallacy types (23 vs. 232) compared to the FALLACIES dataset by Hong et al. (2024) is a strength rather than a weakness, reducing semantic overlap between types. The number of annotations is more or less evenly distributed between the three level 1 categories, but varies considerably between level 2 fallacy types, as shown in Figure 6.



**Figure 5: Taxonomy of Fallacies in the MAFALDA Dataset (Helwe et al., 2024).** Fallacies are categorized into 23 distinct types (level 2) grouped under three categories—Emotion, Logic, and Credibility (level 1)—with a top-level classification as fallacious or non-fallacious (level 0). The reduced number of fallacy types compared to the FALLACIES dataset minimizes semantic overlap.

## 5 Datasets



**Figure 6: Number of Annotations per Fallacy Type in the MAFALDA Dataset.** Annotation counts vary significantly among level 2 types.

A distinguishing feature of MAFALDA is its disjunctive annotation scheme, which allows multiple fallacy annotations to coexist for the same text span, acknowledging the inherent complexity and potential overlap in fallacious reasoning. However, it also increases the complexity of calculating precision and recall evaluation metrics, which will be described in Section 6.4 of the [Methods](#) chapter.

The quality and utility of the dataset have been evaluated by LLM performance testing and human expert benchmarking by Helwe et al. (2024). However, the 200 annotated texts are a relatively small sample size. In contrast to the large FALLACIES dataset described in Section 5.1, the MAFALDA dataset is not suitable for training or fine-tuning, but serves as a benchmark for evaluating the ability of LLMs and humans to detect and classify logical fallacies, as mentioned by the authors.

## 6 Methods

In order to address the research questions described in Chapter 2, a selection of models was evaluated on three different tasks: fallacy identification, fallacy classification, and fallacy search. The following sections describe model selection, tasks, experiments, and performance metrics in more detail. The fallacy identification and classification tasks were performed using the FALLACIES dataset described in Section 5.1, and for the fallacy search task the MAFALDA dataset outlined in Section 5.2 was used.

### 6.1 Model Selection

The selection of large language models (LLMs) for this thesis is guided by their ability to handle the complex tasks of detecting and classifying logical fallacies, which require advanced reasoning, linguistic nuance, and computational efficiency. The chosen models are shown in Table 1 and represent a diverse set of capabilities that align with the requirements of this research, ensuring a balance between performance, scalability, and cost-effectiveness. For analysis, the selected LLMs were categorized into large models (with disclosed or estimated parameter counts exceeding 20 billion) and small models (with fewer parameters). The o1-mini model was placed in a separate "Reasoning" group because it uses a significantly different approach with internal chain-of-thought reasoning. The context window sizes, which determine the maximum number of input tokens a model can process, were sourced from the official provider's model documentations.

## 6 Methods

Provider	Name	Version	Group	Context (k)
OpenAI	GPT-4	0613	Large	8
	GPT-4o	2024-08-06	Large	128
	GPT-4o Mini	2024-07-18	Small	128
	o1-mini	2024-09-12	Reasoning	128
Anthropic	Claude 3.5 Sonnet	20240620	Large	200
	Claude 3 Opus	20240229	Large	200
	Claude 3 Haiku	20240307	Small	200
Google	Gemini 1.5 Pro	002	Large	2097
	Gemini 1.5 Flash	002	Small	1048
	Gemini 1.5 Flash 8B	001	Small	1048
Meta	Llama 3.1 70B	Instruct	Large	128
	Llama 3.1 8B	Instruct	Small	128
Mistral AI	Mistral Large	2407	Large	128
	Mistral Small	2409	Small	32

*Table 1: Model Selection for the Fallacy Identification and Classification Tasks.*

OpenAI’s GPT-4 series, including GPT-4, GPT-4o, GPT-4o Mini, and o1-mini, was selected for its exceptional reasoning capabilities and versatility. These models are particularly well-suited for tasks involving nuanced understanding, such as logical reasoning and argument analysis. The o1-preview model was excluded due to an exceedingly high token cost. Despite the high cost, the relatively outdated GPT-4 model was included to replicate the accuracy results obtained by Hong et al. (2024) as a sanity check. Smaller variants such as GPT-4o Mini and o1-mini provide cost-effective alternatives, making them ideal for experiments that require lower computational resources or fine-tuning for specific use cases.

The Anthropic Claude 3 models, Sonnet, Opus, and Haiku, were chosen for their advanced linguistic capabilities and robust reasoning performance. These models excel at understanding complex linguistic constructs, such as metaphors and implied meanings, which are often integral to identifying logical fallacies. Claude 3 Opus has demonstrated strong results in graduate-level benchmarks such as GPQA and MMLU, highlighting its suitability for analyzing intricate arguments (Akpan, 2024).

Google’s Gemini 1.5 series (Pro and Flash) offers unique advantages that complement the requirements of this research. Gemini 1.5 Pro supports ultra-long context processing of up to two million tokens per input, making it ideal for analyzing lengthy documents

## 6 Methods

or debates where logical fallacies may occur over extended arguments. Features such as function calling and JSON output facilitate structured data extraction from unstructured arguments.

Meta's Llama 3.1 (70B and 8B Instruct) and Mistral AI's models (Large and Small) were included in the selection for a comparative analysis between open-source and proprietary closed-source models. Their open-source nature allows for fine-tuning on custom datasets specific to logical fallacies while maintaining control over the model's deployment environment.

### 6.2 Fallacy Identification

The fallacy identification task is a binary choice in which the model needs to decide whether a given statement is a fallacy or not. This is the simplest of the three tasks and Hong et al. (2024) have reported an average accuracy of 87.7% for GPT-4, surpassing smaller tested models by a substantial margin. In this thesis, GPT-4 was included in the first experiment to replicate these findings and validate the experimental methodology, but excluded from the other experiments due to its high cost of usage.

#### 6.2.1 Experiment 1.1: Fallacy Identification with Zero-Shot Prompt

In zero-shot prompting, the model was asked to perform the identification task without any prior examples or specific training on the task at hand. The model relies solely on its preexisting knowledge and understanding derived from its training data. This method tests the model's inherent ability to generalize and apply its knowledge to new, unseen tasks without additional guidance. The same zero-shot prompt was used as in Hong et al. (2024) to obtain comparable results. The [text] placeholder was replaced with reasoning steps from the FALLACIES dataset described in Section 5.1 to generate the 4640 final prompts:

---

Is the following reasoning step correct? You can only answer "Yes" or "No".  
[text]

---

#### 6.2.2 Experiment 1.2: Fallacy Identification with Few-Shot Prompt

Few-shot prompting involves providing the model with a small number of examples before asking it to perform the task. These examples serve as a brief introduction or

## 6 Methods

context to help the model understand what is expected. Few-shot prompting aims to enhance the model’s performance by giving it minimal but relevant context, which can be particularly beneficial for tasks that require some level of pattern recognition or domain-specific knowledge.

Lim and Perrault (2024) have used few-shot prompting for fallacy classification, providing three examples per fallacy category in a single prompt. However, the FALLACIES dataset contains 232 fallacy types and including several examples per type would result in very large prompts. Instead of providing examples for all fallacy types in each prompt, Hong et al. (2024) provided the same two positive and two negative examples for each few-shot prompt. Again, the same exact prompt was used to obtain comparable results:

---

Is the following reasoning step correct? You can only answer "Yes" or "No".  
Since if it's raining then the streets are wet and it's raining now, therefore, the streets are wet.  
Yes.  
Since I found a shell on the beach and this shell was beautifully shaped and colored, therefore, all shells are beautifully shaped and colored.  
No.  
Since I am at home or I am in the city and I am at home, therefore, I am not in the city.  
No.  
Since heavy snowfall often leads to traffic jams and traffic jams cause delays, therefore, heavy snowfall can lead to delays.  
Yes.  
[text]

---

Due to budget constraints, the more expensive GPT-4, o1-mini, and Claude 3 Opus models were excluded from this and all following experiments. The Mistral Large model was excluded due to an exceedingly high number of invalid responses, as described in Section 6.7.

### 6.2.3 Experiment 1.3: Fallacy Identification with Chain-of-Thought Prompt

Chain-of-thought prompting involves guiding the model through a step-by-step reasoning process to arrive at a conclusion. Instead of simply providing examples, this method encourages the model to articulate its thought process, making its reasoning explicit. This approach can be particularly powerful for tasks that require logical reasoning and multi-step problem-solving (Wei et al., 2023). By breaking down the task into smaller, manageable steps, chain-of-thought prompting helps the model to systemati-

## 6 Methods

cally tackle complex problems and improve its overall performance. Again, the same chain-of-thought prompt was used as in Hong et al. (2024):

---

Is the following reasoning step correct?  
Let's think step by step and then answer "Yes" or "No".  
[text]

---

The more expensive GPT-4, o1-mini, and Claude 3 Opus models were excluded from the experiment to save cost, especially because of the large number of output tokens generated due to chain-of-thought prompting.

### 6.2.4 Experiment 1.4: Fallacy Identification with Fine-Tuning

Fine-tuning is a specialized method used to adapt pre-trained LLMs to specific tasks or datasets. This process involves continuing the training of a model on a new dataset that closely represents the target task, thereby refining the model's parameters to perform better on similar data. Fine-tuning stands in contrast to the previously discussed prompting methods that do not alter the model's underlying weights but instead manipulate the input to achieve desired output. Sourati, Prasanna Venkatesh, et al. (2023) used fine-tuning based on the LOGIC dataset compiled by Jin et al. (2022) and applied the trained models for fine- and coarse-grained logical fallacy classification in a zero-shot fashion to the corresponding LOGIC Climate data.

In this thesis, fine-tuning was performed using a train-test split on the FALLACIES dataset. For each of the 232 fallacy types, eight reasoning steps were included in the training set (1856 steps), two reasoning steps were included in the validation set (464 steps) used to track progress during fine-tuning, and 10 reasoning steps were reserved for the test set (2320 steps) to evaluate accuracy on. Each of the three sets contained 50% correct and 50% incorrect steps. Each training sample included the system prompt, reasoning step, and correct answer, as in the following example:

```
{  
  "messages": [  
    {  
      "role": "system",  
      "content": "You are a logical fallacy detector. Given a reasoning step, your  
      task is to identify whether it is correct or not."  
    },  
    {  
      "role": "user",  
      "content": "Is the following reasoning step correct? You can only answer \"Yes  
      \" or \"No\".\nSince John accuses Mary of being terrified of the beewolf, and  
      Mary remains silent without providing any evidence, therefore, Mary must  
      indeed be terrified of the beewolf."  
    },  
    {  
      "role": "assistant",  
      "content": "No"  
    }  
  ]  
}
```

---

The training and validation sets were prepared as JSONL files and sent to the OpenAI developer platform according to the [OpenAI Fine-tuning](#) instructions (“OpenAI Platform”, [n.d.](#)) for fine-tuning a GPT-4o Mini model over 3 epochs. The default configuration was used, and no hyperparameter tuning was conducted. The fine-tuned GPT-4o Mini model was evaluated on the test set only. For details, see the [fine\\_tuning.ipynb](#) jupyter notebook in the accompanying GitHub repository ([Imfeld, 2024a](#)).

### 6.2.5 Experiment 1.5: Fallacy Identification with Rephrased Prompt

In order to test the effects of slightly rewording a prompt and as an additional sanity check, an alternative zero-shot prompt was used on a subset of models (GPT-4o, Claude 3.5 Sonnet, and Gemini 1.5 Flash). The question “Is the following reasoning step correct?” used in experiment 1.1 was reworded in the following prompt as follows:

---

Does the following reasoning step contain a logical fallacy? You can only answer  
“Yes” or “No”.

[text]

---

## 6 Methods

The main motivation was to test whether the false alarm rate could be reduced by specifically asking about logical fallacies instead of correct reasoning, which may be interpreted in a broader sense.

### 6.2.6 Experiment 1.6: Fallacy Identification Reliability Test

Finally, the zero-shot experiment 1.1 described in Section 6.2.1 was repeated with a subset of small models (GPT-4o Mini, Claude 3 Haiku, and Gemini 1.5 Flash) to ensure reproducibility of the results. For all experiments, the model temperature was set to zero, aiming for deterministic responses and allowing reproducibility of the results obtained.

In natural language generation with LLMs, the temperature parameter controls the randomness of model outputs (Renze & Guven, 2024). A temperature of zero theoretically forces the model to always choose the highest-probability response, making its output deterministic. However, in practice, implementation details may still cause slight variations in model output. This experiment was conducted to quantify the amount of variation in the model output despite setting a temperature of zero.

## 6.3 Fallacy Classification

The second task of fallacy classification is much more difficult than the binary task of fallacy identification. Rather than giving a Yes/No answer, the models need to decide which of the 232 types a given fallacy belongs to. The classification accuracies reported by Hong et al. (2024) were much lower than for fallacy identification, with GPT-4 achieving an overall accuracy of 35.0% and most models reaching less than 10%.

### 6.3.1 Experiment 2.1: Fallacy Classification with Zero-Shot Prompt

Following Hong et al. (2024), the same zero-shot prompt was used, listing all 232 fallacy types:

---

You are a logical fallacy classifier. Given an incorrect reasoning step, your task is to identify its type of fallacy.

Answer by choosing one of these fallacies:

(1) Affirming the Consequent

...

(232) Alleged Certainty

You should only answer the name of the fallacy.

What type of fallacy does the following reasoning step belong to?

[text]

---

In order to meaningfully assign a fallacy type to an argument, the argument must be fallacious in the first place. Therefore, the classification task was performed only on the 2320 *incorrect* reasoning steps, excluding the correct ones and thereby splitting the dataset in half.

### 6.3.2 Experiment 2.2: Fallacy Classification with Fine-Tuning

The same approach was used as for fine-tuning the fallacy identification model described in Section 6.2.4. For fine-tuning a GPT-4o Mini model for fallacy classification, the train-test split included only the 2320 *incorrect* reasoning steps, resulting in half the number of samples in training, validation, and test tests. A slightly different system prompt was used, instructing the model to identify the fallacy type instead of giving a Yes/No response:

---

You are a logical fallacy classifier. Given an incorrect reasoning step, your task is to identify its type of fallacy.

---

## 6.4 Fallacy Search

The fallacy search task sets the foundation for the practical application described in Chapter 8. Using the MAFALDA dataset, which includes real-world text annotated with disjunctive text span labels, the aim is to evaluate performance through a repeatable, fast, and cost-effective benchmark test. This allows for iterative prompt engineering to improve results, rather than blind guessing.

### 6.4.1 Experiment 3.1: Fallacy Search with Zero-Shot Prompt

Helwe et al. (2024)—the authors of the paper introducing the MAFALDA dataset—used a *separate prompt for each sentence* in the input text to let the model decide whether it was part of a fallacious argument or not, and if so, to classify the sentence into one of the 23 listed fallacy types. The exact prompt is shown in their appendix H. Since the model responded with free unstructured text, a complex regular expression was necessary to extract the fallacy type.

The present study introduces a different approach, using a *single prompt* to search for fallacies in a text and analyze them in a more complex way. The model responds with structured output instead of free text, including a list of identified fallacies and their

## 6 Methods

analysis. Structured output not only facilitates statistical evaluation, but also provides a solid basis for the practical application described in Chapter 8. The following system prompt was refined over several iterations and describes the task in detail:

---

You are an expert at detecting and analyzing logical fallacies. Your task is to detect and analyze logical fallacies in the provided text.

Output Format:

Provide your analysis in JSON format with the following structure for each identified fallacy:

```
{  
  "fallacies": [  
    {  
      "fallacy": "<fallacy_type>",  
      "span": "<text_span>",  
      "reason": "<reason>",  
      "defense": "<defense>",  
      "confidence": <confidence>  
    }  
  ]  
}
```

Response Fields:

1. <fallacy\_type>: Only use fallacies from this approved list: Appeal to Anger, Appeal to Fear, Appeal to Pity, Appeal to Positive Emotion, Appeal to Ridicule, Appeal to Worse Problems, Causal Oversimplification, Circular Reasoning, Equivocation, Fallacy of Division, False Analogy, False Causality, False Dilemma, Hasty Generalization, Slippery Slope, Strawman Fallacy, Ad Hominem, Ad Populum, Appeal to Authority, Appeal to Nature, Appeal to Tradition, Guilt by Association, Tu Quoque
2. <text\_span>:
  - Include the complete context needed to understand the fallacy, but keep the span as short as possible
  - Can overlap with other identified fallacies
  - Must be verbatim quotes from the original text
3. <reason>:
  - Provide clear, specific explanations
  - Include both why it qualifies as a fallacy and how it violates logical reasoning
4. <defense>:
  - Provide the strongest possible charitable interpretation under the assumption that the argument is valid or reasonable, and not a fallacy
  - Consider implicit premises that could validate the argument

## 6 Methods

5. <confidence>: Rate your confidence in each fallacy identification from 0.0 to 1.0, taking into account the reasoning and defense

Guidelines:

- Apply the principle of charity, consider the argument in its strongest form, and avoid over-detection
  - Consider principles of formal logical reasoning when judging the validity of an argument
  - For formal logical arguments, accept premises as true for the sake of the argument
  - Return an empty list if no clear logical fallacies are present
  - Adjust confidence scores downward in proportion to the strength and plausibility of the defense
  - Consider context and implicit assumptions
  - Think step by step
- 

To enforce and validate the structured output, [OpenAI Structured Output](#) was implemented (“OpenAI Platform”, [n.d.](#)), specifying the schema in the form of a Pydantic model. This worked on first try, even when no specific JSON format was provided in the system prompt.

### 6.5 Technical Implementation

All experiments were implemented using the LangChain framework ([Chase, 2022](#)) to access provider-specific API endpoints and API keys using a common interface. For Meta’s Llama models, Hugging Face (“Hugging Face”, [2024](#)) endpoints were used. For details, see the accompanying source code ([Imfeld, 2024a](#)), specifically the [fallacy\\_experiments.ipynb](#) jupyter notebook and the included modules.

### 6.6 Performance Metrics

#### 6.6.1 Fallacy Identification Performance Metrics

The main performance metric for fallacy identification is *accuracy*, which was calculated by averaging the scores (1 for correct, 0 for incorrect) for each fallacy type separately. The accuracies of the 9 subcategories, the categories (formal vs. informal), and overall accuracy were calculated by averaging the accuracies of the aggregation level below. The same macro-averages were calculated by Hong et al. ([2024](#)), giving equal weight to the categories and accounting for the imbalance of the FALLACIES dataset. By calculating

## 6 Methods

the same metrics, the results of the present study are directly comparable to the published benchmark results.

In addition, confusion metrics were calculated by summing up the predictions as true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN). Accuracy was calculated as  $Accuracy = (TP + TN) / (TP + TN + FP + FN)$  which corresponds to the micro-average and differs from the macro-average accuracy above. Precision and recall were calculated with the usual formulas shown in Figure 7, and F1-scores were obtained by taking the harmonic mean of precision and recall:  $F1 = (2 * Precision * Recall) / (Precision + Recall)$ . To check for statistically significant differences between the number of false positives (FP) and false negatives (FN), the McNemar test was used.

		Truth		
		positive	negative	
Prediction	positive	True Positive (TP)	False Positive (FP) <i>(Type I error)</i>	Positive predictive value (PPV) = Precision = TP / (TP + FP)
	negative	False Negative (FN) <i>(Type II error)</i>	True Negative (TN)	Negative predictive value (NPV) = TN / (TN + FN)
		Sensitivity = True positive rate (TPR) = Recall = TP / (TP + FN)	Specificity = 1 - False positive rate (FPR) = TN / (TN + FP) ( $FPR = FP / (TN + FP)$ )	

**Figure 7: Confusion Matrix and Key Metrics.** Precision measures the proportion of correctly predicted positive instances (fallacies in the present study) out of all instances predicted as positive, reflecting the model's accuracy in its positive predictions. Recall (sensitivity) quantifies the proportion of actual positive instances (fallacies) correctly identified by the model.

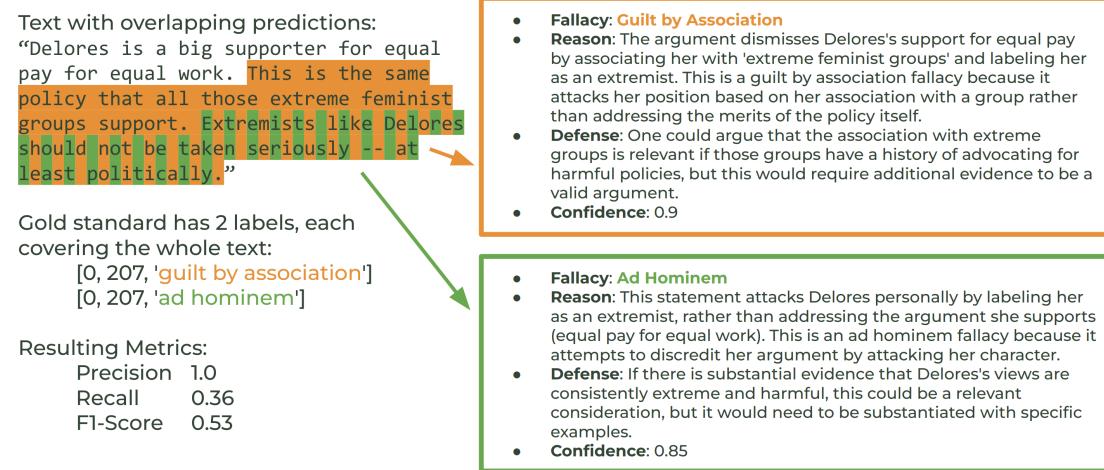
### 6.6.2 Fallacy Classification Performance Metrics

For fallacy classification, the same metrics were computed as for fallacy identification described in the previous section. However, instead of a 2x2 confusion matrix, a 232x232 confusion matrix was calculated, corresponding to the 232 fallacy types. For each fallacy type, the confusion metrics mentioned were calculated. By ordering the fallacy types by recall (sensitivity), the most missed fallacies were identified, and ordering by the number of false positives highlighted fallacies that the LLMs often predicted, but were wrong answers according to the ground-truth labels. In order to identify fallacies with a lot of semantic overlap, the two most common misclassifications were obtained from the confusion matrix for each fallacy type.

### 6.6.3 Fallacy Search Performance Metrics

The performance metrics for fallacy search are based on the disjunctive annotation scheme of the MAFALDA dataset, where multiple fallacy type annotation text spans may overlap. Since this allows for multiple correct classifications of the same text, the calculation of precision and recall is more involved. The main performance metric is the F1-score, defined as the harmonic mean of precision and recall. The exact formulas and proof of certain characteristics are given in Helwe et al. (2024).

Instead of reproducing the math, the metrics will be explained more intuitively with an example shown in Figure 8. The given text consists of three sentences and the MAFALDA gold standard contains two valid annotations ("guilt by association" and "ad hominem"), both spanning the *whole* text. The LLM predicts two fallacies with partially overlapping text spans. The predicted annotations are correct, but the corresponding text spans cover only *parts* of the text. Precision is thus perfect (1.0), since each predicted annotation overlaps with the annotation of the gold standard. However, recall (sensitivity) is penalized, since parts of the gold standard text span annotations were missing in the prediction. Had the model predicted a third fallacy, precision would have been penalized as well.



**Figure 8: Example of Fallacy Search Metrics.** The figure compares two predicted fallacy annotations with the MAFALDA gold standard annotations. While the predicted annotations correctly identify the fallacies, they match only parts of the text spans annotated in the gold standard, resulting in a recall penalty.

In order to determine the inclusive start and end indices, the predicted fallacy text spans were fuzzy-matched against the whole text, allowing for up to five minor differences. Mismatch rates were calculated, and only about 1% of the predicted text spans

## *6 Methods*

could not be matched against the full text. Those mismatches were discarded from the fallacy predictions. For details, see the [fallacy\\_search.ipynb](#) jupyter notebook in the accompanying GitHub repository (Imfeld, 2024a).

As in Helwe et al. (2024), precision and recall were calculated on the three hierarchical levels, starting with the 23 fallacy types at level 2. The metrics for level 1 (Emotion, Logic, and Credibility) and level 0 (Fallacy vs. No Fallacy) were determined by mapping the level 2 fallacy labels to the respective categories.

For reasons not mentioned in the paper, Helwe et al. (2024) added "no fallacy" annotations to all uncovered text spans, both for the gold standard and for model predictions. By using different "no-fallacy" labels for the gold standard and the predictions, a bug seems to have been introduced, distorting their results to a certain extent. Therefore, the present study did not include annotations for uncovered text spans, preserving meaningful metrics verified with unit tests. The issue was [reported](#) to Helwe et al. (2024) on GitHub.

### **6.7 Sanity Checks**

Sanity checks were performed to count the number of missing and invalid responses in each experiment. In very rare cases, models refused to answer due to perceived insensitive language. For few-shot prompts, small models sometimes got confused and responded e.g. with "Yes. No. No. Yes. Yes." instead of making a Yes/No decision. For chain-of-thought prompts where the models include a reasoning process before giving a final answer, it was sometimes not possible to extract a Yes/No decision. If the missing response rate exceeded 5%, the corresponding model was excluded from the experiment. This was only the case for the Mistral Large model in the few-shot experiment described in Section 6.2.2. For details on response extraction and invalid prediction rates, see the [fallacy\\_identification.ipynb](#) and [fallacy\\_classification.ipynb](#) jupyter notebooks in the accompanying GitHub repository (Imfeld, 2024a).

## 7 Results

This chapter presents the findings of the experiments conducted to evaluate the performance of large language models (LLMs) in fallacy identification, classification, and search. The appendix provides additional results for the rephrased prompt ([Appendix A](#)) and the reliability test experiments ([Appendix B](#)). The jupyter notebooks in the accompanying GitHub repository provide additional context, the datasets and model responses, and the code for the analysis (Imfeld, [2024a](#)).

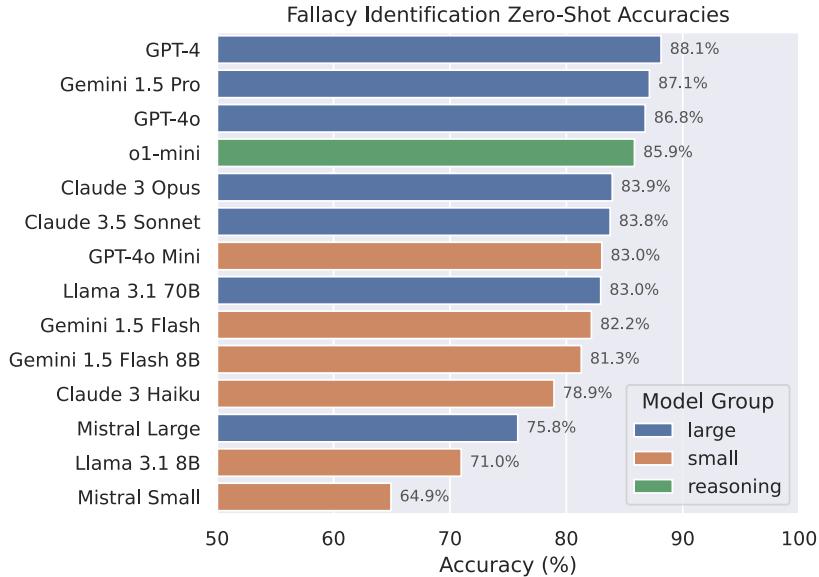
### 7.1 Fallacy Identification

First, zero-shot accuracies for the binary fallacy identification task described in Section [6.2](#) are presented, followed by an analysis of false alarms based on confusion metrics, and a comparison of different prompt methods.

#### 7.1.1 Zero-Shot Accuracies

Figure [9](#) illustrates that GPT-4 achieved the highest overall accuracy among the models tested. This finding aligns with the results reported by Hong et al. ([2024](#)). It is important to note that GPT-4 might have had a slight advantage because the FALLACIES dataset was constructed with guidance from GPT-4, which provided candidate fallacy examples to human experts. All large models demonstrated competitive accuracies ranging from 83% to 88%, showcasing the close competition among models developed by OpenAI, Anthropic, Google, and Meta. While Meta’s open-weight Llama 3.1 70B is closing in on the top proprietary models, Mistral’s models did not perform as well, with Mistral Large achieving an accuracy of 75.8%. Although Meta’s largest Llama model, with 405 billion parameters, was not included in this study, it is anticipated that it could set a new record for open-weight models on the FALLACIES dataset.

## 7 Results

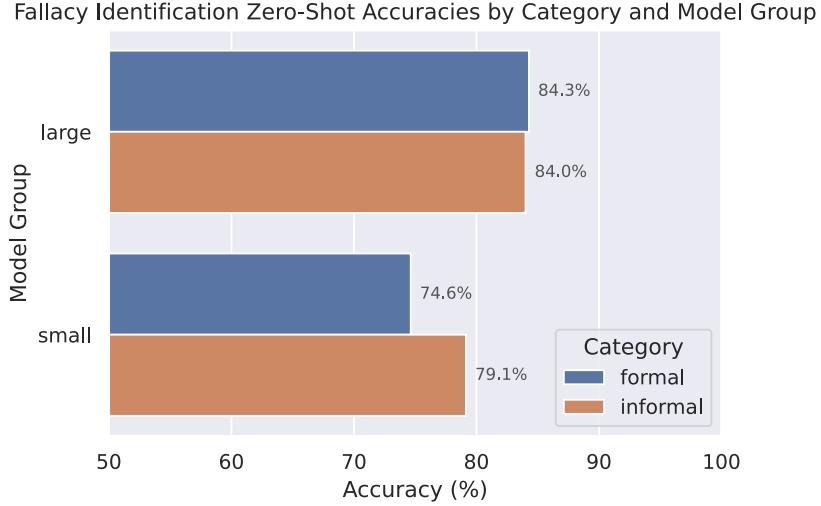


**Figure 9: Fallacy Identification Zero-Shot Accuracies.** Large models from OpenAI, Anthropic, Google, and Meta achieved accuracies between 83% and 88%, consistently outperforming smaller models. OpenAI’s o1-mini model with internal reasoning delivered a strong performance, foreshadowing the potential of upcoming reasoning models to set new benchmark records.

Interestingly, OpenAI’s cost-efficient o1-mini model achieved a very competitive accuracy of 85.9%. The more advanced o1-preview model was excluded due to its high cost, but it potentially could have set a new record for accuracy on the FALLACIES identification benchmark.

The model responses were then aggregated by model group (large vs. small models) and analyzed according to the fallacy category. Figure 10 demonstrates that large models performed similarly well on both formal and informal fallacies. In contrast, small models exhibited weaker performance specifically in formal fallacy identification. This particular shortcoming in formal reasoning aligns with the findings of Hong et al. (2024), who observed similar patterns when testing a variety of smaller models. The consistent pattern of large models outperforming smaller ones suggests that model size is a relevant factor for performance on the fallacy identification task.

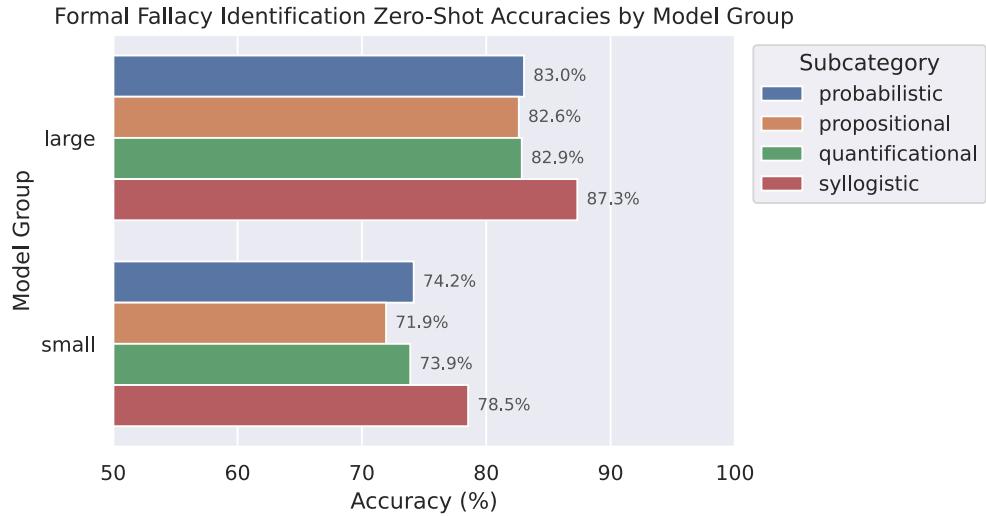
## 7 Results



**Figure 10: Fallacy Identification Zero-Shot Accuracies by Category and Model Group.** Large models demonstrated comparable performance across both categories, whereas small models struggled particularly with formal fallacy identification. This aligns with prior findings by Hong et al. (2024) on the limitations of smaller models in formal reasoning tasks.

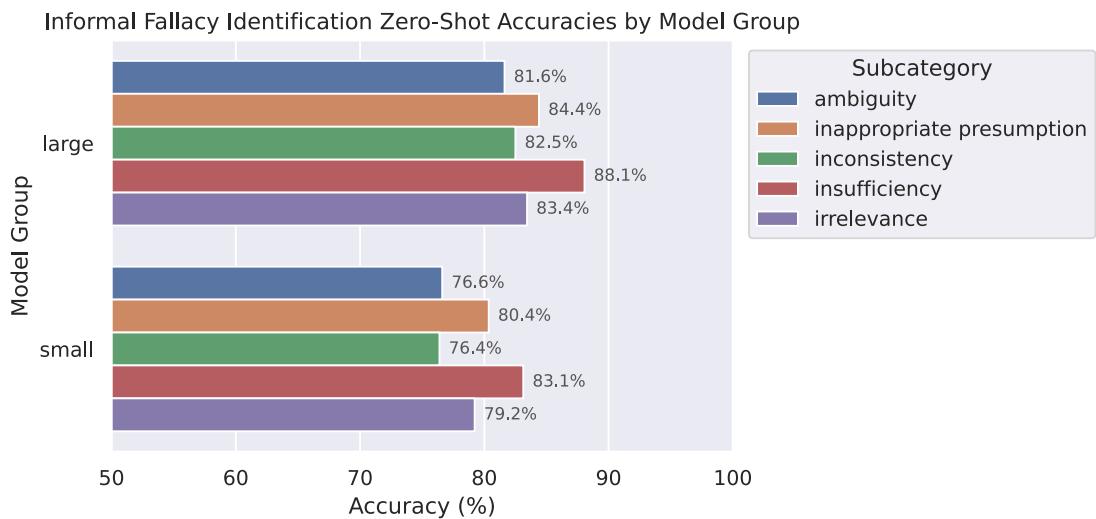
The model responses were further classified into subcategories of formal and informal fallacies. As depicted in Figure 11, large models consistently outperformed smaller models in all four formal subcategories. For both large and small models, the performance in identifying syllogistic fallacies was notably higher compared to the other subcategories, which exhibited similar levels of performance. This enhanced performance in syllogistic reasoning may be attributed to the abundance of syllogistic examples in the training data of LLMs, given the long-standing tradition of studying syllogisms in philosophy.

## 7 Results



**Figure 11: Formal Fallacy Identification Zero-Shot Accuracies by Model Group.** Large models consistently outperformed smaller ones in all four formal subcategories, with syllogistic fallacies showing the highest accuracy.

For the five informal subcategories, the accuracy pattern for large and small models shown in Figure 12 looks very similar, again with large models outperforming smaller ones in all subcategories. The highest accuracy was achieved in identifying insufficiency fallacies, where the premises do not adequately support the conclusion.



**Figure 12: Informal Fallacy Identification Zero-Shot Accuracies by Model Group.** In all five informal subcategories, large models outperformed smaller ones, with the highest accuracy observed in identifying insufficiency fallacies, where premises fail to adequately support the conclusion.

## 7 Results

Finally, Table 2 presents the achieved accuracies, ranking the models according to their overall performance. The table details the accuracies for each subcategory, which are then averaged to provide separate accuracies with respect to formal and informal fallacies. These two averages are combined to calculate the overall accuracy. This macro-average approach ensures equal weighting for all categories and subcategories, addressing the significant differences in the number of fallacy types and reasoning steps within each subcategory.

Model	Formal					Informal					Avg.	
	prop.	quant.	syl.	prob.	Avg.	amb.	incon.	irrel.	insuf.	inappr.		
GPT-4	<b>93.3</b>	84.2	88.8	86.2	<b>88.1</b>	84.7	85.0	<b>90.2</b>	<b>94.0</b>	86.9	<b>88.2</b>	<b>88.1</b>
Gemini 1.5 Pro	85.0	85.8	90.6	<b>88.8</b>	87.6	<b>85.7</b>	80.0	87.8	92.1	<b>88.1</b>	86.7	87.1
GPT-4o	83.3	80.8	90.0	87.5	85.4	85.0	<b>88.3</b>	87.9	92.4	87.0	88.1	86.8
o1-mini	88.3	82.5	90.6	82.5	86.0	78.3	85.0	89.4	90.3	85.5	85.7	85.9
Claude 3 Opus	85.0	<b>86.7</b>	83.8	83.8	84.8	82.0	83.3	79.6	86.6	84.0	83.1	83.9
Claude 3.5 Sonnet	86.7	80.8	<b>91.2</b>	81.2	85.0	78.7	81.7	81.2	86.9	84.3	82.5	83.8
GPT-4o Mini	80.0	74.2	84.4	77.5	79.0	84.3	86.7	86.3	91.1	87.0	87.1	83.0
Llama 3.1 70B	77.5	85.0	83.1	82.5	82.0	83.7	80.0	83.3	87.9	84.4	83.9	83.0
Gemini 1.5 Flash	81.7	81.7	86.9	76.2	81.6	80.0	81.7	80.7	86.6	84.4	82.7	82.2
Gemini 1.5 Flash 8B	77.5	<b>76.7</b>	85.0	76.2	78.9	81.7	81.7	82.5	88.1	84.6	83.7	81.3
Claude 3 Haiku	69.2	<b>74.2</b>	80.6	78.8	75.7	81.0	78.3	82.7	85.9	83.1	82.2	78.9
Mistral Large	67.5	<b>76.7</b>	83.8	71.2	74.8	74.0	78.3	76.2	78.6	77.2	76.9	75.8
Llama 3.1 8B	67.5	71.7	66.9	68.8	68.7	69.3	65.0	74.9	80.3	76.6	73.2	71.0
Mistral Small	55.8	65.0	67.5	67.5	64.0	63.3	65.0	68.2	66.7	66.3	65.9	64.9

**Table 2: Fallacy Identification Accuracies.** The four formal and five informal subcategory percent accuracies (rose) are averaged to compute the mean formal and informal accuracies (violet), which are further averaged to yield the overall accuracy (green). Consistent with the findings of the original FALLACIES dataset study (Hong et al., 2024), GPT-4 achieved the highest overall accuracy. However, the best subcategory accuracies (highlighted in bold) are distributed across different models and the results are much closer in the present study.

The accuracies observed for GPT-4 in this study match closely those previously reported, with variations within 2% at the subcategory level. The overall accuracy of 88.1% was slightly higher by 0.4%, probably not representing a significant difference. Notably, Gemini 1.5 Pro achieved an overall accuracy of 87.1%, placing it second by just a 1% margin. In three subcategories, Google’s flagship model outperformed others, as highlighted in bold in Table 2. Meanwhile, Anthropic’s Claude 3.5 Sonnet and Claude 3 Opus models trailed OpenAI’s and Google’s leading models by approximately 4%.

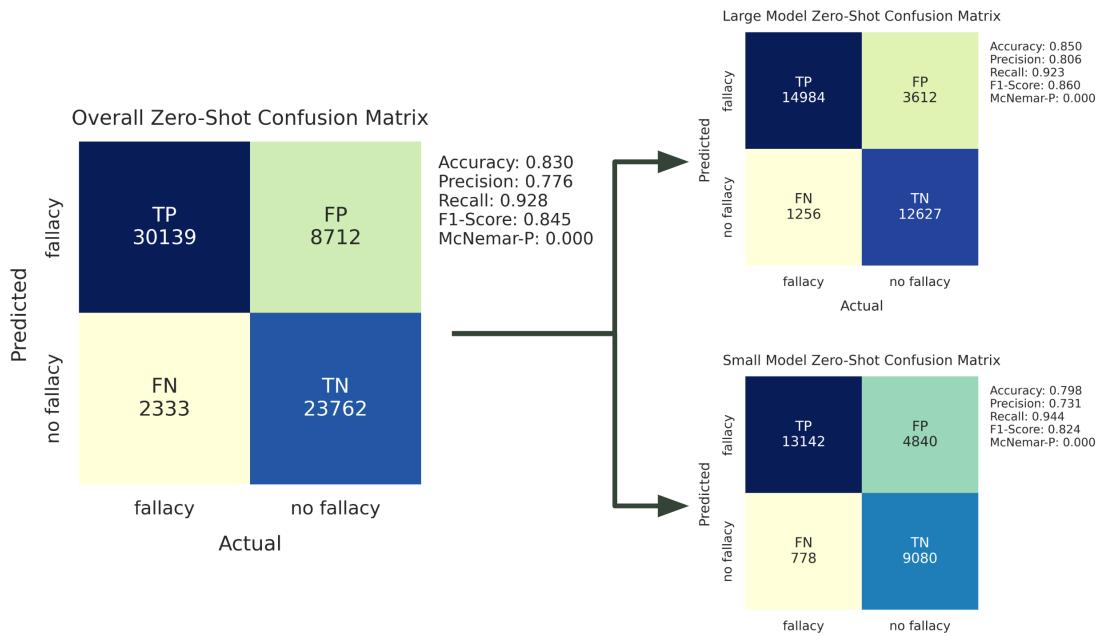
In conclusion, the results of the fallacy identification task demonstrate that the current flagship LLMs exhibit strong performance in detecting logical fallacies. The competitive accuracies observed across various OpenAI, Anthropic, Google, and Meta models high-

## 7 Results

light their effectiveness in complex reasoning tasks. The consistent outperformance of larger models over smaller ones underscores the importance of model size in handling intricate reasoning tasks such as fallacy detection.

### 7.1.2 False Alarms Bias

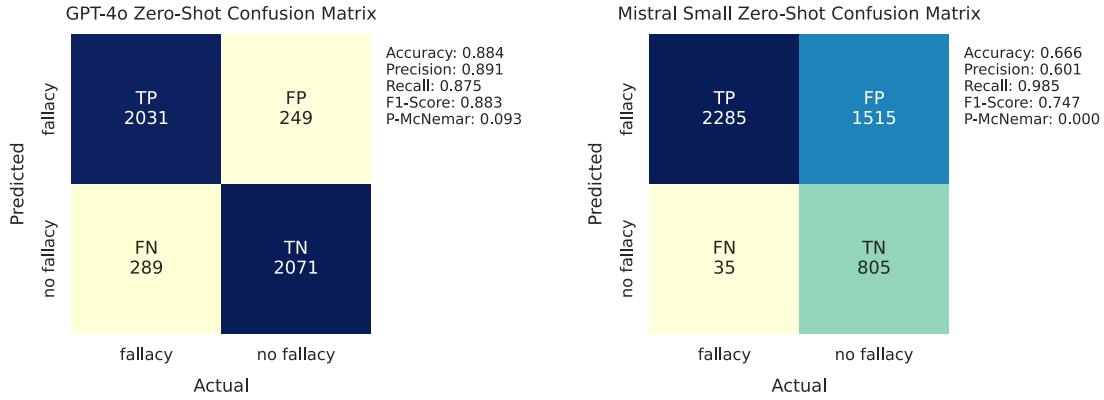
The analysis of confusion metrics, as detailed in Section 6.6, reveals a systematic false alarm bias in LLMs, illustrated in Figure 13. Specifically, LLMs have a propensity to generate more false positives than false negatives. This tendency is negligible in the most robust models, but becomes increasingly pronounced as model accuracy decreases.



**Figure 13: False Alarm Bias.** LLMs tend to generate false positives (FP), in other words they are likely to generate false alarms and judge valid reasoning as invalid. Large models generate about three times more false positives than false negatives on average (top right). Small models generate about six times more false positives than false negatives on average (bottom right).

## 7 Results

Figure 14 contrasts a balanced model (GPT-4o) with the model exhibiting the strongest false alarm bias (Mistral Small).



**Figure 14: Comparison of Zero-Shot Confusion Matrices for GPT-4o and Mistral Small.** This figure highlights the performance differences between a stronger and a weaker model. GPT-4o (left) demonstrates balanced performance without a bias towards false positives (FP), whereas Mistral Small (right) shows a significant inclination towards false positives ( $p < 0.001$ ). Note that the accuracies displayed in this figure differ from the macro-average overall accuracies in Table 2.

Table 3 ranks the models by accuracy and shows that the number of false positives increases as accuracy decreases, indicating that lower accuracy is linked to a false alarm bias. The bias can be quantified with the ratio of false positives to false negatives ( $FP/FN$ ), which is equivalent to the more widely used false positive rate to false negative rate ratio ( $FPR/FNR$ ) when class sizes are equal. OpenAI’s GPT-4o and o1-mini, as well as Google’s Gemini 1.5 Pro, are very balanced with  $FP/FN$  ratios close to 1.0. In contrast, Anthropic’s Claude models are about 5 to 6 times more likely to generate false positives than false negatives. At the lower end, Mistral Large ( $FP/FN = 13.1$ ) and Mistral Small ( $FP/FN = 43.3$ ) exhibit very strong false alarm biases.

Overall, the results reveal a clear false alarm bias in LLMs, as accuracy decreases. This bias is minimal in robust models like GPT-4o, but becomes significantly pronounced in less accurate models such as Mistral Small. The disparity in the  $FP/FN$  ratio underscores the need for careful consideration of this bias when evaluating and deploying LLMs in the context of logical fallacy detection, particularly for applications where false alarms could have critical implications.

## 7 Results

Model	TP	TN	FP	FN	FP/FN	Accuracy	Precision	Recall	F1
GPT-4	2017	2148	172	303	0.6***	0.898	0.921	0.869	0.895
Gemini 1.5 Pro	2090	2025	295	230	1.3**	0.887	0.876	0.901	0.888
GPT-4o	2031	2071	249	289	0.9	0.884	0.891	0.875	0.883
o1-mini	2013	2055	260	299	0.9	0.879	0.886	0.871	0.878
GPT-4o Mini	2086	1944	376	234	1.6***	0.869	0.847	0.899	0.872
Llama 3.1 70B	2202	1723	596	118	5.1***	0.846	0.787	0.949	0.860
Gemini 1.5 Flash 8B	2133	1766	554	187	3.0***	0.840	0.794	0.919	0.852
Claude 3.5 Sonnet	2209	1672	648	111	5.8***	0.836	0.773	0.952	0.853
Gemini 1.5 Flash	2159	1702	618	161	3.8***	0.832	0.777	0.931	0.847
Claude 3 Opus	2191	1665	655	129	5.1***	0.831	0.770	0.944	0.848
Claude 3 Haiku	2203	1635	685	117	5.9***	0.827	0.763	0.950	0.846
Mistral Large	2244	1323	997	76	13.1***	0.769	0.692	0.967	0.807
Llama 3.1 8B	2276	1228	1092	44	24.8***	0.755	0.676	0.981	0.800
Mistral Small	2285	805	1515	35	43.3***	0.666	0.601	0.985	0.747

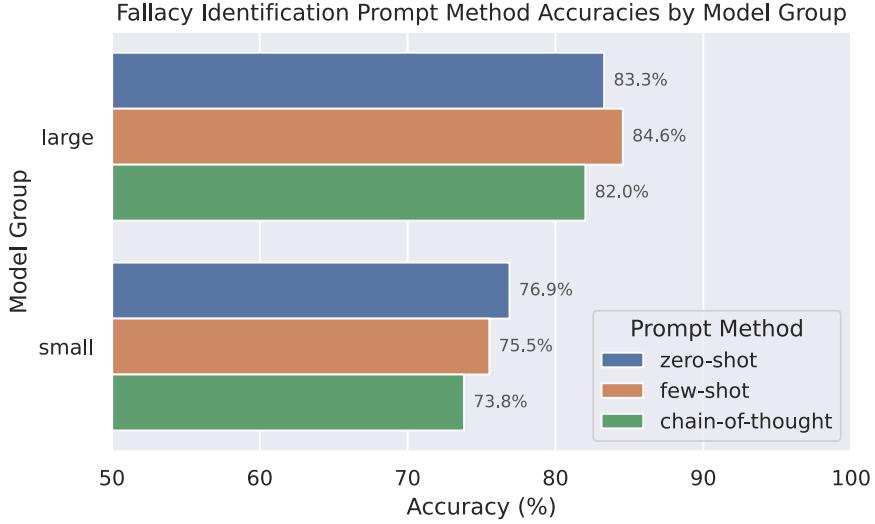
**Table 3: Fallacy Identification Confusion Metrics.** The lower the accuracy, the stronger the false alarm bias with false positives (FP) far exceeding false negatives (FN). The FP/FN ratio quantifies the false alarm bias. Significance is based on the McNemar test and levels are indicated by stars: \* for  $p \leq 0.05$ , \*\* for  $p \leq 0.01$ , and \*\*\* for  $p \leq 0.001$ . Accuracy is computed as the micro-average and therefore differs from the overall accuracy based on macro-averaging in Table 2.

### 7.1.3 Prompt Method Comparison

This section summarizes the experiments on fallacy identification using three prompt methods: zero-shot (experiment 1.1), few-shot (experiment 1.2), and chain-of-thought (CoT, experiment 1.3). These methods, detailed in Section 6.2 of the **Methods** chapter, were evaluated to determine their impact on model performance, particularly comparing small and large models. The results highlight individual model accuracies across different prompting techniques.

Overall, neither few-shot nor chain-of-thought prompts improved accuracy compared to zero-shot prompts. As depicted in Figure 15, the accuracies across the three methods were remarkably similar. Unlike earlier models, current LLMs may not benefit as much from response examples in the prompt or the explicit instruction to think step by step. Consistent with findings in the previous sections, larger models outperformed smaller ones across all prompting techniques.

## 7 Results



**Figure 15: Fallacy Identification Accuracy by Model Group and Prompt Method.** Large models consistently outperformed smaller models across all three prompt methods. However, few-shot and chain-of-thought prompting showed no significant accuracy improvement over zero-shot prompting.

Table 4 provides a detailed breakdown of the accuracy achieved by each model for the different prompt methods. Mistral Large frequently generated invalid responses to few-shot prompts, and was thus excluded from this analysis.

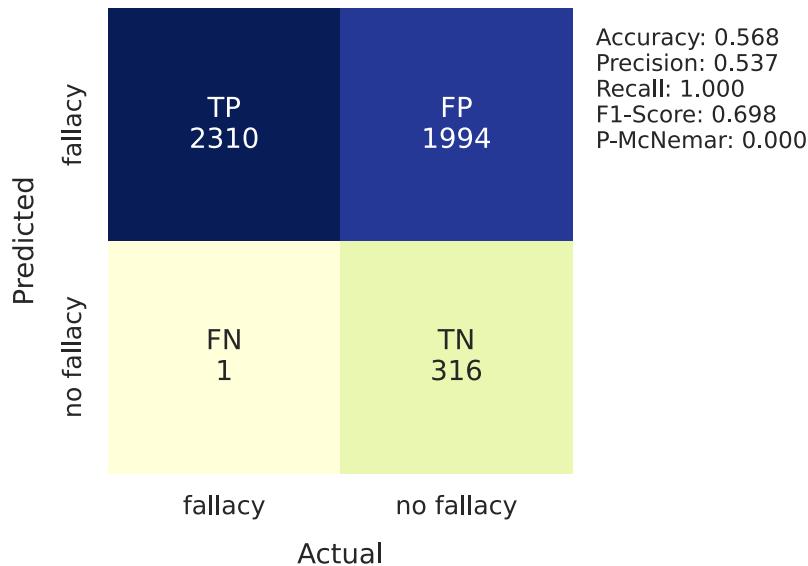
Model	Zero-Shot	Few-Shot	CoT
Gemini 1.5 Pro	<b>87.1</b>	<b>85.7</b>	82.8
GPT-4o	86.8	85.4	<b>86.4</b>
Claude 3.5 Sonnet	83.8	85.4	80.8
GPT-4o Mini	83.0	84.2	81.6
Llama 3.1 70B	83.0	81.7	80.5
Gemini 1.5 Flash	82.2	72.7	79.8
Gemini 1.5 Flash 8B	81.3	77.5	77.7
Claude 3 Haiku	78.9	71.2	79.9
Mistral Large	75.8	-	79.6
Llama 3.1 8B	71.0	75.4	52.7
Mistral Small	64.9	72.1	71.1

**Table 4: Fallacy Identification Accuracies Across Prompt Methods and Models.** OpenAI, Anthropic, Google, and Meta’s flagship models consistently achieved over 80% accuracy across all three prompt methods. In contrast, Mistral Large struggled with few-shot prompts, often producing invalid responses. Llama 3.1 8B achieved only 52.7% accuracy with chain-of-thought (CoT) prompts, due to a large false positive rate.

## 7 Results

The Llama 3.1 8B model exhibited limited reasoning capabilities when using chain-of-thought prompts, as evidenced by its confusion matrix in Figure 16. With an overwhelming number of false positives (1994) compared to just one false negative, the model demonstrated an extreme false alarm bias.

Llama 3.1 8B Chain-of-Thought Confusion Matrix



**Figure 16: Llama 3.1 8B Chain-of-Thought Confusion Matrix.** The small Llama 3.1 8B model consistently ignored the chain-of-thought instruction to think before giving an answer, and denied almost all valid reasoning.

Llama 3.1 8B consistently disregarded instructions to think step-by-step before providing a final answer, as showcased with the following response example:

---

Chain-of-thought prompt with valid reasoning step:

"Is the following reasoning step correct?  
 Let's think step by step and then answer "Yes" or "No".  
 Since I claim that there is a magical genie living inside this cocoon, it is my responsibility to provide evidence or proof for my claim, therefore, without sufficient evidence, we cannot conclusively say there is indeed a magical genie living inside this cocoon."

Llama 3.1 8B response:

"No. The reasoning step is incorrect because it assumes that the claimant (you) has the burden of proof to prove the existence of a magical genie. However, in the

## 7 Results

absence of any prior evidence or context suggesting the existence of magical genies, it's more accurate to say that the claim itself is extraordinary and requires evidence to support it. In other words, the default assumption should be that there is no magical genie living inside the cocoon, and it's up to you to provide evidence to challenge that assumption. This is a common principle in scientific inquiry and critical thinking, known as the burden of proof or the null hypothesis."

---

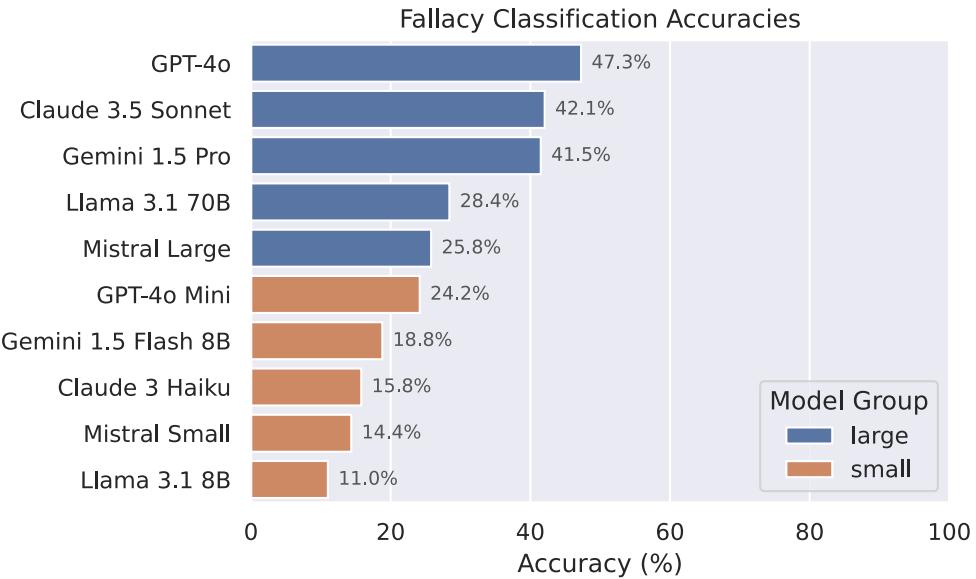
### 7.2 Fallacy Classification

This section analyzes the performance of models in classifying logical fallacies into a comprehensive set of types. The classification task described in Section 6.3 is considerably more complex than the binary identification task, since it requires distinguishing among 232 distinct fallacy types from the FALLACIES dataset. The complexity is reflected in lower accuracy rates and a significantly reduced chance of random guessing success. While random guessing yields about 50% accuracy in binary identification, it drops to merely 0.4% when classifying 232 fallacy types.

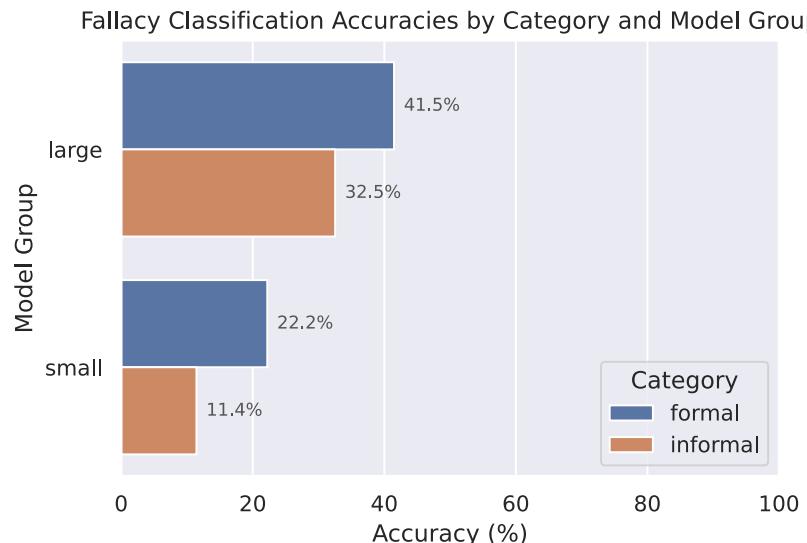
Figure 17 illustrates that although overall accuracies are lower than in the identification task, GPT-4o, Claude 3.5 Sonnet, and Gemini 1.5 Pro achieved over 40% accuracy, demonstrating their relative proficiency in handling such a difficult task. The clear performance gap between large and small models underscores the utility of the fallacy classification task as a benchmark for complex reasoning capabilities.

As shown in Figure 18, classification accuracies varied significantly between formal and informal fallacies. Contrary to the identification task where small models underperformed with formal fallacies, both small and large models performed better for formal than informal fallacy type classification. This reversal aligns with the findings of Hong et al. (2024). A potential explanation not mentioned by the authors is that formal fallacies, with only 24 types, might generally be easier for models to classify accurately compared to the 208 informal types. As the number of fallacy types within a category increases, the semantic overlap among the types also increases, complicating the classification within the informal category.

## 7 Results



**Figure 17: Fallacy Classification Accuracies.** The classification task challenges models to assign one of 232 fallacy types instead of a binary answer. Although accuracies were lower than for the identification task, some large models still exceeded 40% and surpassed GPT-4's accuracy of 35.0% reported by Hong et al. (2024).



**Figure 18: Fallacy Classification Accuracies by Category and Model Group.** Classification accuracies were higher for formal fallacies compared to informal ones, reflecting the greater difficulty of distinguishing among 208 informal types versus 24 formal types. Large models substantially outperformed smaller models across both categories.

## 7 Results

Table 5 provides a detailed breakdown of model performance in both formal and informal fallacy subcategories. GPT-4o emerged as the top performer with an overall accuracy of 47.3%, significantly outperforming its predecessor GPT-4, which was benchmarked at 35.0% by Hong et al. (2024). Claude 3.5 Sonnet and Gemini 1.5 Pro also demonstrated strong performances, with accuracies of 42.1% and 41.5%, respectively. Smaller models like Llama 3.1 8B and Mistral Small failed to achieve high accuracies, highlighting the advantage of larger model architectures in this task.

Model	Formal					Informal					Avg.	
	prop.	quant.	syl.	prob.	Avg.	amb.	incon.	irrel.	insuf.	inappr.		
GPT-4o	65.0	<b>50.0</b>	38.8	57.5	<b>52.8</b>	44.0	<b>46.7</b>	42.1	<b>36.0</b>	<b>39.8</b>	<b>41.7</b>	<b>47.3</b>
Claude 3.5 Sonnet	<b>75.0</b>	26.7	<b>47.5</b>	37.5	46.7	42.0	40.0	<b>43.6</b>	33.4	28.1	37.4	42.1
Gemini 1.5 Pro	55.0	31.7	25.0	<b>67.5</b>	44.8	<b>46.0</b>	40.0	40.6	33.6	30.7	38.2	41.5
Llama 3.1 70B	55.0	3.3	12.5	62.5	33.3	28.7	16.7	26.8	28.6	16.7	23.5	28.4
Mistral Large	46.7	0.0	17.5	55.0	29.8	27.3	6.7	31.4	24.0	19.6	21.8	25.8
GPT-4o Mini	48.3	1.7	22.5	60.0	33.1	19.3	3.3	24.1	17.9	11.3	15.2	24.2
Gemini 1.5 Flash 8B	45.0	15.0	16.2	27.5	<b>25.9</b>	18.0	0.0	13.6	11.2	15.6	11.7	18.8
Claude 3 Haiku	36.7	0.0	7.5	20.0	16.0	15.3	<b>16.7</b>	17.7	16.9	11.1	15.5	15.8
Mistral Small	30.0	3.3	7.5	37.5	19.6	4.7	3.3	17.7	12.6	7.4	9.1	14.4
Llama 3.1 8B	26.7	0.0	8.8	30.0	<b>16.4</b>	8.0	0.0	10.3	6.4	3.7	5.7	11.0

**Table 5: Fallacy Classification Accuracies.** The table shows the average accuracies for formal and informal subcategories (in rose), which are further averaged to yield overall accuracy (in green). GPT-4o (47.3%) outperformed GPT-4 (35%) benchmarked by Hong et al. (2024) by a substantial margin.

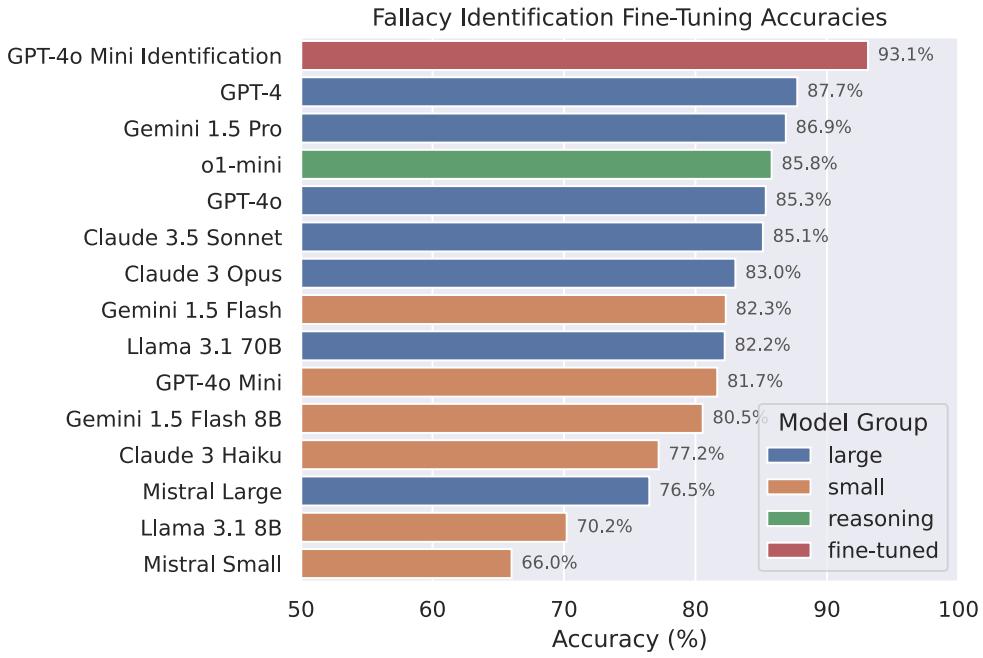
These results highlight rapid advancements in LLM capabilities since all models tested by Hong et al. (2024) achieved classification accuracies below 10%, except for Flan-T5-xxl (14.2%), GPT-3 (17.3%), and GPT-4 (35.0%). Even the weakest model tested here (Llama 3.1 8B) performed with an accuracy of 11%.

### 7.3 Fine-Tuning Performance

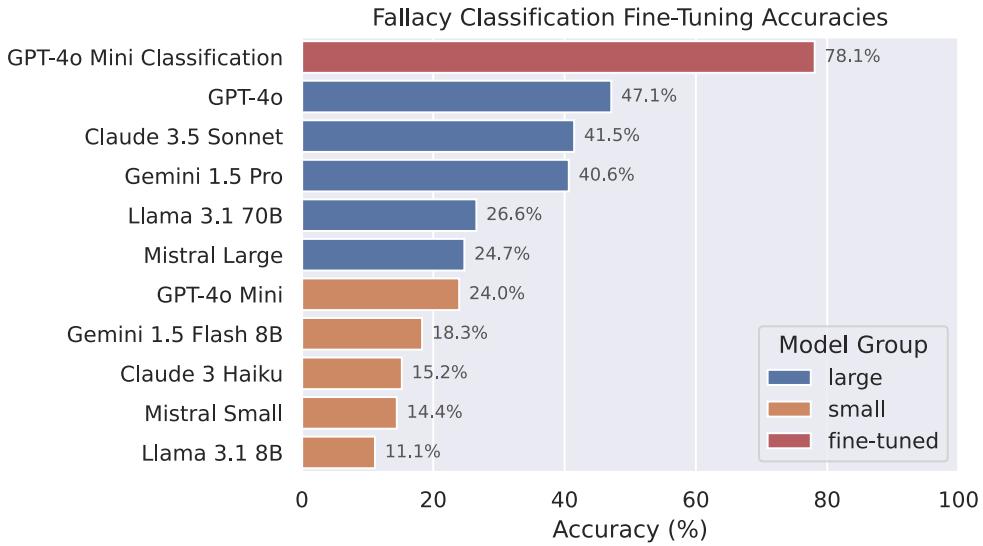
This section presents the results of the fine-tuning experiments 1.4 and 2.2, as detailed in Sections 6.2.4 and 6.3.2, respectively. These experiments involved fine-tuning a GPT-4o Mini model for the identification and classification tasks.

Figure 19 shows that the fine-tuned identification model outperformed all other models with an accuracy of 93.1%, an improvement of 11.4% over the standard GPT-4o Mini model. Similarly, the fine-tuned classification model achieved an accuracy of 78.1%, a substantial increase of 54.1%, as shown in Figure 20.

## 7 Results



**Figure 19: Fallacy Identification Fine-Tuning Accuracies.** The model fine-tuned on the fallacy identification task achieved the highest accuracy of 93.1%, representing an 11.4% improvement over the baseline GPT-4o Mini model. However, these gains may be attributed to overfitting to task-specific patterns in the dataset rather than genuine improvements in reasoning abilities.



**Figure 20: Fallacy Classification Fine-Tuning Accuracies.** The model fine-tuned on the fallacy classification task achieved an accuracy of 78.1%, marking a significant improvement of 54.1% over baseline performance. This large performance gain probably results from overfitting to task-specific patterns.

## 7 Results

Despite these impressive gains, there is concern that these improvements may primarily reflect overfitting to the specific tasks within the FALLACIES dataset rather than a true enhancement in reasoning capabilities. Although accuracies were evaluated on an unseen test set during fine-tuning, the model might have benefited from the close alignment between reasoning steps and fallacy types in the dataset. Additional out-of-distribution (OOD) evaluation results point in that direction, indicating that fine-tuning degraded performance in different but related contexts, see [Appendix C](#).

### 7.4 Fallacy Type Performance

The analysis of fallacy type accuracies highlights the strengths and weaknesses of LLMs in identifying and classifying certain types of fallacies. This involves determining whether the reasoning is valid and, if not, classifying the specific type of fallacy present. The responses of all the tested models were pooled and grouped by fallacy type, allowing the calculation of confusion metrics for each type, including accuracy. Table 6 presents the metrics achieved in the fallacy identification task with zero-shot prompts (experiment 1.1 described in Section [6.2.1](#)), focusing on the five top and bottom fallacies ranked by accuracy.

Fallacy	TP	TN	FP	FN	Accuracy
Historians Fallacy	140	138	2	0	0.993
Appeal to Complexity	140	136	4	0	0.986
Confusing Currently Unexplained ...	140	133	7	0	0.975
Texas Sharpshooter Fallacy	140	129	11	0	0.961
Argument from Incredulity	140	127	13	0	0.954
...	...	...	...	...	...
Existential Fallacy	42	109	31	98	0.539
Argument by Repetition	70	78	62	70	0.529
Illicit Substitution of Identicals	36	110	30	104	0.521
Stereotyping the fallacy	33	110	30	107	0.511
Having Your Cake	29	106	34	111	0.482

**Table 6: Fallacies Ranked by Accuracy.** Distinct performance patterns are revealed by ranking the 232 fallacy types from highest to lowest accuracy based on aggregated LLM fallacy identification responses (top and bottom 5 are shown). Overall, LLMs performed best for fallacies which tend to be common, have distinctive and memorable names, or are less complex. In contrast, the LLMs struggled with fallacies which involve complex formal logic, overlap with other categories, are uncommon, context-dependent, or abstract and philosophical.

## 7 Results

Identifying common patterns among the diverse set of fallacy types proved challenging, given the large number of fallacy types. However, GPT-4o, Gemini 1.5 Pro, and Claude 3.5 Sonnet were able to discern similar trends when prompted for an exploratory analysis of the accuracy table. Common fallacies, such as [Ad Hominem](#), [Slippery Slope](#), [Hasty Generalization](#), [Straw Man Fallacy](#), and [Cherry Picking](#), tend to be associated with higher accuracy rates. Distinctive and memorable fallacy names, such as [Texas Sharpshooter Fallacy](#) and [Straw Man Fallacy](#), also seem to aid in their identification. In addition, less complex fallacies, such as [False Dilemma](#) or [Appeal to Intuition](#), also show higher accuracy.

In contrast, models typically struggled with fallacies characterized by complex formal logic. Examples include [Illicit Substitution of Identicals](#), [Modal Scope Fallacy](#), [Existential Fallacy](#), [Limited Scope Fallacy](#), and [Denying the Antecedent](#). Fallacies that overlap with others, such as the [Ambiguity Fallacy](#) which shares similarities with other vague language fallacies, also pose challenges. Furthermore, uncommon fallacies such as [Having Your Cake Fallacy](#) or [Subjectivist Fallacy](#) are harder for models to identify accurately. Context-dependent fallacies, such as [Stereotyping Fallacy](#), add another layer of difficulty. Lastly, abstract or philosophical fallacies like the [Reification Fallacy](#), [Existential Fallacy](#), and [Modal Scope Fallacy](#) are particularly challenging for these models to identify correctly.

While the confusion metrics for the binary identification task were calculated using a  $2 \times 2$  confusion matrix, the classification performance analysis used a  $232 \times 232$  confusion matrix, based on the aggregated responses of all LLMs tested. From this matrix, the two most common misclassifications for each of the 232 fallacy types were identified. The fallacy types were then ordered by the number of most common misclassifications. The ten most frequent misclassifications are presented in Table 7. To highlight the issue of semantic overlap due to the large number of fallacy types, fallacies related to "Authority" are printed in bold. The models struggled to differentiate between similar fallacies such as [Appeal to Authority](#), [Argument from False Authority](#), [Appeal to False Authority](#), and [Appeal to Trust](#). Although other semantic clusters exist in the FALLACIES dataset, this is the most prominent.

## 7 Results

Fallacy Type	TP	Misclassification 1	Misclassification 2
Negating Antecedent and Consequent	0	Denying the Antecedent (87)	Affirming the Consequent (7)
<b>Appeal to Trust</b>	0	<b>Appeal to Authority</b> (80)	<b>Argument from False Authority</b> (8)
Argument to the Purse	0	<b>Appeal to Authority</b> (80)	<b>Appeal to False Authority</b> (6)
<b>Blind Authority Fallacy</b>	0	<b>Appeal to Authority</b> (79)	<b>Argument from False Authority</b> (12)
Righteousness Fallacy	0	<b>Appeal to Authority</b> (65)	Affirming the Consequent (9)
Limited Scope	0	Circular Reasoning (46)	Affirming the Consequent (18)
<b>Anonymous Authority</b>	0	Appeal to Popularity (34)	<b>Appeal to Authority</b> (15)
Spin Doctoring	0	Hasty Generalization (24)	Fallacy of Composition (11)
Unwarranted Contrast	0	False Conversion (22)	Denying the Antecedent (18)
Use Mention Error	0	Fallacy of Composition (22)	Equivocation (19)

**Table 7: Common Fallacy Misclassifications.** The most frequently misclassified fallacies were never classified correctly ( $TP = 0$ ). The most and second most frequent misclassifications are shown in the Misclassification 1 and 2 columns with the number of misclassifications in parentheses. Fallacy types related to authority are shown in bold font, highlighting the large semantic overlap between certain types in the FALLACIES dataset. The large number of fallacy types (232) makes the classification task very challenging.

In summary, the analysis of fallacy type performance revealed distinct patterns in how LLMs identify and classify logical fallacies. Models distinguished valid from invalid arguments better for common and less complex fallacy types, often benefiting from distinctive and memorable names. However, they had difficulty with fallacies involving complex formal logic, semantic overlap, and those that are context-dependent or abstract. Misclassification analysis underscores the difficulty posed by semantic overlap between certain fallacy types, particularly those related to authority.

### 7.5 Fallacy Search

The fallacy search task was carried out using the MAFALDA dataset as detailed in Section 5.2. Although the MAFALDA dataset contains fewer text samples (200 compared to 4640 in the FALLACIES dataset), it offers a richer annotation scheme. Each sample may include multiple overlapping annotated text spans, unlike the FALLACIES dataset which provides only a label indicating correctness and fallacy type. This comprehensive annotation allows for multiple correct interpretations within the gold standard, encoding more nuanced information per text. Consequently, calculating precision and recall metrics is more complex, as explained in Section 6.6.3.

Following several iterations of prompt engineering, the results presented in Table 8 were obtained. In a previous study by Helwe et al. (2024), GPT-3.5 achieved an F1-score of 0.138 at level 2, which was surpassed by a small group of four human study

## 7 Results

Level	All			Subset		
	0	1	2	0	1	2
GPT-4o	0.595	0.519	0.442	0.744	0.633	0.521
GPT-4o Mini	0.587	0.491	0.374	0.733	0.592	0.422

**Table 8: MAFALDA F1-Scores for Fallacy Search.** The left section displays F1-scores based on all 200 annotated texts across three hierarchy levels, while the right section excludes texts without annotations, focusing on the subset of 137 texts with at least one fallacy annotation.

participants who scored 0.186 on 20 randomly selected texts from the MAFALDA dataset. This study found that GPT-4o achieved a significantly higher F1-score of 0.442 at level 2, outperforming GPT-4o Mini which scored 0.374.

However, the performance evaluation of Helwe et al. (2024) may have been affected by a methodological issue discussed in Section 6.6.3. An alternative analysis incorporating additional labels for uncovered text spans allows for a more direct comparison with their results, showing GPT-4o achieving a level 2 F1-score of 0.264 (Appendix D).

Given that the MAFALDA dataset includes 63 texts without annotations in the gold standard, additional F1-scores were computed exclusively for the subset of 137 texts with at least one annotation. As noted in Section 7.1.2, LLMs tend to generate false alarms, which are severely penalized in non-fallacious texts. If an LLM identifies a fallacy in a text span where none exists according to the gold standard, both precision and the F1-score drop to zero. By refining prompt instructions to reduce false alarms, such as replacing "Be very thorough and include all potential fallacies" with "Apply the principle of charity, consider the argument in its strongest form, and avoid over-detection", the F1-scores were improved.

Despite using a single prompt per text rather than prompting each sentence individually as described in Section 6.4, the more advanced LLMs used in this study surpassed previous results reported by Helwe et al. (2024). This approach is also more scalable for longer texts, allowing for efficient analysis of entire newspaper articles without separate prompts for each sentence. Performance evaluation on the MAFALDA benchmark can be completed quickly and cost-effectively, enabling rapid iterations with different system prompts. Taken together, these features establish a solid foundation for the web application discussed in Section 8.

## 8 Practical Implementation

This chapter addresses the second part of the third objective outlined in Section 1.2, transitioning from basic research to a real-world application by implementing the Fallacy Search web application. Previous chapters demonstrated that current leading LLMs can effectively distinguish between valid arguments and logical fallacies, achieving high accuracy in classification. This chapter focuses on the practical application of these findings.

The application is hosted on <https://fallacy-search.streamlit.app> and was developed using the Streamlit framework (“Streamlit”, 2021). In the backend, requests are sent to OpenAI’s GPT-4o model, requesting structured output (“OpenAI Platform”, n.d.), using the LangChain framework (Chase, 2022). The source code is available on GitHub (Imfeld, 2024b).

### 8.1 From Research to Application

The fallacy search experiment described in Section 6.4 was designed with practical application in mind, requiring only minor modifications. Initially, detected fallacies were limited to the 23 types used in the MAFALDA dataset for performance measurement. However, to enhance versatility, the system prompt now instructs the model to consider any formal or informal logical fallacy without an explicit list. The model also provides one-sentence definitions of identified fallacies, an overall summary, and a logical reasoning quality rating on a scale from 1 to 10. If no arguments are present, the rating is omitted. The close parity between the experimental and application system prompt allows for evaluating prompt modifications based on F1-score improvements.

### 8.2 Fallacy Search Web Application User Interface

The application features a simple and mobile-friendly user interface interface, shown in Figure 21. Users can input texts of substantial length, such as newspaper articles or blog posts. The application outputs a list of identified fallacies followed by an overall analysis and reasoning score. The analysis considers both fallacies and their defenses

and concludes with a disclaimer about the subjective nature of fallacy interpretation. Additional information is available via an "About" page.

## Fallacy Search

Enter text to detect logical fallacies and get a reasoning score:

Even if the planets are spheres, how does that prove the earth is a globe? That's like saying the lights on your ceiling are bulbs, therefore the floor is a bulb.

Analyze

### Detected Fallacies

⌚ Fallacy #1: False Analogy

**False Analogy:** A false analogy occurs when an argument is based on misleading, superficial, or implausible comparisons.

**Quote:** "*That's like saying the lights on your ceiling are bulbs, therefore the floor is a bulb.*"

**Reason:** The analogy compares the shape of planets to the shape of the Earth, suggesting that just because other planets are spheres, it doesn't necessarily mean the Earth is a sphere. However, the comparison to ceiling lights and the floor is misleading because the relationship between planets and the Earth is not analogous to the relationship between ceiling lights and the floor. Planets and the Earth are both celestial bodies subject to similar physical laws, whereas ceiling lights and floors are not comparable in this way.

**Defense:** The argument could be interpreted as emphasizing the need for direct evidence of Earth's shape rather than relying on analogies with other celestial bodies. It suggests caution in assuming Earth's shape based solely on the shape of other planets.

**Confidence:** 90%

**Figure 21: Fallacy Search Web Application Interface.** In response to prominent conspiracy theorist Alex Jones, who wondered why flat Earth believers can't just "look at other planets and see that they are globes as well," the following argument was put forward: "Even if the planets are spheres, how does that prove the earth is a globe? That's like saying the lights on your ceiling are bulbs, therefore the floor is a bulb." (Witsit [@WitsitGetsIt], 2024). The flawed reasoning was correctly identified as a false analogy with high confidence. According to Poe's law, without a clear indicator of the author's intent, we may never know if this was intended as a serious argument.

## 8.3 Use Cases

The following sections discuss various use cases to illustrate the application's capabilities.

### 8.3.1 Basic Logical Reasoning

The first use case is shown in Figure 22 and tests simple logical reasoning by applying an if-then rule according to [Modus Ponens](#). To make the test a little more difficult, the if-part (antecedent) is stated *after* the then-part (consequent): "Sarah will go outside if it's sunny today. Today it is sunny, so Sarah will go outside".

Sarah will go outside if it's sunny today. Today it is sunny, so Sarah will go outside.

Analyze

## Overall Analysis

### Summary

The reasoning in the text is logically sound, as it follows a valid modus ponens structure: If P, then Q; P is true, therefore Q is true. There are no logical fallacies present.

### Reasoning Score

 (10 out of 10)

**Figure 22: Fallacy Search Demonstrating Logical Reasoning.** The application correctly identifies the argument as valid reasoning based on modus ponens.

The model initially misclassified the argument as an [Affirming the Consequent](#) logical fallacy with the following justification: "The statement assumes that because it is sunny, Sarah will definitely go outside. However, the original condition only states that if it is sunny, Sarah will go outside, not that sunny weather is the only condition for her going outside." Although the distinction of necessary and sufficient conditions is important in other contexts, an [Affirming the Consequent](#) fallacy in this case would be the following: "Sarah is going outside today, therefore it is sunny." The false alarm was fixed by adding

the following guidelines to the system prompt:

- Consider principles of formal logical reasoning when judging the validity of an argument
- For formal logical arguments, accept premises as true for the sake of the argument

Basic logical reasoning seems to work quite often, as evidenced by the high formal reasoning accuracy scores demonstrated in Section 7.1. However, high accuracy scores on benchmark tests do not prove logical reasoning on a deeper level. This example underscores both the model’s adaptability to changes in the system prompt and its occasional brittleness in logical reasoning.

Experimentation with the following argument highlights the fragility of logical reasoning: “I’m friends with everyone in the book club, and Sarah is part of the book club, so obviously I’m friends with Sarah.” This is a valid deductive argument using universal instantiation, where an inference is made from a general truth about all members of a group to a specific member of that group. However, the current version of the Fallacy Search application identifies a [Begging the Question](#) fallacy, explaining that “The statement assumes the conclusion (being friends with Sarah) in the premise (being friends with everyone in the book club), without providing independent evidence for the conclusion.” When OpenAI’s current version of ChatGPT is repeatedly prompted to “Detect any formal or informal logical fallacy in the following text: I’m friends with ...”, it detects fallacies such as [Hasty Generalization](#), [Fallacy of Composition](#), or [Fallacy of Division](#). Yet, when asked “Is this a valid argument? I’m friends with ...”, ChatGPT asserts that “This argument is valid. It employs a form of logical reasoning known as a categorical syllogism ...”. This illustrates how rephrasing prompts can lead LLMs to inconsistent logical conclusions.

### 8.3.2 Analyzing a Blog Post

In another use case, the application analyzed a blog post by Deepak Chopra titled “Why Your Brain Isn’t Experiencing Anything” (Chopra, 2024) and identified several logical fallacies such as a [False Dichotomy](#) and [Begging the Question](#). The analysis in Figure 23 showcases the ability of the application to handle longer and more complex texts.

## 8 Practical Implementation

The screenshot shows a user interface for the Fallacy Search application. It displays a list of identified fallacies with their descriptions and analysis. The fallacies listed are:

- Fallacy #1: False Dichotomy** (closed)
- Fallacy #2: Begging the Question** (open)
  - Begging the Question:** Begging the question is a logical fallacy where the conclusion is assumed in one of the premises.
  - Quote:** "The complexity of the human nervous system is fascinating, but the mind doesn't need the brain in order to exist. Consciousness preceded all living things by being fundamental in creation."
  - Reason:** The statement assumes that consciousness is fundamental and does not require the brain, which is the conclusion the author is trying to prove. This assumption is not supported by evidence within the text.
  - Defense:** The author might be referencing philosophical or spiritual beliefs that posit consciousness as a fundamental aspect of the universe, which are not meant to be proven but rather accepted as axiomatic within those frameworks.
  - Confidence:** 70%
- Fallacy #3: Appeal to Authority** (closed)
- Fallacy #4: Non Sequitur** (closed)

**Figure 23: Fallacy Search Analysis of Arguments Against Physicalism.** Deepak Chopra argues that "the mind doesn't need the brain in order to exist" (Chopra, 2024) by simply asserting that "consciousness preceded all living things by being fundamental in creation." Since the text does not contain any evidence to support the claim, the Fallacy Search analysis justifiably classifies the argument as a begging the question fallacy.

A thorough examination of the blog article or the topic of physicalism within the philosophy of mind is beyond the scope of this discussion. The primary aim here is to illustrate how the Fallacy Search application can assist in analyzing longer and more intricate texts. By providing a defense for each identified fallacy, the application supports critical thinking while adhering to the principle of charity.

### 8.3.3 Detecting Fallacies in a Political Debate

The third use case involves an analysis of the presidential debate of 2024 between Biden and Trump (The Wall Street Journal, 2024). A transcript of the debate was extracted using a third-party tool, and a portion was analyzed with the Fallacy Search application. As illustrated in Figure 24, various logical fallacies were identified, including a notable

instance in which Trump used a **Straw Man Fallacy**. He misrepresented the stance of abortion supporters by asserting that "they will take the life of a child in the eighth month, the ninth month, and even after birth." This analysis accurately highlights that such claims do not reflect the views held by pro-choice advocates.

 Fallacy #2: Straw Man

**Straw Man:** A straw man fallacy misrepresents an opponent's argument to make it easier to attack.

**Quote:** "*they will take the life of a child in the eighth month the ninth month and even after birth*"

**Reason:** The statement misrepresents the position of those who support abortion rights by suggesting they advocate for infanticide, which is not a position held by mainstream pro-choice advocates.

**Defense:** The speaker might be referring to extreme cases or misinterpretations of late-term abortion laws, though this is not representative of the general pro-choice stance.

**Confidence:** 90%

**Figure 24: Fallacy Search Detection of Fallacies in Presidential Debate.** Fallacy Search detected multiple logical fallacies in the 2024 presidential debate between Trump and Biden, including Trump's use of a Straw Man Fallacy to misrepresent pro-choice positions.

Similar propaganda and misinformation is spread on social media faster than it can be corrected. The example illustrates that the application can be used to identify and address flawed arguments in debates with substantial real-world impact.

## 8.4 Real-World Implications

The Fallacy Search web application showcases the utility of cutting-edge language models for practical tasks, bridging the gap between academic research and real-world use cases. By identifying logical fallacies in diverse contexts, ranging from simple arguments to complex texts like blog posts and political debates, the application demonstrates its potential to support critical thinking and argument analysis.

While the performance of the underlying LLM in logical reasoning is remarkable, occasional misclassifications highlight the limitations of current AI systems in consistently handling nuanced arguments. These challenges underscore the need for even more powerful models with greater complex reasoning skills and reliability.

Looking ahead, expanding the capabilities of the Fallacy Search application could further improve its utility as a tool for educators, journalists, researchers, and anyone navigating the complexities of argumentation in today's information-dense environment.

## 9 Discussion

The following concluding sections of this thesis highlight the research context and significance, summarize and interpret key findings, discuss the limitations of the study, and suggest future research directions.

### 9.1 Context and Significance

In the contemporary digital landscape, the proliferation of misinformation and manipulative content poses significant challenges. Social media platforms, in particular, are rife with pseudoscience, motivated reasoning, and polarized political statements. The advent of generative AI technologies, such as large language models (LLMs), has further complicated this issue by enabling the creation of highly persuasive and contextually tailored messages. Detecting logical fallacies in such content is crucial for mitigating the spread of misinformation, improving the quality of public discourse, and promoting critical thinking. LLMs offer a promising solution because of their ability to process large amounts of text and identify subtle patterns indicative of fallacious reasoning. By automating the detection of logical fallacies, LLMs can assist in various domains, including journalism, education, and social media monitoring, thereby contributing to more informed decision-making and public awareness.

### 9.2 Summary of Key Findings

In order to address the objectives of this thesis and answer the research questions stated in Chapter 2, the key findings in the [Results](#) chapter can be summarized as follows:

#### 1. Fallacy Identification:

- a) **Zero-Shot Accuracies:** The study found that the current flagship models of OpenAI, Anthropic, and Google perform very well on the fallacy identification task with accuracies in a close range between 83% and 88%. Open source models by Meta are catching up, while Mistral AI's models are trailing behind. GPT-4 achieved the highest accuracy of 88.1%, replicating the findings in

## 9 Discussion

Hong et al. (2024). Large models consistently outperformed smaller ones, particularly in identifying formal fallacies.

- b) **False Alarm Bias:** A systematic bias was observed in which LLMs tend to misjudge valid reasoning as invalid. While GPT-4o and o1-mini appear to be fairly balanced, smaller and less accurate models generated six times more false positives than false negatives on average.
  - c) **Prompt Method Comparison:** Different prompting techniques (zero-shot, few-shot, and chain-of-thought) were tested, but none significantly improved accuracy over zero-shot prompting. The larger models performed better across all methods.
2. **Fallacy Classification:** Classifying fallacies into 232 distinct types in the FAL-LACIES taxonomy proved challenging, with lower accuracy rates compared to identification tasks. However, GPT-4o, Claude 3.5 Sonnet, and Gemini 1.5 Pro still outperformed previously reported results by Hong et al. (2024) and achieved accuracies greater than 40% (GPT-4o: 47.3%, Claude 3.5 Sonnet: 42.1%, Gemini 1.5 Pro: 41.5%).
  3. **Fine-Tuning Performance:** Fine-tuning a GPT-4o Mini model significantly improved its performance in both identification (93.1% accuracy, an improvement of 11.4%) and classification tasks (78.1% accuracy, an improvement of 54.1%). Despite these gains, there is concern about overfitting to the specific dataset.
  4. **Fallacy Type Performance:** The analysis of fallacy type accuracies revealed distinct patterns in how language models identify and classify logical fallacies. Models performed best on common fallacies with distinctive names or on those that are less complex. In contrast, they struggled with uncommon and complex formal logic fallacies, those with semantic overlap, and abstract philosophical fallacies.
  5. **Fallacy Search:**
    - a) **Research:** The approach of using a single prompt with structured output to detect and analyze several fallacies at once proved to be successful and reliable. GPT-4o achieved a significantly higher F1-score of 0.442 compared to previous results reported by Helwe et al. (2024). False alarms had a negative effect, as performance was better ( $F1 = 0.521$ ) when texts without fallacies were excluded from the analysis. By instructing the model to apply the principle of charity and consider the argument in its strongest form, false alarms were

reduced and F1-scores improved, but more work is needed to fully address and mitigate this problem.

- b) **Practical Implementation:** The method developed for the fallacy search task was successfully implemented in the Fallacy Search web application. The application demonstrates the utility of cutting-edge LLMs in supporting critical thinking and argument analysis. However, certain weaknesses and inconsistencies in logical reasoning were revealed and discussed.

### 9.3 Interpretation of Results

The impressive fallacy identification accuracies in the range between 83% and 88% achieved by the flagship models of leading providers underscore the advanced capabilities of LLMs in processing complex linguistic patterns. The superior performance of larger models suggests that model size is a critical factor in handling nuanced tasks such as fallacy detection. This observation is consistent with existing literature, which highlights the importance of scale in improving model performance. It remains to be seen whether scaling laws hold up and further increases in model size keep improving complex reasoning capabilities. Simply stacking more layers in transformer-based LLMs may soon result in diminishing returns, and new approaches including breakthroughs at the algorithmic level may be necessary to reach the next stages.

The introduction of models like OpenAI's o1, which incorporate "internal reasoning" mechanisms based on chain-of-thought reasoning and reinforcement learning, represents a new approach to scaling compute OpenAI (2024). The fallacy identification accuracy of 85.9% achieved by o1-mini in this study is competitive with the best models tested, raising expectations for o1 results, once the model is available via API. Unlike traditional methods that focus predominantly on increasing model size during training, o1 leverages test-time compute to enhance performance dynamically during inference. This allows the model to allocate more computational resources for complex tasks, simulating a human-like "thinking" process. By breaking problems into manageable steps, refining strategies, and correcting errors in real time, OpenAI claims that o1 exceeds human PhD-level accuracy on a benchmark of physics, biology, and chemistry problems (GPQA). This paradigm shift suggests that scaling compute through reasoning-focused algorithms, rather than sheer size, can lead to significant performance enhancements for tasks requiring deep logical and mathematical reasoning.

The discovered false alarm bias, particularly prevalent in weaker models with about six times more false positives than false negatives on average, presents an intriguing

## 9 Discussion

finding. Although being overly skeptical might seem preferable to being gullible, this bias should be considered when deploying these models in real-world applications. False alarms could undermine the credibility of logical fallacy detection systems. For example, a tester of an early version of the Fallacy Search application provided critical feedback: "Actually, I'm struggling to find an example where it doesn't think the result is a fallacy." As discussed in Section 8.3.1 on logical reasoning, even after improving the system prompt, false positives persist to a certain extent, especially for valid formal reasoning.

The lack of performance differences found for zero-shot, few-shot, and chain-of-thought (CoT) prompting may be explained by several factors. The fallacy identification task based on the short and isolated arguments in the FALLACIES dataset may not require the multi-step reasoning processes that CoT prompting is designed to enhance. CoT prompting typically excels in tasks like arithmetic or symbolic reasoning, where intermediate reasoning steps are critical (Kojima et al., 2023). If fallacy identification relies primarily on the recognition of semantic inconsistencies or discourse patterns, simpler prompting methods might suffice. Moreover, if the dataset contains linguistic cues or shortcut patterns that the models can exploit, all prompting methods may yield similar performance. The suitability of the relatively short text samples in the FALLACIES dataset as a rigorous test of advanced reasoning capabilities remains debatable. The substantial performance gains observed when models are fine-tuned on the dataset suggest that LLMs may exploit subtle linguistic patterns that correlate with ground-truth labels. Such patterns may have been unintentionally introduced by human annotators during dataset construction by making slight changes in wording when deriving valid arguments from invalid ones. Additionally, benchmark leakage cannot be entirely dismissed, as some recent models might have been trained on text corpora that include the publicly available FALLACIES text samples from GitHub. However, the FALLACIES benchmark was introduced only recently, reducing the risk of leakage.

It is uncertain whether future models will surpass 90% accuracy on the FALLACIES benchmark or if a performance ceiling has been reached. Although the overall quality of the dataset appears to be high, some ground-truth labels are questionable. For instance, the following examples are labeled as valid reasoning in the dataset, but have been justifiably flagged as fallacious by several stronger models:

- "Since I will visit Cornwallis or I will stay home this weekend and I have decided to visit Cornwallis, it does not necessarily follow that I cannot stay home this weekend."

- "Since Dr. Ironstone, a renowned metallurgist, asserts that siderosomes are the most abundant element in the earth's core and there is a significant body of empirical evidence supporting this claim, therefore, siderosomes are the most abundant element in the earth's core."
- "Since the weather house indicates sunny weather and the weather house can only show sunny or rainy weather, therefore, it is at least as likely it is sunny as it is sunny or rainy."

A more systematic review by human experts of arguments in the FALLACIES dataset may be warranted, especially for items where there is disagreement between LLM responses and ground-truth labels. It remains unclear whether questionable labels result in more false positives or false negatives.

Differences in identification and classification accuracies across fallacy types may partially be explained by their respective frequencies in the text corpora used to train LLMs. The more examples of a certain fallacy are present in the training material, the better the expected performance. While some formal logical fallacies are rarely discussed outside introductory logic books, other fallacies are frequently used in debates and thus more commonly addressed in literature. However, this analysis was performed on pooled responses from both small and large models. The finding that powerful LLMs have narrowed or even closed the performance gap between formal and informal logical fallacies within the FALLACIES dataset reflects their increasing capacity for complex reasoning.

#### **9.4 Limitations of the Study**

As this research has shown, LLMs can simulate logical reasoning to a remarkable degree. However, they may or may not engage in "true" reasoning, as understood in philosophical terms. The philosophical exploration of whether LLMs possess genuine reasoning capabilities falls outside the scope of this thesis. Such discussions are more appropriately situated within the domain of philosophy of mind. This limitation highlights the need for further interdisciplinary research to bridge the gap between computational models and philosophical understandings of reasoning.

To narrow the focus, complex multi-step reasoning tasks have not been included in this thesis. Although the tested LLMs performed well in detecting, classifying, and analyzing a broad spectrum of logical fallacies, they often encounter difficulties with more complex reasoning tasks. While they can simulate logical rules, they frequently struggle

to effectively chain these rules together to produce and verify complex conclusions. This limitation may impact the accuracy of fallacy detection, particularly in intricate arguments where multiple logical steps are involved. As a result, the models may fail to identify or correctly classify fallacies that require deeper logical analysis of multi-step arguments.

This thesis used the FALLACIES and MAFALDA datasets to evaluate performance on logical fallacy detection and classification. These datasets are comprehensive, yet limited in scope. They provide a structured environment for training and evaluating models, but any biases or gaps in these datasets may lead to incomplete or skewed assessments of model capabilities. The extent to which findings from these datasets can be generalized to other datasets or real-world scenarios remains uncertain. Real-world applications require a holistic understanding of arguments that goes beyond isolated fallacy detection. Current metrics may overlook subtleties that are crucial for accurately assessing logical coherence and validity in complex argumentative contexts.

Lastly, LLMs can inherit biases from their training data, which may lead to biased outputs when detecting fallacies. This is a critical concern as it affects the fairness and reliability of fallacy detection systems. Ensuring that models do not perpetuate or amplify existing biases is essential for developing ethical AI systems capable of fair and unbiased reasoning.

## 9.5 Future Research Directions

Future research could benefit from establishing a comprehensive human benchmark using the FALLACIES dataset. This would involve assessing human performance in identifying logical fallacies within the dataset, providing a baseline for comparing LLM performance. Such a benchmark could highlight areas where LLMs excel or lag behind human reasoning, offering insights into the strengths and limitations of AI in logical reasoning tasks. In the context of this study, a preliminary self-assessment test with 180 arguments was created by taking a stratified random sample from the FALLACIES dataset. The author achieved an accuracy of 86.0% on that test, spending an average of 60 seconds per argument. This performance likely exceeds the human average, given the author's expertise in logical fallacies and experience in logical reasoning. The comparable or even superior accuracies achieved by top-performing LLMs are certainly noteworthy. Before a broader application of the test, a more rigorous psychometric methodology should be applied to validate the test and narrow the selection to high-quality items.

To advance the understanding of LLM capabilities, future studies should focus on

more complex reasoning involving multi-step arguments. Current models often struggle with chaining logical steps effectively, which is crucial for evaluating intricate arguments. Investigating how LLMs handle these complex scenarios could lead to improvements in their reasoning abilities and better fallacy detection.

Exploring how variations in prompts affect LLMs' logical reasoning abilities is another promising avenue for future research. Different prompt formulations may influence model output significantly, affecting their effectiveness in identifying fallacies. Systematic experimentation with prompt engineering could uncover optimal strategies for maximizing model performance in logical reasoning tasks.

Testing advanced reasoning models such as OpenAI's o1 model could offer insights into cutting-edge capabilities in logical fallacy detection. These models may set new benchmarks for accuracy and efficiency, providing a glimpse into the future potential of AI-driven reasoning systems. Evaluating these models will help identify areas for improvement and innovation in logical reasoning technologies.

Finally, the Fallacy Search application developed during this thesis should undergo more systematic testing to evaluate its practical utility and reliability. This involves assessing its performance across diverse datasets and real-world scenarios to ensure its robustness and generalizability. Such testing will help refine the application, making it a more effective tool for detecting logical fallacies.

## 9.6 Conclusion

This thesis explored the capabilities of LLMs in detecting and classifying logical fallacies, a task that combines elements of artificial intelligence, linguistics, and philosophy. The research was driven by the increasing prevalence of misinformation and manipulative content in digital media, highlighting the need for effective tools to support critical thinking and argument analysis.

The study found that LLMs show promising results in detecting logical fallacies, including binary identification tasks and classifying fallacies into specific categories. Different prompting techniques were explored to improve results, but few-shot and chain-of-thought prompting yielded comparable results to zero-shot prompting. Fine-tuning models improved performance on specific datasets, although challenges such as overfitting and generalization remain. The FALLACIES and MAFALDA datasets provided a robust framework for evaluation, but highlighted difficulties due to semantic overlap of fallacy types. In general, larger models consistently outperformed smaller ones, highlighting the importance of model size in handling intricate reasoning tasks.

## *9 Discussion*

Despite these advancements, several weaknesses were identified. Smaller models struggled with formal logical fallacies compared to informal ones, highlighting the importance of model size for complex reasoning tasks. Additionally, the tendency towards false alarms in fallacy detection needs to be taken into consideration when developing real-world applications.

The Fallacy Search web application, developed as part of this thesis, serves as a practical implementation of the research findings on logical fallacy detection using LLMs. The application allows users to input lengthy texts, such as newspaper articles or blog posts, and outputs a list of identified logical fallacies along with a summary and a reasoning quality score. By transitioning from research to a real-world tool, the application demonstrates the feasibility of using advanced LLMs for practical fallacy detection and analysis, supporting critical thinking and enhancing public discourse by identifying manipulative or flawed reasoning in various contexts.

In conclusion, this thesis demonstrated the potential of LLMs in detecting and classifying logical fallacies, bridging the domains of artificial intelligence, linguistics, and philosophy to address a pressing societal need. While the results underscore the promise of LLMs in supporting critical thinking and argument analysis, they also revealed limitations such as challenges with formal fallacies, false positives, and generalization across datasets.

## References

- Aberdein, A. (2023). The Fallacy Fallacy: From the Owl of Minerva to the Lark of Arete. *Argumentation*, 37(2), 269–280. <https://doi.org/10.1007/s10503-023-09595-9>
- Akpan, M. (2024, July). Have We Reached AGI? Comparing ChatGPT, Claude, and Gemini to Human Literacy and Education Benchmarks [arXiv:2407.09573]. <https://doi.org/10.48550/arXiv.2407.09573>
- Alhindhi, T., Chakrabarty, T., Musi, E., & Muresan, S. (2022, December). Multitask Instruction-based Prompting for Fallacy Recognition [arXiv:2301.09992]. <https://doi.org/10.48550/arXiv.2301.09992>
- Anthropic. (2024, October). Introducing computer use, a new Claude 3.5 Sonnet, and Claude 3.5 Haiku. Retrieved December 12, 2024, from <https://www.anthropic.com/news/3-5-models-and-computer-use>
- Bach, S. (2024, February). Large language model training: How three training phases shape LLMs. Retrieved December 5, 2024, from <https://snorkel.ai/blog/large-language-model-training-three-phases-shape-lm-training/>
- Bartolo, M., Thrush, T., Riedel, S., Stenetorp, P., Jia, R., & Kiela, D. (2022, May). Models in the Loop: Aiding Crowdworkers with Generative Annotation Assistants [arXiv:2112.09062 [cs]]. <https://doi.org/10.48550/arXiv.2112.09062>
- Bennet, B. (2021, May). *Logically Fallacious: The Ultimate Collection of Over 300 Logical Fallacies*. eBookit.com.
- Bonial, C., Blodgett, A., Hudson, T., Lukin, S. M., Micher, J., Summers-Stay, D., Sutor, P., & Voss, C. (2022, June). The Search for Agreement on Logical Fallacy Annotation of an Infodemic. In N. Calzolari, F. Béchet, P. Blache, K. Choukri, C. Cieri, T. Declerck, S. Goggi, H. Isahara, B. Maegaard, J. Mariani, H. Mazo, J. Odijk, & S. Piperidis (Eds.), *Proceedings of the Thirteenth Language Resources and Evaluation Conference* (pp. 4430–4438). European Language Resources Association. Retrieved December 11, 2024, from <https://aclanthology.org/2022.lrec-1.471>
- Boudry, M. (2017, June). The Fallacy Fork: Why It's Time to Get Rid of Fallacy Theory. Retrieved May 12, 2024, from <https://maartenboudry.be/2017/06/the-fallacy-fork-why-its-time-to-get.html>
- Bouygues, H. L. (2022, August). Critical Thinking Skills Not Emphasized By Most Middle School Teachers [Section: Education]. Retrieved May 26, 2024, from <https://www.forbes.com/sites/helenleebouygues/2022/08/17/critical-skills-not-emphasized-by-most-middle-school-teachers/>

## References

- Bowman, S. R., Angeli, G., Potts, C., & Manning, C. D. (2015, September). A large annotated corpus for learning natural language inference. In L. Márquez, C. Callison-Burch, & J. Su (Eds.), *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing* (pp. 632–642). Association for Computational Linguistics. <https://doi.org/10.18653/v1/D15-1075>
- Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., ... Amodei, D. (2020, July). Language Models are Few-Shot Learners [arXiv:2005.14165 [cs]]. <https://doi.org/10.48550/arXiv.2005.14165>
- Chase, H. (2022, October). LangChain [original-date: 2022-10-17T02:58:36Z]. Retrieved November 23, 2024, from <https://github.com/langchain-ai/langchain>
- Chopra, D. (2024, February). Why Your Brain Isn't Experiencing Anything. Retrieved December 2, 2024, from <https://deepakchopra.medium.com/why-your-brain-isnt-experiencing-anything-dad59b85238b>
- Christiano, P. F., Leike, J., Brown, T., Martic, M., Legg, S., & Amodei, D. (2017). Deep Reinforcement Learning from Human Preferences. *Advances in Neural Information Processing Systems*, 30. Retrieved December 5, 2024, from [https://proceedings.neurips.cc/paper\\_files/paper/2017/hash/d5e2c0adad503c91f91df240d0cd4e49-Abstract.html](https://proceedings.neurips.cc/paper_files/paper/2017/hash/d5e2c0adad503c91f91df240d0cd4e49-Abstract.html)
- Cook, J. (2024, July). OpenAI's 5 Levels Of 'Super AI' (AGI To Outperform Human Capability) [Section: Entrepreneurs]. Retrieved December 12, 2024, from <https://www.forbes.com/sites/jodiecook/2024/07/16/openais-5-levels-of-super-ai-agi-to-outperform-human-capability/>
- Critical thinking [Page Version ID: 1224032996]. (2024, May). Retrieved May 17, 2024, from [https://en.wikipedia.org/w/index.php?title=Critical\\_thinking&oldid=1224032996](https://en.wikipedia.org/w/index.php?title=Critical_thinking&oldid=1224032996)
- Curtis, G. N. (2001, March). The Fallacy Files. Retrieved May 18, 2024, from <https://www.fallacyfiles.org/howtouse.html>
- Da San Martino, G., Yu, S., Barrón-Cedeño, A., Petrov, R., & Nakov, P. (2019, November). Fine-Grained Analysis of Propaganda in News Article. In K. Inui, J. Jiang, V. Ng, & X. Wan (Eds.), *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)* (pp. 5636–5646). Association for Computational Linguistics. <https://doi.org/10.18653/v1/D19-1565>
- Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019, May). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding [arXiv:1810.04805 [cs] version: 2]. <https://doi.org/10.48550/arXiv.1810.04805>
- Dutilh Novaes, C. (2022). Argument and Argumentation. In E. N. Zalta & U. Nodelman (Eds.), *The Stanford Encyclopedia of Philosophy* (Fall 2022). Metaphysics Research Lab, Stanford University. Retrieved May 30, 2024, from <https://plato.stanford.edu/archives/fall2022/entries/argument/>

## References

- Fallacy [Page Version ID: 1214270216]. (2024, March). Retrieved May 10, 2024, from <https://en.wikipedia.org/wiki/Fallacy>
- Goffredo, P., Haddadan, S., Vorakitphan, V., Cabrio, E., & Villata, S. (2022). Fallacious Argument Classification in Political Debates [ISSN: 1045-0823], 5, 4143–4149. <https://doi.org/10.24963/ijcai.2022/575>
- Habernal, I., Hannemann, R., Pollak, C., Klamm, C., Pauli, P., & Gurevych, I. (2017, September). Argotario: Computational Argumentation Meets Serious Games. In L. Specia, M. Post, & M. Paul (Eds.), *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing: System Demonstrations* (pp. 7–12). Association for Computational Linguistics. <https://doi.org/10.18653/v1/D17-2002>
- Habernal, I., Pauli, P., & Gurevych, I. (2018, May). Adapting Serious Game for Fallacious Argumentation to German: Pitfalls, Insights, and Best Practices. In N. Calzolari, K. Choukri, C. Cieri, T. Declerck, S. Goggi, K. Hasida, H. Isahara, B. Maegaard, J. Mariani, H. Mazo, A. Moreno, J. Odijk, S. Piperidis, & T. Tokunaga (Eds.), *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*. European Language Resources Association (ELRA). Retrieved May 18, 2024, from <https://aclanthology.org/L18-1526>
- Habernal, I., Wachsmuth, H., Gurevych, I., & Stein, B. (2018). Before Name-calling: Dynamics and Triggers of Ad Hominem Fallacies in Web Argumentation [arXiv: 1802.06613 [cs]]. *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, 386–396. <https://doi.org/10.18653/v1/N18-1036>
- Haddadan, S., Cabrio, E., & Villata, S. (2019, July). Yes, we can! Mining Arguments in 50 Years of US Presidential Campaign Debates. In A. Korhonen, D. Traum, & L. Màrquez (Eds.), *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics* (pp. 4684–4690). Association for Computational Linguistics. <https://doi.org/10.18653/v1/P19-1463>
- Hansen, H. (2023). Fallacies. In E. N. Zalta & U. Nodelman (Eds.), *The Stanford Encyclopedia of Philosophy* (Spring 2023). Metaphysics Research Lab, Stanford University. Retrieved May 18, 2024, from <https://plato.stanford.edu/archives/spr2023/entries/fallacies/>
- Helwe, C., Calamai, T., Paris, P.-H., Clavel, C., & Suchanek, F. (2024, April). MAFALDA: A Benchmark and Comprehensive Study of Fallacy Detection and Classification [arXiv:2311.09761]. <https://doi.org/10.48550/arXiv.2311.09761>
- Hong, R., Zhang, H., Pang, X., Yu, D., & Zhang, C. (2024, March). A Closer Look at the Self-Verification Abilities of Large Language Models in Logical Reasoning [arXiv:2311.07954 [cs]]. <https://doi.org/10.48550/arXiv.2311.07954>
- Hugging Face. (2024, November). Retrieved November 23, 2024, from <https://huggingface.co/>
- Imfeld, A. (2024a, November). Detecting Logical Fallacies with LLMs [Computer software] [original-date: 2024-10-14T11:39:48Z]. Retrieved November 23, 2024, from <https://github.com/aimfeld/fallacy-detection>
- Imfeld, A. (2024b, December). Fallacy Search Web Application [Computer software]. Retrieved December 1, 2024, from <https://github.com/aimfeld/fallacy-search>

## References

- Jin, Z., Lalwani, A., Vaidhya, T., Shen, X., Ding, Y., Lyu, Z., Sachan, M., Mihalcea, R., & Schoelkopf, B. (2022, December). Logical Fallacy Detection. In Y. Goldberg, Z. Kozareva, & Y. Zhang (Eds.), *Findings of the Association for Computational Linguistics: EMNLP 2022* (pp. 7180–7198). Association for Computational Linguistics. <https://doi.org/10.18653/v1/2022.findings-emnlp.532>
- Kojima, T., Gu, S. S., Reid, M., Matsuo, Y., & Iwasawa, Y. (2023, January). Large Language Models are Zero-Shot Reasoners [arXiv:2205.11916 [cs]]. <https://doi.org/10.48550/arXiv.2205.11916>
- Lee, J., Sakaguchi, K., & Bak, J. (2024, November). Self-Training Meets Consistency: Improving LLMs' Reasoning With Consistency-Driven Rationale Evaluation [arXiv:2411.06387 [cs]]. <https://doi.org/10.48550/arXiv.2411.06387>
- Li, Y., Wang, D., Liang, J., Jiang, G., He, Q., Xiao, Y., & Yang, D. (2024, April). Reason from Fallacy: Enhancing Large Language Models' Logical Reasoning through Logical Fallacy Understanding [arXiv:2404.04293 [cs]]. <https://doi.org/10.48550/arXiv.2404.04293>
- Lim, G., & Perrault, S. T. (2024, April). Evaluation of an LLM in Identifying Logical Fallacies: A Call for Rigor When Adopting LLMs in HCI Research [arXiv:2404.05213 [cs]]. <https://doi.org/10.48550/arXiv.2404.05213>
- Liu, H., Ning, R., Teng, Z., Liu, J., Zhou, Q., & Zhang, Y. (2023, May). Evaluating the Logical Reasoning Ability of ChatGPT and GPT-4 [arXiv:2304.03439 [cs]]. <https://doi.org/10.48550/arXiv.2304.03439>
- MacCartney, B., & Manning, C. D. (2007). Natural logic for textual inference. *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, 193–200. <https://aclanthology.org/W07-1431.pdf>
- Małkiński, M., Pawlonka, S., & Mańdziuk, J. (2024, November). Reasoning Limitations of Multi-modal Large Language Models. A case study of Bongard Problems [arXiv:2411.01173 [cs]]. <https://doi.org/10.48550/arXiv.2411.01173>
- Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013, September). Efficient Estimation of Word Representations in Vector Space [arXiv:1301.3781 [cs]]. <https://doi.org/10.48550/arXiv.1301.3781>
- Mirzadeh, I., Alizadeh, K., Shahrokh, H., Tuzel, O., Bengio, S., & Farajtabar, M. (2024, October). GSM-Symbolic: Understanding the Limitations of Mathematical Reasoning in Large Language Models [arXiv:2410.05229 [cs]]. <https://doi.org/10.48550/arXiv.2410.05229>
- Musi, E., Aloumpi, M., Carmi, E., Yates, S., & O'Halloran, K. (2022). Developing Fake News Immunity: Fallacies as Misinformation Triggers During the Pandemic [Publisher: Bastas]. *Online Journal of Communication and Media Technologies*, 12(3), e202217. <https://doi.org/10.30935/ojcmt/12083>
- OpenAI. (2024, September). Learning to Reason with LLMs. Retrieved December 17, 2024, from <https://openai.com/index/learning-to-reason-with-langs/>
- OpenAI Platform. (n.d.). Retrieved November 23, 2024, from <https://platform.openai.com>

## References

- Panagopoulou, A., Xue, L., Yu, N., Li, J., Li, D., Joty, S., Xu, R., Savarese, S., Xiong, C., & Niebles, J. C. (2024, September). X-InstructBLIP: A Framework for aligning X-Modal instruction-aware representations to LLMs and Emergent Cross-modal Reasoning [arXiv:2311.18799 [cs]]. <https://doi.org/10.48550/arXiv.2311.18799>
- Payandeh, A., Pluth, D., Hosier, J., Xiao, X., & Gurbani, V. K. (2023, August). How susceptible are LLMs to Logical Fallacies? [arXiv:2308.09853 [cs]]. <https://doi.org/10.48550/arXiv.2308.09853>
- Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., & Sutskever, I. (2021, July). Learning Transferable Visual Models From Natural Language Supervision. In M. Meila & T. Zhang (Eds.), *Proceedings of the 38th International Conference on Machine Learning* (pp. 8748–8763, Vol. 139). PMLR. <https://proceedings.mlr.press/v139/radford21a.html>
- Renze, M., & Guven, E. (2024, October). The Effect of Sampling Temperature on Problem Solving in Large Language Models [arXiv:2402.05201]. <https://doi.org/10.48550/arXiv.2402.05201>
- Sagan, C. (1995). *The Demon-haunted World: Science as a Candle in the Dark* [Google-Books-ID: ybvaAAAAMAAJ]. Random House.
- Sahai, S., Balalau, O., & Horincar, R. (2021, August). Breaking Down the Invisible Wall of Informal Fallacies in Online Discussions. In C. Zong, F. Xia, W. Li, & R. Navigli (Eds.), *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)* (pp. 644–657). Association for Computational Linguistics. <https://doi.org/10.18653/v1/2021.acl-long.53>
- Salvi, F., Ribeiro, M. H., Gallotti, R., & West, R. (2024, March). On the Conversational Persuasiveness of Large Language Models: A Randomized Controlled Trial [arXiv:2403.14380 [cs]]. <https://doi.org/10.48550/arXiv.2403.14380>
- Smith, R. (2000, March). Aristotle's Logic [Last Modified: 2022-11-22]. Retrieved May 29, 2024, from <https://plato.stanford.edu/ENTRIES/aristotle-logic/>
- Sourati, Z., Ilievski, F., Sandlin, H.-A., & Mermoud, A. (2023, May). Case-Based Reasoning with Language Models for Classification of Logical Fallacies [arXiv:2301.11879 [cs]]. <https://doi.org/10.48550/arXiv.2301.11879>
- Sourati, Z., Prasanna Venkatesh, V. P., Deshpande, D., Rawlani, H., Ilievski, F., Sandlin, H.-A., & Mermoud, A. (2023). Robust and explainable identification of logical fallacies in natural language arguments. *Knowledge-Based Systems*, 266, 110418. <https://doi.org/10.1016/j.knosys.2023.110418>
- Stechly, K., Valmeekam, K., & Kambhampati, S. (2024, August). On the Self-Verification Limitations of Large Language Models on Reasoning and Planning Tasks [arXiv: 2402.08115 [cs]]. <https://doi.org/10.48550/arXiv.2402.08115>
- Strachan, J. W. A., Albergo, D., Borghini, G., Pansardi, O., Scaliti, E., Gupta, S., Saxena, K., Rufo, A., Panzeri, S., Manzi, G., Graziano, M. S. A., & Becchio, C. (2024). Testing theory of mind in large language models and humans [Publisher: Nature Publishing Group]. *Nature Human Behaviour*, 1–11. <https://doi.org/10.1038/s41562-024-01882-z>

## References

- Streamlit. (2021, January). Retrieved December 1, 2024, from <https://streamlit.io/>
- Taye, M. M. (2023). Understanding of Machine Learning with Deep Learning: Architectures, Workflow, Applications and Future Directions. *Comput.*, 12, 91. <https://api.semanticscholar.org/CorpusID:258339625>
- The Wall Street Journal. (2024, June). Full Debate: Biden and Trump in the First 2024 Presidential Debate — WSJ. Retrieved December 2, 2024, from <https://www.youtube.com/watch?v=qqG96G8YdcE>
- Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.-A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., Rodriguez, A., Joulin, A., Grave, E., & Lample, G. (2023, February). LLaMA: Open and Efficient Foundation Language Models [arXiv:2302.13971 [cs]]. <https://doi.org/10.48550/arXiv.2302.13971>
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. u., & Polosukhin, I. (2017). Attention is All you Need. *Advances in Neural Information Processing Systems*, 30. Retrieved December 5, 2024, from <https://papers.nips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html>
- Veselovsky, V., Ribeiro, M. H., & West, R. (2023, June). Artificial Artificial Artificial Intelligence: Crowd Workers Widely Use Large Language Models for Text Production Tasks [arXiv:2306.07899]. <https://doi.org/10.48550/arXiv.2306.07899>
- Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., Chi, E., Le, Q., & Zhou, D. (2023, January). Chain-of-Thought Prompting Elicits Reasoning in Large Language Models [arXiv:2201.11903 [cs]]. <https://doi.org/10.48550/arXiv.2201.11903>
- Williams, A., Nangia, N., & Bowman, S. (2018, June). A Broad-Coverage Challenge Corpus for Sentence Understanding through Inference. In M. Walker, H. Ji, & A. Stent (Eds.), *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)* (pp. 1112–1122). Association for Computational Linguistics. <https://doi.org/10.18653/v1/N18-1101>
- Witsit [@WitsitGetsIt]. (2024, November). @RealAlexJones Even if the planets are spheres, how does that prove the earth is a globe? That's like saying the lights on your ceiling are bulbs, therefore the floor is a bulb. Retrieved December 1, 2024, from <https://x.com/WitsitGetsIt/status/1861410407937642947>
- Yeh, M.-H., Wan, R., & Huang, T.-H. ' (2024, October). CoCoLoFa: A Dataset of News Comments with Common Logical Fallacies Written by LLM-Assisted Crowds [arXiv:2410.03457]. <https://doi.org/10.48550/arXiv.2410.03457>
- Zhao, J., & Zhang, X. (2024, September). Exploring the Limitations of Large Language Models in Compositional Relation Reasoning [arXiv:2403.02615 [cs]]. <https://doi.org/10.48550/arXiv.2403.02615>

# Glossary

## **Ad Hominem (Abusive)**

A logical fallacy committed by attacking the person making the argument, rather than the argument itself, when the attack on the person is completely irrelevant to the argument the person is making. [9](#), [50](#)

## **Affirming the Consequent**

A formal logical fallacy that occurs when an argument assumes that if "If A, then B" is true, then the presence of B implies the presence of A. This reasoning is flawed because B could result from causes other than A, making the conclusion invalid. [9](#), [55](#)

## **Ambiguity Fallacy**

A logical fallacy that occurs when a word or phrase is used ambiguously within an argument, leading to misleading conclusions. [50](#)

## **Appeal to Authority**

Insisting that a claim is true simply because a valid authority or expert on the issue said it was true, without any other supporting evidence offered. [9](#), [50](#)

## **Appeal to Emotion**

An informal fallacy where emotions such as fear, pity, or joy are manipulated to persuade an audience, rather than relying on logical reasoning or factual evidence. [2](#)

## **Appeal to False Authority**

Using an alleged authority as evidence in your argument when the authority is not really an authority on the facts relevant to the argument. [50](#)

## **Appeal to Intuition**

A fallacious argument that relies on intuitive judgment or gut feeling rather than evidence or reasoned argumentation. [50](#)

## *Glossary*

### **Appeal to Trust**

The belief that if a source is considered trustworthy or untrustworthy, then any information from that source must be true or false, respectively. [50](#)

### **Argument from False Authority**

When a person making a claim is presented as an expert who should be trusted when his or her expertise is not in the area being discussed. [50](#)

### **Begging the Question**

A logical fallacy where the conclusion of an argument is assumed in the premise, essentially arguing in a circle without providing actual evidence. [56](#)

### **Cherry Picking**

A fallacy in which only select evidence is presented in order to persuade the audience to accept a position, while evidence that would go against the position is withheld. [50](#)

### **Denying the Antecedent**

A formal fallacy of inferring the inverse from an implication; if 'If P then Q' is true, then 'Not P implies Not Q' is incorrectly assumed to be true. [50](#)

### **Existential Fallacy**

A formal error in reasoning that arises from assuming that a universal claim implies the existence of something. Example: All forest creatures live in the woods. All leprechauns are forest creatures. Therefore, some leprechauns live in the woods. [50](#)

### **Fallacy of Composition**

A logical fallacy that occurs when it is assumed that what is true for the parts of a whole must also be true for the whole itself, without sufficient justification. [56](#)

### **Fallacy of Division**

An informal logical fallacy that occurs when one assumes that what is true for a whole must also be true for its individual parts. This reasoning error arises from incorrect inferences about part-whole relationships, leading to conclusions that are not logically supported by the premises. [56](#)

## *Glossary*

### **False Dichotomy**

A logical fallacy that involves presenting two opposing options as the only possibilities, when in fact more possibilities exist. [56](#)

### **False Dilemma**

An informal fallacy based on a premise that erroneously limits what options are available, often presenting two choices as if they are the only possibilities. [50](#)

### **Hasty Generalization**

A logical fallacy where a broad conclusion is drawn from a small or unrepresentative sample of data. [50](#), [56](#)

### **Having Your Cake Fallacy**

An informal fallacy where one assumes they can have two mutually exclusive outcomes simultaneously without acknowledging their incompatibility. [50](#)

### **Illicit Substitution of Identicals**

A formal logical error involving the inappropriate substitution of identicals in contexts where such substitution changes the truth value of statements. [50](#)

### **Limited Scope Fallacy**

A reasoning error where conclusions are drawn from premises with an unjustifiably narrow focus or perspective. Example: People often make hasty decisions because they don't take enough time to consider their choices. [50](#)

### **Modal Scope Fallacy**

A fallacy involving incorrect assumptions about necessity and possibility within modal logic. [50](#)

### **Modus Ponens**

A valid rule of inference in formal logic, also known as "affirming the antecedent." It states that if a conditional statement ("If P, then Q") is true and its antecedent (P) is true, then the consequent (Q) must also be true. This logical form is widely used in philosophy, mathematics, computer science, and everyday reasoning. [55](#)

## *Glossary*

### **Reification Fallacy**

A logical error where abstract concepts are treated as if they were concrete objects or entities. Example: If you are open to it, love will find you. [50](#)

### **Slippery Slope**

A logical fallacy that suggests a minor action will lead to major and oftentimes ludicrous consequences without sufficient evidence to support such claims. [50](#)

### **Stereotyping Fallacy**

An error in reasoning where generalized beliefs about a group are applied to individuals without sufficient evidence. [50](#)

### **Straw Man Fallacy**

An informal fallacy where an opponent's argument is misrepresented to make it easier to attack. [50](#), [58](#)

### **Subjectivist Fallacy**

Claiming something is true for one person, but not for someone else when, in fact, it is true for everyone (objective) as demonstrated by empirical evidence. [50](#)

### **Texas Sharpshooter Fallacy**

A fallacy where specific data points are chosen and emphasized to fit an argument, ignoring data that does not support it. [50](#)

# Acronyms

## BERT

Bidirectional Encoder Representations from Transformers (BERT) is a language model developed by Google in 2018. It uses a transformer-based architecture to process text bidirectionally, enabling it to understand the context of words based on their surrounding text. BERT is pre-trained on large text corpora like Wikipedia using self-supervised learning and can be fine-tuned for various natural language processing (NLP) tasks, such as question answering, sentiment analysis, and text classification. [12](#)

## GPT

A Generative Pretrained Transformer (GPT) is a type of large language model based on the transformer architecture. GPT excels at natural language understanding and generation, enabling applications such as text completion, summarization, translation, and conversational AI. [12](#)

## LLM

A Large Language Model (LLM) is a type of artificial intelligence that processes, understands, and generates human language using deep learning algorithms. [1](#)

## LSTM

A Long Short-Term Memory (LSTM) network is a specialized type of recurrent neural network (RNN) that addresses the vanishing gradient problem by introducing memory cells and gates (input, forget, and output). These gates control the flow of information, allowing LSTMs to retain or discard information selectively over long sequences. LSTMs excel in capturing long-term dependencies and are widely used in applications like machine translation, speech recognition, and time series prediction. [11](#)

## *Acronyms*

### **MRC**

Machine Reading Comprehension (MRC) is a natural language processing task that focuses on teaching machines to read and understand text, enabling them to answer questions based on the provided content. MRC models are evaluated on their ability to extract relevant information, infer meaning, and reason over text. [16](#)

### **NLI**

Natural Language Inference (NLI) is a task in natural language processing that involves determining the relationship between two sentences: a premise and a hypothesis. The task requires classifying the relationship as entailment, neutral, or contradiction. NLI is crucial for understanding and reasoning in various NLP applications, such as question answering, summarization, and information retrieval. It tests a model's ability to infer meaning and understand linguistic nuances. [16](#)

### **NLP**

Natural Language Processing (NLP) is a branch of artificial intelligence that enables computers to understand, interpret, and generate human language by employing various computational techniques. It involves tasks such as speech recognition, text classification, and natural-language understanding, leveraging methods like machine learning and deep learning to process and analyze large volumes of natural language data. [10](#), [16](#)

### **OOD**

Out-of-Distribution (OOD) refers to data that significantly differs from the training data distribution used by a machine learning model. OOD detection is crucial for ensuring the reliability and safety of machine learning systems, as it helps identify when a model encounters unfamiliar or unexpected input. This concept is particularly important in applications like natural language processing and computer vision, where models must generalize beyond their training data to handle real-world scenarios effectively. [16](#)

### **RNN**

A Recurrent Neural Network (RNN) is a type of artificial neural network designed to process sequential data by maintaining a hidden state that captures information from previous time steps. RNNs use feedback loops to allow information persistence, enabling them to model temporal dependencies and context in tasks such

### *Acronyms*

as natural language processing, speech recognition, and time series forecasting. However, they can suffer from issues like the vanishing gradient problem, which limits their ability to learn long-term dependencies. [11](#)

## List of Tables

1	Model Selection for the Fallacy Identification and Classification Tasks . . . . .	23
2	Fallacy Identification Accuracies . . . . .	39
3	Fallacy Identification Confusion Metrics . . . . .	42
4	Fallacy Identification Accuracies Across Prompt Methods and Models . . . . .	43
5	Fallacy Classification Accuracies . . . . .	47
6	Fallacies Ranked by Accuracy . . . . .	49
7	Common Fallacy Misclassifications . . . . .	51
8	MAFALDA F1-Scores for Fallacy Search . . . . .	52
9	Fallacy Identification with Rephrased Prompt Accuracies . . . . .	84
10	Fallacy Identification with Rephrased Prompt Confusion Metrics . . . . .	84
11	Fallacy Identification with Reliability Test Accuracies . . . . .	85
12	MAFALDA F1-Scores for Fine-Tuned Models . . . . .	86
13	MAFALDA F1-Scores for Fallacy Search with Uncovered Text Spans . . . . .	87

# List of Figures

1	Common Informal Logical Fallacies . . . . .	10
2	Taxonomy of Fallacies in the FALLACIES Dataset . . . . .	18
3	Comparison of Logical Fallacy Datasets . . . . .	19
4	Examples of Fallacies in the MAFALDA Dataset . . . . .	20
5	Taxonomy of Fallacies in the MAFALDA Dataset . . . . .	20
6	Number of Annotations per Fallacy Type in the MAFALDA Dataset . . . . .	21
7	Confusion Matrix and Key Metrics . . . . .	32
8	Example of Fallacy Search Metrics . . . . .	33
9	Fallacy Identification Zero-Shot Accuracies . . . . .	36
10	Fallacy Identification Zero-Shot Accuracies by Category and Model Group . . . . .	37
11	Formal Fallacy Identification Zero-Shot Accuracies by Model Group . . . . .	38
12	Informal Fallacy Identification Zero-Shot Accuracies by Model Group . . . . .	38
13	False Alarm Bias . . . . .	40
14	Comparison of Zero-Shot Confusion Matrices for GPT-4o and Mistral Small . . . . .	41
15	Fallacy Identification Accuracy by Model Group and Prompt Method . . . . .	43
16	Llama 3.1 8B Chain-of-Thought Confusion Matrix . . . . .	44
17	Fallacy Classification Accuracies . . . . .	46
18	Fallacy Classification Accuracies by Category and Model Group . . . . .	46
19	Fallacy Identification Fine-Tuning Accuracies . . . . .	48
20	Fallacy Classification Fine-Tuning Accuracies . . . . .	48
21	Fallacy Search Web Application Interface . . . . .	54
22	Fallacy Search Demonstrating Logical Reasoning . . . . .	55
23	Fallacy Search Analysis of Arguments Against Physicalism . . . . .	57
24	Fallacy Search Detection of Fallacies in Presidential Debate . . . . .	58

## List of Data Sources

Helwe, C. (2024, June). ChadiHelwe/MAFALDA [original-date: 2023-12-11T11:58:41Z]. Retrieved December 5, 2024, from <https://github.com/ChadiHelwe/MAFALDA>

Hong, R. (2024, November). Raising-hrx/FALLACIES [original-date: 2024-03-15T07:23:17Z]. Retrieved December 5, 2024, from <https://github.com/Raising-hrx/FALLACIES>

# **Appendices**

## Appendix A: Rephrased Prompt (Experiment 1.5)

Model	Formal					Informal					Avg.	
	prop.	quant.	syl.	prob.	Avg.	amb.	incon.	irrel.	insuf.	inappr.		
GPT-4o	80.0	<b>80.0</b>	<b>85.6</b>	<b>83.8</b>	<b>82.3</b>	<b>88.0</b>	85.0	<b>91.9</b>	92.8	<b>90.5</b>	89.6	<b>86.0</b>
Claude 3.5 Sonnet	<b>85.0</b>	75.0	81.2	80.0	80.3	85.7	<b>90.0</b>	91.0	<b>93.4</b>	89.4	<b>89.9</b>	85.1
Gemini 1.5 Flash	65.8	67.5	65.6	62.5	65.4	66.3	68.3	66.9	70.3	66.9	67.7	66.5

**Table 9: Fallacy Identification with Rephrased Prompt Accuracies.** The four formal and five informal subcategory percent accuracies (rose) are averaged to compute the mean formal and informal accuracies (violet), which are further averaged to yield the overall accuracy (green). Rephrasing the zero-shot prompt to "Does the following reasoning step contain a logical fallacy?" did not make a notable difference, compared to the accuracies for the zero-shot prompt "Is the following reasoning step correct?".

Model	TP	TN	FP	FN	FP/FN	Accuracy	Precision	Recall	F1
GPT-4o	2133	2064	256	187	1.4**	0.905	0.893	0.919	0.906
Claude 3.5 Sonnet	2146	2019	301	174	1.7***	0.898	0.877	0.925	0.900
Gemini 1.5 Flash	2278	857	1463	42	34.8***	0.676	0.609	0.982	0.752

**Table 10: Fallacy Identification with Rephrased Prompt Confusion Metrics.** The rephrased prompt had a positive effect for Claude 3.5 Sonnet where false positives dropped from 648 to 301, but a negative effect for the smaller Gemini 1.5 Flash where false positives increased from 618 to 1463, compared to the results for the zero-shot prompt. The rephrased prompt did not have the desired effect of systematically reducing false positives, at least not for the three tested LLMs. Significance is based on the McNemar test and levels are indicated by stars: \* for  $p \leq 0.05$ , \*\* for  $p \leq 0.01$ , and \*\*\* for  $p \leq 0.001$ . Accuracy is computed as the micro-average and therefore differs from the overall accuracy based on macro-averaging.

## Appendix B: Reliability Test (Experiment 1.6)

Model	Formal					Informal					Avg.	
	prop.	quant.	syl.	prob.	Avg.	amb.	incon.	irrel.	insuf.	inappr.		
GPT-4o Mini	<b>80.8</b>	75.0	83.1	77.5	79.1	<b>84.7</b>	<b>86.7</b>	<b>86.2</b>	<b>90.7</b>	<b>87.5</b>	<b>87.1</b>	<b>83.1</b>
Gemini 1.5 Flash	78.3	<b>80.8</b>	<b>86.2</b>	75.0	<b>80.1</b>	80.7	83.3	80.8	86.6	83.8	83.0	81.6
Claude 3 Haiku	69.2	74.2	80.6	<b>78.8</b>	75.7	81.0	78.3	82.7	85.9	83.1	82.2	78.9

**Table 11: Fallacy Identification with Reliability Test Accuracies.** The four formal and five informal subcategory percent accuracies (rose) are averaged to compute the mean formal and informal accuracies (violet), which are further averaged to yield the overall accuracy (green). In the reliability test, the rate of predictions flips from Yes to No and vice versa did not exceed 2% (GPT-4o Mini: 1.2%, Gemini 1.5 Flash: 1.9%, and Claude 3 Haiku: 0.0%). Setting the temperature to zero did not guarantee deterministic response, except for Claude 3 Haiku. The accuracies in this replication test did not change much, since some predictions flipped from correct to incorrect and vice versa, balancing each other out.

## Appendix C: Out-of-Distribution (OOD) Evaluation of Fine-Tuned Models

Level	All			Subset		
	0	1	2	0	1	2
GPT-4o	0.566	0.487	0.401	0.760	0.645	0.520
GPT-4o Mini	0.594	0.485	0.365	0.743	0.583	0.409
GPT-4o Mini Classification	0.474	0.375	0.275	0.648	0.504	0.358
GPT-4o Mini Identification	0.421	0.415	0.399	0.154	0.146	0.122

**Table 12: MAFALDA F1-Scores for Fine-Tuned Models.** The left section displays F1-scores based on all 200 annotated texts across three hierarchy levels, while the right section excludes texts without annotations, focusing on the subset of 137 texts with at least one fallacy annotation. The GPT-4o Mini models fine-tuned for the fallacy identification and classification tasks underperform on the fallacy search task, especially when only the subset with at least one fallacy annotation is considered where the F1-scores are not inflated by empty responses. This means that the fine-tuned models have either been overfitted and don't generalize well to out-of-distribution (OOD) challenges, or that the system prompt used for fine-tuning has a negative impact on performance.

## Appendix D: Fallacy Search F1-Scores with Uncovered Text Spans

Level	All			Subset		
	0	1	2	0	1	2
GPT-4o	0.399	0.306	0.264	0.582	0.447	0.385
GPT-4o Mini	0.396	0.291	0.219	0.578	0.425	0.320

**Table 13: MAFALDA F1-Scores for Fallacy Search with Uncovered Text Spans.** On the left, the F1-scores based on all 200 annotated texts are shown across the three hierarchy levels. On the right, texts without annotations were excluded, taking into account only the 137 texts with at least one fallacy annotation. By adding labels for uncovered text spans, the results are comparable with Helwe et al. (2024). However, due to a methodological problem with inconsistent labeling of the uncovered spans in the mentioned study, performance is likely underestimated. Nevertheless, level 2 F1-scores are still slightly higher in the present study.

**Master of Science (MScIDS) in Applied Information and Data Science**

**Declaration of originality**

The following declaration of originality must be signed by hand and included at the end of the Master's Thesis:

"The undersigned hereby declares that he or she

- wrote the work in question independently and without the help of any third party,
- has provided all the sources and cited the literature used,
- will protect the confidentiality interests of the client and respect the copyright regulations of Lucerne University of Applied Sciences and Arts."

Date and signature

18.12.2024 A. Imfeld

**Master of Science (MScIDS) in Applied Information and Data Science**

**Declaration of the use of Generative AI**

The following declaration of the use of Generative AI must be filled in and included at the end of the Master's Thesis.

It is not relevant for the grading.

For the work in question, I have used Generative AI to:  
(please select all that apply)

- brainstorm new ideas
- generate new research hypotheses
- do scientific research
- summarize other research
- set up the research/project design
- code
- write the manuscript
- edit or translate the text