

Fathom: A Fast and Modular Evidence Based Automated
Fact-Checking System

by

Farrukh Bin Rashid

Bachelors of Software Engineering, National University of Sciences and Technology,
2020

A Thesis Submitted in Partial Fulfilment of
the Requirements for the Degree of

Master of Computer Science

In the Graduate Academic Unit of Computer Science

Supervisor: Saqib Hakak, Ph.D., Computer Science

Examining Board: Fatemeh Khoda Parast, Ph.D., Computer Science,
Wei Song, Ph.D., Computer Science,
Mohsen Mohammadi, Ph.D., Mechanical Engineering

This thesis is accepted by the
Dean of Graduate Studies

THE UNIVERSITY OF NEW BRUNSWICK

July, 2025

© Farrukh Bin Rashid, 2025

Abstract

The growing spread of misinformation, driven by social media and generative AI, has underscored the urgent need for scalable Automated Fact-Checking (AFC) systems. These tools are essential for verifying claims efficiently, as manual efforts cannot keep pace with the volume and speed of online content. We present an evidence-based verification pipeline designed for both efficiency and real-world applicability. Evaluated on the AVeriTeC 2025 shared task, our system achieved an Ev2R score of 0.3423 on the development set and 0.2043 on the test set, with an average claim verification time of under 22 seconds. To test real-world effectiveness, we constructed a new dataset by scraping claims from PolitiFact and Snopes, pairing them with up-to-date supporting evidence and metadata such as source credibility and content type. On this dataset, our system achieved 87% accuracy on Snopes and 68% on PolitiFact, demonstrating strong performance in dynamic, real-world fact-checking beyond benchmark constraints.

Dedication

In the span of my degree, I have witnessed the unfolding of a sad reality, an ongoing tragedy marked by violence, displacement, and immense human suffering. Among the most heartbreaking has been the loss of innocent lives in Palestine, especially those of children whose futures were stolen before they could begin.

This thesis is dedicated to the people of Palestine, and to all those living under colonization and occupation, to the children who are no longer with us, the families torn apart, and the voices silenced too soon. To the colonized people of Palestine, whose struggle and resilience endure despite profound injustice.

May this work serve, in its own small way, as a testament to their memory etching their stories into these pages, where they will not be erased.

Acknowledgements

First and foremost, I would like to thank Allah (SWT), who granted me the opportunity to pursue this journey and guided me through every step of it. Despite my shortcomings, He held my hand and carried me through the most challenging moments. Without His mercy, strength, and guidance, none of this would have been possible.

I also want to thank my supervisor, Dr. Saqib Hakak, for his continuous support and guidance. I am incredibly grateful to him for providing me a chance to do research and always being there during this journey - best supervisor ever.

I wouldn't have been able to do this degree without the financial support provided by University of New Brunswick, Faculty of Computer Science, and the Canadian Institute of CyberSecurity (CIC), I am incredibly for their support throughout the two years.

I want to express my deepest gratitude to my family for their unwavering support and belief in me. Firstly I want to thank my Abbu (father), whose constant faith in me has been my greatest source of strength. He has always encouraged me to aim high and believe in my potential. It was his inspiration that led me to pursue a master's degree and commit to personal growth. Even today, his values and vision continue to guide and motivate me. Thank you Abbu, for everything. I also want

to thank my Ammi, who has been my constant companion over the past two years. From patiently listening to my rants to offering advice during my lowest moments, she has been a true pillar of strength. Everything I've achieved up to this degree is a result of her endless prayers and unwavering support. I am deeply grateful for her. Thank you Ammi, for everything. I want to thank my siblings as well, who have always shown me love, respect, and the kind of quiet strength that pushes me to grow into a better version of myself. To Hamdan and Mohib, thank you for holding down the fort while I've been away, chasing one degree after another. Just knowing that you both were there gave me the peace and comfort I needed to keep going. Your presence has meant more to me than you probably realize. To Faham, Jawaher, and Junaid, watching you grow into such thoughtful, kind, and capable human beings has been a quiet source of strength for me. Your existence inspires me to do better and to be someone you can always look up to.

I also want to thank my housemate, Dwight. I'm incredibly grateful to him, for everything from late-night tea and pizza runs, to countless trips to Walmart that saved me money on delivery. He even drove me an hour away to help me pass my driving test, never once complaining or putting his own time first. I truly couldn't have asked for a better housemate.

I also want to thank one of my closest friends in Fredericton, my gym partner and research collaborator, Arbaaz. With him, I didn't just get fitter and stronger, we also pushed each other to grow intellectually. Whether it was spotting each other at the gym or hopelessly brainstorming over research ideas, Arbaaz brought energy, focus, and a sense of perfection to everything we did. Working alongside him made both the workouts and the research so much more rewarding.

When I first arrived in Fredericton, one of my biggest concerns was staying connected to my faith finding the mosque, meeting fellow Muslim friends, and, of course, accessing halal food. I'd like to sincerely thank the Fredericton Islamic Association

for being a constant source of support for new Muslim students like me. From the comforting mutton biryani sessions to the warm, late-night suhoor gatherings, they created a space that truly felt like home away from home.

During my time in Fredericton, I was fortunate to meet some incredible fellow Muslims who truly inspired me to become not just a better Muslim, but a better person. I want to especially thank Abdul Qadir and Adnan, it's been an absolute honor standing behind them as they led prayers in Fredericton. Their quiet strength, unwavering faith, and constant presence have left a deep impact on me.

I'm especially grateful for the way they showed up for me pulling me toward the mosque during times when I lacked motivation, and never letting me drift too far. From our peaceful after Fajr coffee sessions to deep, late-night conversations after Isha, their companionship has been a grounding force and a source of real brotherhood.

Alongside them, I also had the privilege of building and leading the UNB Muslim Students' Association (MSA) here in Fredericton. From handing out iftar meals in the freezing cold to running across campus to make it to Taraweeh prayers, every moment was filled with shared purpose and a sense of community. Leading the MSA with Abdul Qadir and Adnan has been one of the most meaningful and rewarding experiences of my time here, and I'm incredibly thankful for the bond we've built both as friends and as brothers in faith.

And of course, saving the best for last. I want to thank four amazing people I met in late Fall 2023, Aniqah, Muzammil, Aamina, and Hassan. Over time, they became more than just friends, they became family. Together, we formed a group we call *tatreez*, a name that now carries with it so many memories, jokes, and shared moments. With them, I found a space where I could truly be myself and share

everything without hesitation. Each week became something to look forward to, knowing the weekends would bring laughter, board games, runs, and the kind of food comas only close friends can create. They've brought so much joy, comfort, and meaning to my time here, and I'm endlessly grateful for the bond we share and the memories we've made together.

Table of Contents

Abstract	ii
Dedication	iii
Acknowledgments	iv
Table of Contents	viii
List of Tables	xi
List of Figures	xii
1 Introduction	1
1.1 Introduction	1
1.2 Motivation and Objectives	5
1.3 Summary of Contributions	6
1.4 Thesis Organization	7
2 Literature Review	8
2.1 Overview	8
2.2 Definitions	8
2.3 Detection methods	9
2.3.1 Content based	10
2.3.2 Context Enriched methods	13
2.4 Related Works	16

2.4.1	Overview	16
2.4.2	Recent Advancements in AFC systems	18
2.4.3	Limitations	21
2.5	Concluding Remarks	22
3	Methodology	23
3.1	Overview	23
3.2	Claim Decomposition	25
3.3	Evidence Retrieval	28
3.3.1	Sparse Retrieval (BM25)	29
3.3.2	Semantic Re-ranking	30
3.4	Veracity Prediction	32
3.5	Concluding Remarks	33
4	Datasets	36
4.1	Overview	36
4.2	AVeriTeC dataset	38
4.2.1	AVeriTeC Score	39
4.3	FactCellar	40
4.3.1	Data Construction	42
4.3.2	Dataset Analysis	45
4.3.2.1	Snopes	45
4.3.2.2	Politifact	46
4.4	Concluding Remarks	47
5	Experiments and Results	53
5.1	Overview	53
5.2	Experimental Setup	53
5.3	Results	55

5.4	Concluding Remarks	58
6	Conclusion and Future Work	59
6.1	Conclusion	59
6.2	Future Work	60
	Bibliography	70
	Vita	

List of Tables

2.1	Overview of AVeriTeC submissions: brief descriptions and key limitations	17
4.1	Fact checking datasets	37
4.2	Class-wise distribution of train and dev sets of the AVeriTeC dataset.	39
5.1	AVeriTeC score on the dev set	56
5.2	Per-class F1 scores, overall accuracy, and macro-averaged F1 score on the development set.	57
5.3	AVeriTeC Scores on the test set	57
5.4	Results for the FactCellar dataset	58

List of Figures

1.1	Key categories of global economic costs of misinformation [21]	2
1.2	Illustration of a claim verified by the PolitiFact website [18]	4
3.1	Overview of the Fathom system architecture.	24
3.2	HyDE prompt with few shot samples	27
3.3	Example of generated QA pairs for a Refuted claim.	29
3.4	Prompt for our veracity prediction step.	34
3.5	Example of LLM-generated chain-of-thought reasoning with partial QA evidence (Q1–Q2 shown).	35
4.1	Prompt instructions for societal impact annotation task.	49
4.2	Snopes: Impact of True claims	50
4.3	Snopes: Impact of False claims	50
4.4	Politifact: Impact of True claims	51
4.5	Politifact: Impact of Mixed claims	51
4.6	Politifact: Impact of False claims	52

Chapter 1

Introduction

1.1 Introduction

Social media usage has increased worldwide over the past decade. As of year 2025, more than two-thirds of the world's population (around 5.3 billion people) use social media, up from roughly one-third in 2019 [33, 44]. This growth is driven by the increasing internet access and the number of platforms available such as Facebook, Youtube, Instagram and other similar platforms. For example, Facebook alone reports 3.07 billion active users and at least 7 other platforms boast over 1 billion users each. These figures highlight a shift in global communication, where social networks have now evolved from niche online communities into essential infrastructure for news, entertainment and a variety of other uses across all demographics and regions. On average, social media users engage with nearly 7 different platforms per month and spend over 2 hours per day on these apps [33].

This massive reach of social media has a troubling consequence though, the viral spread of misinformation (unintentional false information) and disinformation (deliberate deceptive information). A study in 2018 [62] showed that false news prop-

agates "farther, faster, deeper and more broadly" than true news on social media platforms. For example on Twitter, false claims are about 70% more likely to be retweeted by users than true claims, and they reach an audience of 1,500 people, six times faster than accurate news stories. Human psychology and engagement-driven algorithms are what fuel this propagation, because emotionally charged false information captures attention and gets shared rapidly which often outpaces the slower efforts of truthful information. As a result, this has created an information ecosystem in which misleading content often surpasses fact-based information in terms of visibility and reach.

Beyond its social and political implications, misinformation incurs significant economic costs. One study [21, 13] estimated that fake news inflicts about \$78 billion in global economic damage annually.

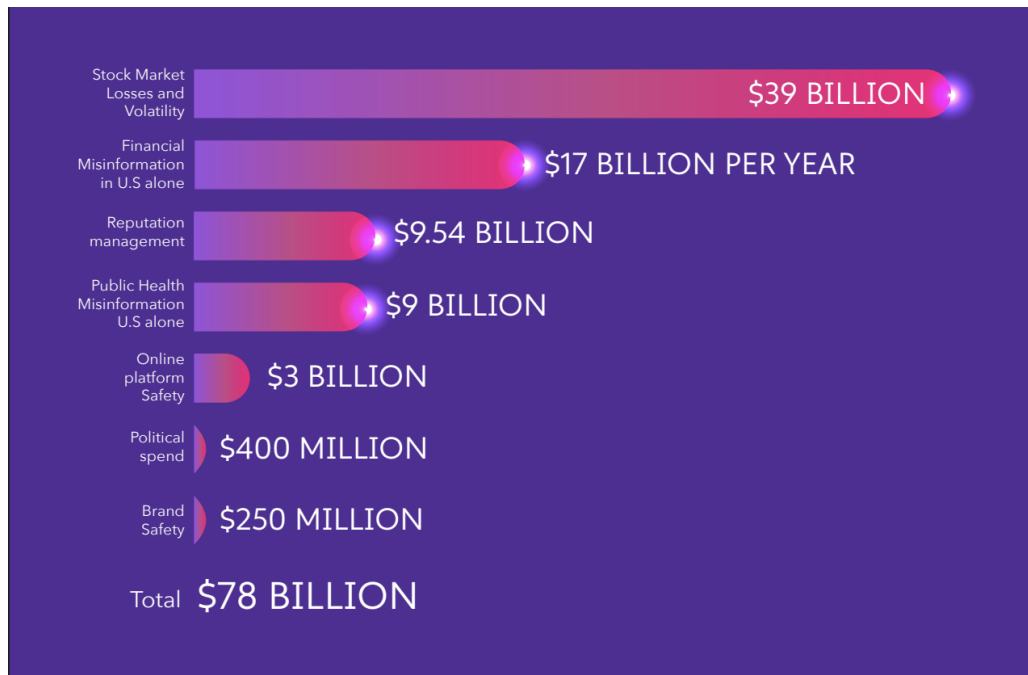


Figure 1.1: Key categories of global economic costs of misinformation [21]

Figure 1.1 reflects how mis/disinformation can ripple through economies, where bogus information triggers stock sell-offs, scams defraud investors, brands lose consumer

trust, and public health agencies spend resources countering medical falsehoods. For example, a single piece of misinformation in 2017 temporarily wiped out \$341 billion in the stock market in one day [20] by sending the S&P 500 into a brief panic, similarly a viral artificial intelligence generated hoax image of an explosion near Pentagon in 2023 caused a 0.3% intraday dip in the US stock market [38]. Beyond finances, the toll of misinformation is starkly seen in the health industry. One analysis found that COVID-19 vaccine misinformation in the U.S contributed to an estimated 319,000 avoidable deaths and cost the U.S. economy between \$50–300 million per day in health expenses [63].

Furthermore, a "disinformation-for-hire" industry has emerged. There are companies which now offer professional influence operations and fake news campaigns as a service, which blurs the lines between marketing and propaganda. For example, reports [22] indicate that in Israel a network of private firms staffed by former intelligence officers conduct coordinated online manipulation campaigns, thus allowing the political clients a measure of plausible deniability. Even during various elections and geopolitical conflicts, "troll farms" and digital mercenaries have been employed to spread false narratives, and stir up division online. These actors often operate thousands of fake social media accounts and websites, influencing public opinion while getting paid for outcomes (clicks or views). Essentially misinformation has been systematized into an industry. In January 2024, the World Economic Forum declared misinformation and disinformation as the greatest short-term risks worldwide [59].

In the face of this rampant misinformation and disinformation, fact-checking remains a vital safeguard for truth. Professional fact-checkers, often trained journalists or researchers, investigate claims manually by consulting credible sources and subject-matter experts. Their approach is systematic and transparent. For example, PolitiFact, a U.S.-based outlet, follows a structured process that includes

contacting sources, reviewing primary data, and conducting independent verification before reaching a verdict. An example of such a verified claim is illustrated in Figure 1.2. However, even though manual fact-checking is effective, it is inherently

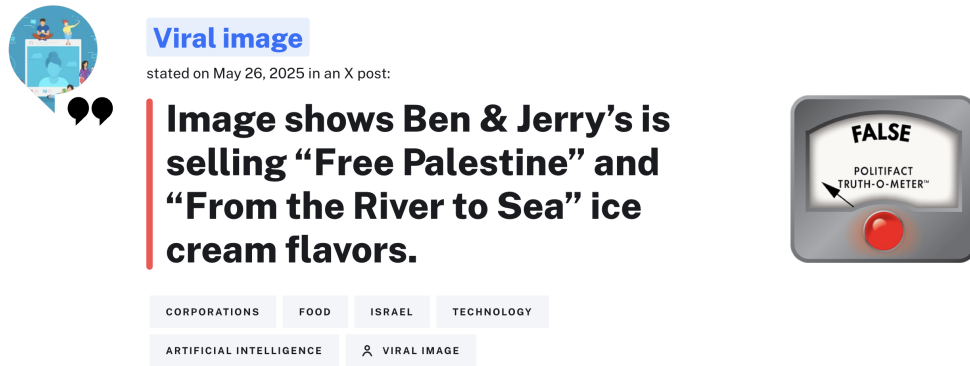


Figure 1.2: Illustration of a claim verified by the PolitiFact website [18]

labor-intensive and reactive, by the time a false claim is debunked, it may have already spread to millions. This limitation has prompted the development of automated fact-checking tools designed to assist researchers in identifying and addressing misinformation at scale. Early systems relied on linguistic features and traditional classifiers. For example, one method proposed in 2017, trained a support-vector machine on bag-of-words, part-of-speech tags, and entity-type features to detect factual claims [29]. In recent years, this approach has then given way to deep learning models, where these models are trained end-to-end on claim text for veracity classification. These methods rely on content-based analysis, which focuses solely on the textual content of a claim without incorporating any external context or supporting information beyond the claim itself.

However, content-based analysis alone often fails to catch misinformation, since fake news is deliberately crafted to deceive, so simple textual cues may reflect the writing more than the truth [28]. In practice, manual fact-checkers mitigate this by consulting multiple external sources, they identify a claim, gather evidence from di-

verse documents, check source credibility and verify its reasoning chain. Accordingly, modern systems increasingly now are proposing and researching into evidence based fact-checking systems. One approach is to enrich the claim using structured knowledge, wherein these pipelines apply named-entity recognition (NER) to the claim, link entities to a knowledge graph and then retrieve relevant facts for comparison [43]. However, with the advent of large language models (LLMs), recent systems typically retrieve external textual evidence via search or knowledge bases, then use LLMs to assess a claim’s veracity based on the retrieved evidence. As seen in the AVeriTeC Shared Task 2024 (a benchmark competition designed to evaluate systems on automated claim verification using textual evidence) [53], LLMs have become central to modern fact-checking pipelines, serving as the final reasoning component in all top-performing systems, and in some cases, being used at multiple stages. However, most of these systems relied on closed-source LLMs such as GPT-4o, which, while highly capable, come with limitations related to transparency, cost, and accessibility. In contrast, teams that employed open-source models like LLaMA showcased the potential of publicly available alternatives. Still, these approaches often required access to substantial computational resources, posing practical challenges for broader deployment [52].

1.2 Motivation and Objectives

This research aims to address the limitations of prior work by developing an evidence-based automated fact-checking system capable of retrieving and reasoning over relevant evidence in under a minute. Unlike previous studies that rely on extensive computational resources, our system demonstrates that high-performance fact-checking is achievable with limited resources by leveraging efficient retrieval methods and locally run open-source language models.

We present Fathom, a lightweight Retrieval-Augmented Generation (RAG) pipeline for automated fact-checking, built entirely with open-source language models. The system operates in three stages: it first generates hypothetical questions using a HyDE-style prompting approach to expand the context of a claim, it then performs a two-step retrieval process combining Best Matching-25 and semantic similarity to collect the most relevant evidence, finally a compact language model evaluates the claim’s veracity by reasoning over the retrieved evidence, producing both a verdict and an explanatory rationale. Despite using only lightweight models, Fathom delivers competitive performance while remaining resource-efficient and fully open.

1.3 Summary of Contributions

The main goal of this research is to develop an evidence-based fact-checking system that is fast, efficient, and capable of running under limited computational resources. This thesis makes the following key contributions:

- We present a Retrieval-Augmented Generation (RAG) pipeline that retrieves relevant evidence and reasons over it to classify claims into four categories: Supported, Refuted, Not Enough Evidence, and Conflicting Evidence or Cherry-picking.
- We introduce a new dataset, FactCellar, inspired by previous benchmarks such as the AVeriTeC dataset. FactCellar includes additional features like source credibility, impact scores, and more recent evidence.
- We evaluate our system on both the AVeriTeC dataset and FactCellar, and conduct experiments to assess its performance across various claim types and settings.

1.4 Thesis Organization

Chapter 1 discusses the global rise of social media and its role in accelerating the spread of misinformation and disinformation. Moreover, it touches upon the societal and economic impacts of disinformation, highlighting the growing need for automated fact-checking. Chapter 2 presents a comprehensive literature review, covering content-based detection methods, context-enriched approaches, and the growing need for automated fact-checking (AFC), including recent advancements in the field. Chapter 3 outlines our proposed methodology, detailing the overall architecture and system components. Chapter 4 describes the datasets used in our study, including both existing resources and the dataset we have constructed. Chapter 5 discusses our experimental results, including evaluations of both datasets. Finally, Chapter 6 presents our conclusions and outlines directions for future research.

Chapter 2

Literature Review

2.1 Overview

In this chapter, we begin by outlining the fundamental concepts of misinformation, disinformation, and fake news, establishing clear definitions and distinctions between them. We then explore the various detection methods that have been employed in past research, highlighting their methodologies and limitations. Building on this, we delve into the emergence of automated fact-checking as a response to these limitations, examining the motivations behind its development and surveying recent advancements in the field. Finally, we provide an overview of the datasets commonly used for automated fact-checking, critically analyzing their strengths and shortcomings.

2.2 Definitions

Misinformation refers to false or inaccurate information that is shared without any malicious intent. It can arise from errors, misconceptions or lack of knowledge and

although the information itself is false, the intention behind its dissemination is not deceitful [17].

Disinformation on the other hand refers to false information that is deliberately created and spread with the intention to deceive and manipulate others. Disinformation campaigns are typically designed to mislead, influence public opinion, cause confusion, or advance a specific agenda. An example of disinformation could be the deliberate creation of fake stories or manipulated data that are spread to sway voters, stir up conflict, or damage a person’s reputation [17].

Fake news lacks a universally accepted definition. However, a commonly cited narrow definition, as used by [17, 55], describes fake news as information that is both intentionally created and verifiably false. This definition highlights two key distinguishing features of fake news: authenticity and intentionality. These features help differentiate fake news from other related concepts.

For instance, when the authenticity of information is unverified and the intention behind it is unknown, the content is generally categorized as a rumor. If the information is false but the intention is not to deceive, it is considered misinformation. On the other hand, when both the authenticity is false and the intention is malicious, the content is classified as disinformation. It is important to note that fake news is a subset of disinformation, it is restricted specifically to news content, whereas disinformation can encompass any type of false and intentionally misleading information.

2.3 Detection methods

Researchers have addressed the challenge of detecting misinformation and disinformation using multiple analytical perspectives. While various detection frameworks exist, we specifically categorize them into two main types, content-based and context-

enriched. Content-based methods include a range of approaches such as Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), transformer-based models like Bidirectional Encoder Representations from Transformers (BERT), as well as traditional classifiers such as Support Vector Machines (SVMs) and logistic regression. In contrast, context-enriched methods focus on leveraging external textual evidence to support claim verification. This categorization highlights why automated fact-checking has emerged as essential for verifying claims beyond pure text analysis.

2.3.1 Content based

The underlying premise in content based detection is that misinformation or disinformation exhibits distinctive linguistic or semantic features that can be algorithmically identified. These features range from stylistic and grammatical irregularities to semantic inconsistencies and emotionally charged language. Broadly, we explore a range of content-based detection strategies, starting with classical machine learning approaches, followed by neural models such as CNNs and GNNs, and culminating in transformer-based architectures like BERT.

Early research focused on hand-crafted linguistic features such as n-gram frequencies, part-of-speech patterns and sentiment polarity combined with classical classifiers like Support Vector Machines (SVM), Logistic Regression (LR) and Random Forests (RF). These methods sought to capture observable stylistic or lexical differences between fake and real news. One notable research by Pérez-Rosas et al. [48] conducted automatic fake news detection using Linear Support Vector Machines (SVMs). They used linguistic features such as unigrams, bigrams, punctuation and syntactic structures. Additionally they incorporated psycholinguistic features covering emotional tone and pronoun usage. Their classifiers achieved 70% accuracy using

the features mentioned. Another study by Bali et al. [14], applied seven machine learning (ML) classifiers, including Gradient Boosting and XGBoost, to multiple datasets using n-gram features extracted from both headlines and full texts. Gradient Boosting yielded the best accuracy across all datasets, showing the effectiveness of simple lexical features combined with ensemble methods. While these methods are interpretable and computationally efficient, these models often suffer from generalizability across domains, as stylistic cues are sensitive to topic and genre.

As feature engineering become a bottleneck, researchers adopted deep learning methods, particularly Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) which learn the semantic and syntactic patterns from raw text. A notable example is the system proposed by Goldani et al. [25], which employs a deep learning model based on capsule networks for fake news detection. Capsule networks represent an advanced neural architecture designed to preserve hierarchical and spatial relationships between features, enabling the model to more effectively capture the structure and context of the input data. Rather than relying on traditional lexical or syntactic features, the authors utilized pre-trained GloVe word embeddings, which were fine-tuned during training to better align with the specific task.

The model architecture began with convolutional layers that extracted local n-gram features using filter sizes of 2, 3, 4, and 5. These features were then passed through a series of capsule layers comprising primary capsules, convolutional capsules, and a fully connected capsule layer designed to retain the spatial and contextual dependencies among features. The outputs from parallel capsule paths were aggregated using average pooling to generate the final classification decision.

When evaluated on the ISOT dataset, the model achieved an impressive 99.8% accuracy, significantly outperforming traditional baseline methods such as Support Vector Machines (SVM) and Logistic Regression (LR). This result highlights the effective-

ness of capsule networks in modeling spatial patterns within fake news text and their potential superiority in this domain. Additionally, [24] explored the performance of sequential deep learning models, vanilla RNNs, Gated Recurrent Unit (GRUs) and Long Short-Term Memory (LSTMs) on the LIAR dataset. The models were trained using preprocessed textual claims without external context or metadata. Among them, the GRU-based model yielded the highest accuracy at 21.7% marginally outperforming both LSTM and vanilla RNN variants. The authors attributes the overall low performance to the limited length and high ambiguity of the LIAR statements. These examples demonstrated improved generalization and abstraction abilities but typically required large annotated datasets and incurred higher training costs.

Recent work in misinformation or disinformation detection leverages transformer-based pre-trained language models like BERT, RoBERTa and XLNet. These models are pre-trained on massive corpora in a self supervised fashion and then fine-tuned on downstream tasks, including fake news classification. Aggarwal et al [8] demonstrates this, by fine-tuning BERT models for fake news classification, achieving superior performance compared to traditional deep learning and machine learning baselines. Without extensive preprocessing or handcrafted features, BERT attained an accuracy of 98% on the Fake News Detection (FND) dataset. In comparison, LSTM and gradient-boosted trees reached 92.5% and 94.5%, respectively. The study leveraged BERT’s contextualized word embeddings to better capture semantic nuances in news text, which traditional models struggled with. The results showed the advantage of transfer learning and pre-trained language models in low-resource and high-noise settings.

2.3.2 Context Enriched methods

A key limitation of content-based approaches is their reliance on analyzing only the textual or linguistic features within a claim to assess its veracity. These models operate under the assumption that deceptive content inherently differs in writing style or semantics from truthful content, an assumption that does not always hold. In reality, false claims can be intentionally crafted to closely mimic the structure, tone, and language of legitimate information, thereby reducing the effectiveness of content-driven methods. Moreover, content-based models are trained on static datasets and lack access to real-time information. This limits their ability to verify emerging events or novel claims. The dynamic nature of news means that such models, once trained, cannot adapt to unseen future developments.

To overcome these limitations, recent research [68, 49, 39] has shifted towards incorporating external content to enhance claim verification. Broadly, this contextual information falls under two categories: social signals (e.g., user interactions, propagation patterns) and textual evidence (e.g., supporting or refuting documents retrieved from external sources) [68]. In this thesis, we focus specifically on textual evidence-based verification, as it directly aligns with the objectives and design of the solution we propose.

One of the earliest evidence-aware fact verification systems is DeClarE [49], which introduced a complete neural pipeline for assessing the credibility of natural language claims using supporting or refuting web articles. The process begins by converting the claim and each associated article into word embeddings using GloVe [47]. These embeddings are then passed through a bidirectional LSTM to capture contextual representations of article content in both the directions. Next the model uses an attention mechanism which basically assigns higher weights to more relevant words (to the claim), which outputs a weighted summary of the article that focuses only on

claim relevant content (supporting or refuting). In parallel DeClarE learns embedding vectors for both the claim source and the article source which can be thought of as source-credibility. These embeddings, along with the attention focused article representation, are concatenated to form a single feature vector which is then passed through a feedforward neural network consisting of multiple fully connected layers with non-linear activations (e.g., ReLU). This network learns to model complex interactions between the content of the article, its relevance to the claim, and the credibility of the sources. The final layer outputs a credibility score (via sigmoid for classification or linear activation for regression), representing the likelihood that the claim is true based on that particular article. Since a single claim may be linked to multiple evidence articles, the model processes each article independently, and then averages the credibility scores from all associated articles to generate the final prediction for the claim’s overall credibility. DeClarE was evaluated across four datasets and results show that it consistently outperforms traditional baselines, while showing interpretability by highlighting which parts of the article contributed the most to the decision.

While DeClarE relies on unstructured evidence from web articles to assess the truthfulness of a claim, DEAP-FAKED [39] takes a different approach by using structured evidence from a knowledge graph. The system starts by encoding the news title using a BiLSTM, which captures the contextual meaning of the text. At the same time, it uses Named Entity Recognition (NER) to identify key entities in the title, such as people, places, or organizations. These entities are then linked to entries in Wikidata, and represented using ComplEx, a knowledge graph embedding method that captures both one-way and mutual relationships between entities. For example, the fact (Trump, birthCountry, USA) would be embedded into a meaningful vector that reflects real-world connections. The title representation and the entity

embeddings are then combined and passed into a classifier to predict whether the news is real or fake. DEAP-FAKED grounds its predictions in structured, relational knowledge, offering a fact-oriented alternative for fake news detection and achieving strong overall performance across two evaluated datasets.

Another graph-based approach, GETRAL [66], takes a different route by constructing graphs directly from the claim and its supporting evidence text, rather than relying on external knowledge bases. GETRAL detects fake news by modeling both the claim and its supporting evidence as word co-occurrence graphs. These graphs capture important word relationships, especially when keywords appear far apart in the text. A graph neural network processes these graphs to learn deep representations of both the claim and the evidence. Then, an attention mechanism selects the most relevant parts of the evidence by comparing them directly to the claim. These selected features which includes the claim, the focused evidence, and meta-data like the speaker or publisher are combined into a single vector. This vector is then passed to a classifier that predicts whether the claim is true or false. During training, the model uses contrastive learning to pull claims with the same label closer together and push apart those with different labels, making the predictions more stable and accurate. Empirically, GETRAL achieves state-of-the-art results on Snopes and PolitiFact, while using fewer parameters and computations than previous models, demonstrating that combining graph-based reasoning with contrastive learning is a powerful approach for fact-checking.

Early approaches focused on content based detection which, as discussed before, were fundamentally flawed. These approaches inherently relied on stylistic and linguistic features of a claim, which could be easily emulated to turn disinformation into a

legitimate claim. In order to address this, research progressed into context-enriched models which augment the claim with additional information such as supporting documents, knowledge graphs, or social metadata. Systems like DeClarE, DEAP-FAKED, and GETRAL exemplify this shift, introducing mechanisms to incorporate textual evidence or structured knowledge into the decision-making process.

While these methods improved accuracy and interpretability, they still relied on semi-manual pipeline components and domain-constrained evidence sources. Systems trained on structured knowledge graphs or curated document sets often depend on static corpora like Wikipedia. However, Wikipedia may not contain up-to-date or sufficiently detailed information to verify certain real-world claims, especially those related to recent events or niche-topics. On the other hand, open web articles might contain relevant evidence, but retrieving and filtering that evidence introduces new challenges, including noise, redundancy, and the need for manual or heuristic curation. Consequently, many of these systems remain limited in scalability and adaptability, particularly when applied outside their training domain or to unseen, evolving claims.

2.4 Related Works

2.4.1 Overview

These limitations motivated the rise of Automated Fact-Checking (AFC) as a more scalable, dynamic alternative [61]. Unlike earlier approaches that depended on pre-matched evidence or static sources, AFC systems strive to verify claims end-to-end, they automatically retrieve relevant evidence from large, often unstructured corpora and reason over it to decide claim veracity. Early prototypes such as ClaimBuster [30] and the FEVER baseline pipeline [58] demonstrated that a unified retrieval,

reasoning, classification workflow could handle diverse, open-domain claims, even though they still relied on sparse retrieval and shallow entailment models.

Table 2.1: Overview of AVeriTeC submissions: brief descriptions and key limitations

System	Description	Limitation
Infact [51]	Six-stage pipeline using GPT-4o for claim interpretation, sub-question generation, query generation, retrieval, answer synthesis, and final verdict.	High operational cost (\$0.46/claim), reliance on closed-source models, and long/inconsistent inference due to multi-stage design.
HerO [67]	Fully open-source LLM pipeline: HyDE-FC retrieval with LLaMA-3.1-70B, question generation with LLaMA-3.8B, and veracity prediction with fine-tuned LLaMA-3.1-70B.	Requires high-end hardware (e.g., two H100 GPUs with 480 GB RAM)
AIC CTU [60]	Lightweight RAG approach: BM25 + embedding + MMR retrieval, then unified GPT-4o-driven generation of question-answer pairs and verdict with chain-of-thought prompting.	Nontrivial API cost (9,000 GPT-4o calls, \$363 total).
Dunamu-ML [45]	Enhanced knowledge store via custom crawling (PDFs, YouTube transcripts), BM25 + dense re-rank stack, followed by GPT-4 for question generation and verdict prediction.	Dependence on closed-source GPT-4 for key subtasks, leading to API costs and reproducibility issues
Papelo [37]	Live Google search-based retrieval with multi-hop strategy: initial query with T5-large, follow-up queries via GPT-4o, iterative retrieval and reasoning over web results.	Relies on closed-source models (GPT-4o)

Since then, AFC has progressed beyond modular, pre-LLM architectures toward more integrated and flexible solutions that incorporate pretrained LLMs. These models enable end-to-end fact verification, from retrieving evidence to generating verdicts, and have significantly reshaped the landscape of automated fact-checking (AFC). A notable development in this area is the *AVeriTeC 2024 Shared Task* [53], which introduces a standardized benchmark for evaluating LLM-based fact-checking

systems. Designed to assess the entire verification pipeline, the task challenges systems to perform claim verification holistically, retrieving relevant evidence and predicting the final label. Since this is the most recent advancement in the area, in the following sections, we examine the top-performing submissions to the *AVeriTeC 2024 task* [53] (see Table 2.1 for a high-level summary), highlighting the diverse strategies they employed and the specific limitations they faced.

2.4.2 Recent Advancements in AFC systems

The top performing system in the AVeriTeC 2024 shared task, Infact [51], proposed a six-stage pipeline that mimics a structured human fact-checking process. The pipeline is built using large language models primarily using GPT-4o for best performance. In the initial stage, Infact interprets each claim by reformulating the claim using its metadata (author, date, source URL) to clarify ambiguities or temporal vagueness. Next, it uses GPT-4o to generate 10 focused sub-questions, which are designed to break down the claim into verifiable elements. In the third stage, GPT-4o is again used to generate search queries for each question which are then used in the semantic retrieval module (dense retrieval) to retrieve relevant documents from the AVeriTeC knowledge store. Once the relevant documents are retrieved Infact uses GPT-4o again to process the retrieved documents to answer each question which were generated in the first stage. Finally based on the question-answer pairs Infact leverages GPT-4o to produce a final verdict (Supported, Refuted, Not Enough Evidence, or Conflicting/Cherrypicking).

Another strong contender in the AVeriTeC 2024 shared task was HeRO [67], which secured the second-highest overall score, trailing just behind INFAC. HeRO adopts a fully open-source approach, relying on publicly available large language models at every stage of the fact-checking process. The pipeline begins with a novel evidence

retrieval method called HyDE-FC, where an instruction-tuned model (LLaMA-3.1-70B) generates hypothetical fact-checking documents to augment the claim. These documents are used to build hybrid retrieval queries that first retrieve 10,000 candidate sentences using BM25, followed by re-ranking with dense embeddings to select the top 10 evidential sentences. Then HeRO leverages LLaMA-3.8B to generate clarifying questions based on the claim and its corresponding evidence retrieved in the previous steps. In the final stage, the system performs veracity prediction using a fine-tuned LLaMA-3.1-70B. It predicts the final verdict by reasoning over the top question-answer pairs generated in the previous steps. HeRO’s success highlights the strong performance potential of open LLMs, achieving competitive accuracy without reliance on proprietary models like GPT-4o.

A third notable system in the AVeriTeC 2024 shared task was the AIC CTU system [60], which secured the third-highest overall score. Unlike the modular and heavily engineered pipelines of INFACt and HerO, AIC CTU framed fact-checking as a lightweight Retrieval-Augmented Generation (RAG) problem, prioritizing simplicity and leveraging the capabilities of large language models. The pipeline begins with a retrieval module that selects document chunks from a pre-built knowledge store. To do this efficiently, it first narrows down the candidate set using Best Matching 25 (BM25), a ranking function based on term frequency and document relevance, to filter out unrelated content. The remaining chunks are embedded using the mxbai-large-v1 [41] model, and cosine similarity is applied to find the most relevant passages for a given claim. To ensure both relevance and diversity, the top results are reranked using Maximal Marginal Relevance (MMR), which balances relevance to the query with novelty compared to already selected results.

The selected chunks are then passed to a large language model, typically GPT-4o, which is prompted to perform multiple tasks in a single step. Specifically, the model generates up to ten question-answer pairs that capture key factual dimensions of the

claim and produces a structured JSON output containing Likert-scale confidence scores for each possible verdict (Supported, Refuted, Not Enough Evidence, or Conflicting/Cherry-picking). The final verdict is then selected using a chain-of-thought reasoning strategy that encourages the model to justify its decision based on the generated evidence. In contrast to HerO, which relies on multiple open models fine-tuned for specific subtasks, the AIC CTU approach shows that a unified, prompt-driven RAG pipeline can deliver competitive results with minimal engineering overhead.

Beyond the three podium systems, several additional submissions showcased inventive variations on the AFC pipeline. *Dunamu-ML* [45] matched the top retrieval score by enriching the AVeriTeC knowledge store, their crawler extracts text from PDFs and YouTube transcripts, then feeds this evidence into a BM25, dense-rerank stack topped with GPT-4 for question generation and verdict prediction. *Papelo* [37] obtained the highest score among teams that opted out of the shared knowledge store, relying instead on live Google search. Its multi-hop strategy asks an initial query with T5-large, generates follow-up queries with GPT-4o, and iteratively retrieves and reasons, enabling strong performance despite noisier web inputs.

These systems underline two broad lessons: (i) creative retrieval methodologies including custom scraping, multi-hop querying, and hybrid sparse+dense rerankers, remains a major leverage point, and (ii) carefully orchestrated smaller or open models show clear promise in approaching GPT-4-level performance when paired with robust retrieval and prompt engineering, but fully harnessing this potential requires substantial further research into model architectures, retrieval-integration strategies, and prompt optimization.

2.4.3 Limitations

While these systems have achieved strong results on the AVeriTeC dataset, several limitations remain. The top-performing system, Infact [51], emphasizes transparency and modular reasoning, but it is computationally expensive, costing around \$0.46 per claim, due to its reliance on closed-source models like GPT-4o and the use of multiple stages (on average seven operations per claim). Additionally, the system’s six-stage pipeline introduces significant delays, making inference time both long and inconsistent. However, the second-best performing system, HerO [67], relies entirely on open-source LLMs, yet still demands substantial computational resources to achieve efficient inference. While it processes each claim in under one minute, this performance is only achieved using high-end hardware, specifically two H100 GPUs (80GB each) and 480GB of RAM. For 500 claims, the system takes approximately 6.6 hours in total, with the majority of time spent in the evidence retrieval stagwee (six hours), followed by question generation (25 minutes) and veracity prediction (12 minutes). This setup, while fast, significantly raises the operational cost and makes reproducibility more difficult.

Similar to Infact, AIC CTU [60] incurs nontrivial costs and resource requirements. It relies on GPT-4o for evidence & label generation, making around 9,000 API requests during experimentation at a total cost of \$363 (approximately \$0.04 per claim). While the authors demonstrate that Llama 3.1-70B can serve as an open-source alternative, running such a large model on premise also demands substantial GPU memory and inference time.

Beyond the podium systems, many submissions [45, 37] either depend on closed-source models, hindering reproducibility or employ open-source models that demand high-end GPUs to achieve reasonable throughput, and when run on more modest hardware they incur impractically long inference times.

2.5 Concluding Remarks

In this chapter, we reviewed early detection efforts, covering content-based methods that leverage linguistic and semantic patterns and context-enriched approaches that incorporate external evidence. We highlighted how content-based models advanced performance yet struggle with domain shifts and emerging claims. Context-enriched systems improve robustness by retrieving or integrating external sources, but often rely on static corpora, manual curation, or costly pipelines. These limitations highlight the importance of end-to-end Automated Fact-Checking (AFC) frameworks that can dynamically retrieve diverse evidence, operate efficiently with open or lightweight models, and adapt to evolving information. The insights from this survey motivate the design of our proposed AFC architecture in the next chapter, aiming to balance accuracy, timeliness, and resource constraints in real-world fact-checking scenarios.

Chapter 3

Methodology

3.1 Overview

In this chapter, we introduce our fact-checking system, *Fathom*, developed in direct response to the limitations outlined in Section 2.4. Building on these insights, *Fathom* is designed to operate under realistic time constraints while leveraging lightweight, open-source language models. This makes the system reproducible, scalable, and easier to deploy across diverse operational settings.

Fathom draws inspiration from prior fact-checking systems such as InFact [51] and HeRO [67], both of which emulate core aspects of the manual fact-checking workflow. While building on these foundations, our system introduces several key innovations aimed at improving efficiency and performance across the entire pipeline. A major novelty of *Fathom* lies in its approach to claim decomposition and veracity classification. Unlike existing systems that rely heavily on large, resource-intensive models and lengthy inference times, *Fathom* uses compact, local language models to perform these tasks efficiently decomposing and labeling claims in under 30 seconds. This enables rapid, low-resource operation without sacrificing output quality. At

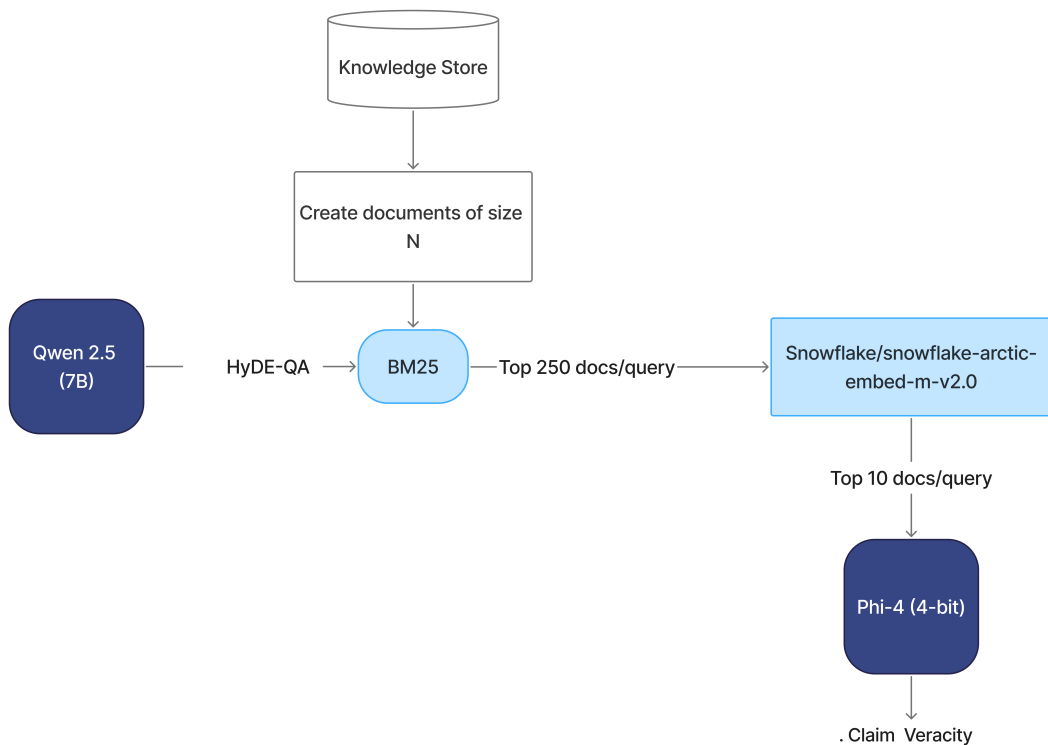


Figure 3.1: Overview of the Fathom system architecture.

the claim decomposition stage, we further enhance model reasoning by employing a few-shot prompting strategy using Conflicting/Cherry-picking examples. This approach encourages the model to generate sub-questions that reflect both supporting and refuting perspectives, resulting in a more balanced and comprehensive evidence-seeking process. By guiding the model in this way, we effectively emulate the critical thinking strategies of human fact-checkers.

As shown in Figure 3.1, our system follows a four-stage pipeline. The first stage begins with HyDE-style question generation, in which a lightweight open-source LLM expands the context around each claim by generating plausible question-answer pairs. In the second stage, we employ a two-step retrieval strategy: an initial sparse retrieval using BM25, followed by semantic reranking based on embedding similarity to surface the most relevant evidence. Finally, in the last stage, the retrieved evidence

is passed to another lightweight open-source LLM, which generates both a supporting rationale and a final veracity verdict.

We evaluate *Fathom* on two datasets. The first is the AVeriTeC dataset [54], which includes a curated knowledge store, allowing for controlled experimentation without the variability of live web search. Each claim in AVeriTeC is annotated with one of four veracity labels: Supported, Refuted, Not Enough Evidence (NEE), or Conflicting / Cherry-picking (where the evidence contains both support and contradiction, or the claim selectively misrepresents facts). To address some of the limitations observed in AVeriTeC, we also construct our own dataset, which will be introduced and analyzed in detail in later chapters.

In the sections that follow, we present a detailed walkthrough of each stage in the *Fathom* pipeline, outlining the key technical components, design considerations, and implementation strategies that drive the system.

3.2 Claim Decomposition

In this section, we describe the methodology used to enhance the contextual understanding of claims, an essential step that significantly improves the relevance of evidence retrieved by our system. To achieve this, we draw on two complementary strategies: Hypothetical Document Embeddings (HyDE) and question-answer (QA) generation.

HyDE, first proposed by Gao et al. [23], is a technique for enriching query representations in retrieval-augmented generation (RAG) systems. It works by generating a plausible, hypothetical document in response to a given query, in our case, the claim and then encoding this generated passage to produce a more nuanced, context-rich embedding. This embedding captures not only the surface-level content of the claim

but also its underlying semantic intent, thereby improving the retrieval of semantically relevant documents. Recently, HyDE has been successfully incorporated into fact-checking systems. For instance, the HeRO system [67] uses a synthetic passage generation step to enrich the claim context before retrieval. These synthetic passages, when used as search queries, enhance retrieval performance by embedding latent contextual signals that go beyond the original wording of the claim.

In parallel, several other fact-checking pipelines [46, 51, 60] have demonstrated the effectiveness of question generation as a means of structuring the verification process. These systems generate explicit questions from claims and attempt to answer them using retrieved evidence. Such question-answer pairs help not only with more targeted retrieval but also with organizing information in a way that supports stronger reasoning by downstream language models.

Inspired by these complementary lines of work, we propose a unified approach that combines both: Hypothetical Document Embedding with Question-Answer generation (HyDE-QA). For each claim, we generate multiple plausible HyDE-style question-answer pairs that simulate what a fact-checker might ask and answer while verifying the claim. These QA pairs are then used to drive evidence retrieval, enriching the claim’s context from multiple angles, both semantically and structurally. This combined method allows our system to retrieve higher-quality evidence and provides a more coherent basis for downstream reasoning and classification.

In order to generate the question–answer pairs, a lightweight open-source large language model (LLM) is prompted with the claim, along with optional metadata such as the speaker, date, and location, and instructed to produce multiple detailed QA pairs. To steer generation quality, we incorporate carefully selected few-shot examples that emphasize well-reasoned, structured question–answer formats based on the nature of the claim. Few-shot prompting—popularized by Brown et al. [19] in

You are a fact-checking assistant.
The given claim falls into one of the following four categories.

1. Supported
2. Refuted
3. Not Enough Evidence (if there isn't sufficient evidence to either support or refute it)
4. Conflicting Evidence/Cherry-picking (if the claim has both supporting and refuting evidence)

Assume that the claim is a label from the 4 categories and then provide evidence in the form of question-answer pairs. You will be given optional metadata such as speaker, date etc, you can use them if needed.

IMPORTANT: Follow these rules:

1. State every single question-answer pair in a way that it can be understood independently and without additional context. Therefore, be explicit and do not use pronouns or generic terms in place of names or objects.

Examples:
Claim: Over 700,000 children [accessed] early childhood education in the last financial year
Speaker: Cyril Ramaphosa
Date: 7-2-2019
Location: ZA
Label: Conflicting Evidence/Cherry-picking

QA:
Q1: Did Cyril Ramaphosa underestimate the number of children in early childhood development facilities in South Africa?
A1: Yes. Ramaphosa put the number at “nearly one million children” in 2018. Then in 2019, his figure was “over 700,000”. Both figures are an underestimate.

Figure 3.2: HyDE prompt with few shot samples

the original GPT-3 paper has proven effective in guiding LLMs to perform complex reasoning tasks by providing a small number of illustrative examples.

In our setting, these few-shot examples act as structured templates, demonstrating the kind of claim decomposition we expect. They guide the LLM to turn claims into meaningful, targeted questions and to generate plausible answers that simulate the reasoning of human fact-checkers. Notably, these examples are handpicked from the AVeriTeC training dataset, allowing the model to internalize domain-specific

patterns of reasoning and question formulation commonly found in that dataset.

While these examples serve purely as prompts and are not seen during evaluation, it is important to note that their alignment with the AVeriTeC dataset likely contributes to the system’s strong performance on that benchmark. As such, performance may vary when applied to other datasets with different characteristics, underscoring the need to adapt or diversify few-shot examples for broader generalizability across fact-checking domains.

We incorporate eight few-shot examples, six of which correspond to claims labeled as Conflicting Evidence/Cherry-picking. We intentionally focus on this category because, after a careful review of the AVeriTeC dataset, we observed that these claims often give rise to both supporting and refuting question-answer (QA) pairs. As a result when we add more samples from this category, they would naturally encourage the model to explore multiple facets of a claim, promoting more balanced and comprehensive reasoning. The remaining few-shot examples include one Supported and one Refuted case. An example of our prompt is shown in Figure 3.2. For brevity, the figure does not display all eight few-shot examples or the full set of QA pairs. We use this prompt to condition a lightweight LLM, appending the input claim after the examples. The model then generates question-answer pairs based on the claim, as illustrated in Figure 3.3.

The QA pairs generated in this step are then passed to the next stage, where they are used to guide the evidence retrieval process.

3.3 Evidence Retrieval

To retrieve evidence effectively, we employ a hybrid strategy that combines BM25-based sparse retrieval [50] with dense semantic reranking via vector similarity [32, 36].

<p>Claim: In a letter to Steve Jobs, Sean Connery refused to appear in an Apple commercial.</p> <p>Q1: Did Sean Connery refuse to appear in an Apple commercial?</p> <p>A1: No, because in 1998, Steve Jobs reached out to Sean Connery to star in a commercial for Apple’s iMac G3. Connery agreed and filmed the commercial.</p> <p>Q2: Was there a letter from Sean Connery refusing to appear in an Apple commercial?</p> <p>A2: No, there is no record of such a letter. Instead, Connery accepted the invitation and starred in the commercial.</p> <p>Q3: Did Steve Jobs request Sean Connery for a commercial?</p> <p>A3: Yes, in 1998, Steve Jobs, the CEO of Apple, personally wrote to Sean Connery, asking him to star in a commercial for the iMac G3.</p>
--

Figure 3.3: Example of generated QA pairs for a Refuted claim.

This design is motivated by recent fact-checking and RAG systems such as AICTU [60], HeRO [67], and Anthropic’s Contextual Retrieval proposal [11] which demonstrate that integrating both lexical matching and semantic signals leads to more comprehensive and precise evidence retrieval. In the subsequent subsections, we detail the implementation and configuration of both the components in our pipeline.

3.3.1 Sparse Retrieval (BM25)

BM25 builds upon the TF-IDF (Term Frequency–Inverse Document Frequency) concept, where TF-IDF measures how important a word is to a document within a collection. BM25 refines this by incorporating document length normalization and applying a saturation function to term frequency, preventing very common words from disproportionately dominating retrieval scores [50]. For example, consider a user querying “Error code TS-999” in a technical support database. An embedding-based retriever might surface general information about error codes but could miss documents containing the exact “TS-999” string. In contrast, BM25 directly matches the specific text, ensuring that documentation explicitly mentioning “TS-999” is retrieved. Additionally, in the AVeriTeC dataset each claim comes with its own knowl-

edge store, and each knowledge store consists of hundreds of thousands of webpages, making direct dense retrieval prohibitively expensive. Thus the usage of BM25 would considerably narrow down the candidate pool, and help *Fathom* stay in the time and resource constraints of fact-checking a claim under one minute.

Our system begins by splitting each document associated with a given claim into smaller passages: consecutive sentences are grouped until their combined character length reaches a threshold N . Each chunk is annotated with metadata linking it to its preceding and following chunks, and includes the source URL to preserve provenance when presented to the LLM. We then apply BM25 retrieval [50] to narrow down relevant passages. For each generated question-answer pair, we form a BM25 query and retrieve the top $k = 250$ chunks. For example, with 5 QA pairs we retrieve up to 1,250 candidates, which are then deduplicated to ensure uniqueness. We choose to retrieve 250 chunks per QA pair as a balance between performance and efficiency, increasing this to 500 chunks yielded negligible improvements in results while significantly slowing inference, whereas reducing the number below 250 degraded performance without meaningful speed gains.

3.3.2 Semantic Re-ranking

Semantic re-ranking serves to refine the BM25-filtered candidate pool by leveraging dense embeddings to capture deeper contextual matches that sparse retrieval may miss. After the initial BM25 stage reduces the search space, semantic re-ranking encodes both queries and candidate chunks into vector representations and scores them by similarity, ensuring that passages with strong semantic alignment to the query are prioritized. By focusing computationally heavier embedding comparisons on a smaller set of candidates, we maintain throughput while improving the quality

of evidence passed to downstream reasoning modules.

Following this general approach, we select an embedding model under 500M parameters using the Massive Text Embedding Benchmark (MTEB) benchmark [42]. We choose Snowflake/snowflake-arctic-embed-m-v2.0 [40], a compact model that ranks highly on retrieval tasks in MTEB, fitting our runtime and memory constraints. Each candidate chunk is encoded into a dense vector \mathbf{d} , and each query formed by concatenating the question and answer from a QA pair is encoded into \mathbf{q} .

We use cosine similarity to measure semantic similarity between the query vector \mathbf{q} and the document vector \mathbf{d} . Cosine similarity captures the angle between two vectors, emphasizing their directional alignment rather than their magnitude. This is particularly well-suited for text embeddings, where the direction of the vector represents semantic content. Although we considered Euclidean distance as an alternative, we found it less appropriate in this context. Euclidean distance accounts for both magnitude and direction, which can penalize sentences of different lengths or scales, even when they are semantically similar. For example, if two sentences convey the same meaning but one has more words or higher embedding magnitudes, Euclidean distance may assign a larger distance, thereby underestimating their similarity. In contrast, cosine similarity is invariant to vector magnitude and has been shown to work effectively with sentence embeddings. Moreover, the embedding model we selected Snowflake/snowflake-arctic-embed-m-v2.0 is evaluated on benchmarks such as MTEB [42] using cosine similarity as the standard metric. This alignment between model evaluation and retrieval scoring further supports our choice of cosine similarity for identifying semantically relevant chunks.

We compute cosine similarity using Equation 3.1, selecting the top 10 highest-scoring chunks for each QA pair. This cutoff was chosen based on manual inspection, which revealed that beyond the top 10 chunks, the semantic similarity with the QA pair

dropped significantly, providing little additional value.

$$\text{sim}(\mathbf{q}, \mathbf{d}) = \frac{\mathbf{q} \cdot \mathbf{d}}{\|\mathbf{q}\| \|\mathbf{d}\|} \quad (3.1)$$

The result is a ranked list of evidence passages per QA pair. For example, if we had 5 QA pairs with initially synthetic answers, the retrieved evidence for each pair is ideally expected to align with and support the question. Additionally, we record the similarity score of each evidence chunk with its respective QA pair to assess the relevance and quality of the retrieved evidence.

At this stage, we have generated a set of questions for each claim, which aids *Fathom* in decomposing the original claim. For each question, we have retrieved 10 corresponding evidence chunks. This structured data, comprising decomposed questions and their top-ranked evidence will now be utilized in the final veracity prediction stage.

3.4 Veracity Prediction

In the final stage, the system has to assign one of the four labels (Supported, Refuted, Not Enough Evidence, or Conflicting/Cherry-picking) based on the structured evidence retrieved in the previous step, along with a clear rationale for its decision. We prompt a lightweight open source LLM using a structured prompt that includes:

- the original claim
- a set of generated questions (from the QA generation step)
- For each question, up to $N = 8$ top-ranked evidence passages (ranked by the similarity score)

Through our preliminary testing, we found that $N = 8$ strikes a balance between sufficient context and avoiding prompt length limitations or information overload. Furthermore we notice a significant improvement when we prompt the LLM perform structured reasoning, by employing a Chain-of-Thought (CoT) strategy, wherein the LLM reasons through the evidence provided step-by-step before providing a final verdict.

Before going ahead we also want to expand on the idea of chain-of-thought reasoning strategy we have employed. We draw inspiration from prior work [65] which demonstrates that chain-of-thought enhances the reasoning capabilities of large language models across various tasks. Chain-of-thought essentially means that there are a sequence of intermediate natural-language reasoning steps before arriving at the final answer, in simpler words, encouraging the model to "think" through sub-problems rather than directly outputting a conclusion. Subsequent work [34] has shown that even without few-shot examples, appending a simple instruction such as "Let's think step by step" can induce a zero-shot chain-of-thought effect, significantly improving performance on diverse reasoning tasks.

The prompt we used in our veracity prediction step is shown in Figure 3.4, which essentially shows how we ask the LLM to reason through the evidence provided. Furthermore, an excerpt of the final-stage output is shown in Figure 3.5, where the LLM can be seen reasoning through the provided questions and their corresponding evidence passages.

3.5 Concluding Remarks

In this chapter, we introduced *Fathom*, the central system proposed in this thesis. We outlined its four core modules and discussed the key technical decisions and

Claim: You are a fact-checking helpful assistant.

Task: Your task is to evaluate the truthfulness of a claim using associated question–answer (QA) evidence pairs, where each question has several pieces of evidence (answers). You must analyze the claim and reason step-by-step through the evidence provided. Use a chain-of-thought reasoning approach to determine the final label.

The given claim falls into one of the following four categories:

1. Supported
2. Refuted
3. Not Enough Evidence
4. Conflicting Evidence/Cherry-picking

Input Format:
Claim: <claim>
QA: <Question answer pairs>

Output:
Reasoning: [Use chain-of-thought reasoning on the claim based on the evidence. Incorporate evaluation of the content and optionally consider the trustworthiness or context of the source URLs.]
Label: <Supported, Refuted, Not Enough Evidence, Conflicting Evidence/Cherry-picking>

Figure 3.4: Prompt for our veracity prediction step.

implementation steps within each. In the following chapters, we will delve into the LLMs, the techniques and the programming strategies employed which allowed *Fathom* to operate efficiently under a stringent time and resource constraint.

Claim: In a letter to Steve Jobs, Sean Connery refused to appear in an Apple commercial.

Reasoning: To evaluate the claim that "In a letter to Steve Jobs, Sean Connery refused to appear in an Apple commercial," we need to examine the evidence provided in the QA pairs.

Q1: Did Sean Connery refuse to appear in an Apple commercial?

- **Answer 1** references multiple sources discussing a supposed rejection letter from Sean Connery to Steve Jobs, suggesting that such a letter exists. However, these sources are primarily linked to discussions about a viral letter rather than confirming its authenticity.
- **Answer 2** and **Answer 3** provide context about Apple's attempt to create a celebrity Christmas ad featuring Sean Connery, but they emphasize that the ad was never completed due to its poor quality. They do not confirm the existence of a refusal letter.
- **Answer 4** describes a letter that was widely believed to be authentic but later identified as a hoax. This suggests that while the letter was circulated, it was not genuine.

(Q2–Q5 omitted for brevity. They further support the same conclusion.)

Conclusion: The evidence consistently indicates that the letter from Sean Connery to Steve Jobs was a hoax and not a genuine refusal. While Steve Jobs did request Connery for a commercial, Connery did not actually respond with a refusal. Therefore, the claim is refuted.

Label: Refuted

Figure 3.5: Example of LLM-generated chain-of-thought reasoning with partial QA evidence (Q1–Q2 shown).

Chapter 4

Datasets

4.1 Overview

In recent years, researchers have released a wide range of fact-checking datasets to curb the spread of misinformation (see Table 4.1). Broadly, these resources fall into two categories. Synthetic datasets such as FEVER [58] and FEVEROUS [10] generate claims by modifying reliable Wikipedia sentences, providing large-scale, well-structured training material. Real-world datasets, by contrast, collect naturally occurring claims that have already been fact-checked. Early efforts like MultiFC [12] brought together claims from 26 fact-checking outlets but often relied on context that was not self-contained. More recently, AVeriTeC [54] pushed the field forward with 4,568 claims vetted by 50 organisations and accompanied by web evidence and fine-grained rationales. One major limitation of the AVeriTeC dataset is the temporal mismatch between when claims were made (1919–2020) and when evidence was retrieved (early 2024), often long after the claim’s original context. This gap risks introducing evidence that may not reflect what was available or relevant at the time, potentially skewing veracity assessments. Additionally, AVeriTeC lacks strict

enforcement of source credibility; despite annotator guidance, the dataset includes at least 85 documents from known misinformative sites, and the provided blacklist is incomplete. This undermines reliability, as low-credibility sources may be treated equivalently to authoritative ones.

Table 4.1: Fact checking datasets

Dataset	Dataset Type	Domain	#Claims
FEVER [58]	Synthetic	Multiple	185,445
FEVEROUS [10]	Synthetic	Multiple	87,026
HoVer [31]	Synthetic	Multiple	26,171
AveriTEC [54]	Real-world	Multiple	4,568
XFact [27]	Real-world	Multiple	31,189
MultiFC [64]	Real-world	Multiple	36,534
Liar [12]	Real-world	Politics	12,836

Lastly, in order to fully test our system, we also wanted to test *Fathom* on real-world recent claims. Thus we introduce FactCellar. It is a dataset that includes the most recent claims collected between 2021 and 2024, along with its corresponding evidence gathered from various sources. In addition to the evidence, it provides valuable metadata, such as the source credibility of evidence sources and an impact analysis of the claims. Our system *Fathom* also sets a baseline for future systems using the dataset.

In this chapter, we present the datasets on which we evaluated our system, *Fathom*. One of the primary datasets used is the AVeriTeC dataset, which served as the initial benchmark for testing and developing our pipeline. Subsequently, to address the limitations of existing datasets and test our system on recent real-world claims we constructed our own dataset, FactCellar, which will be discussed in detail in the following sections.

4.2 AVeriTeC dataset

As discussed before, we initially develop and test our system on the AVeriTeC dataset [54], a resource for automated fact-checking containing 4,568 real-world claims collected from 50 professional fact-checking organizations. Each claim is annotated with a veracity label *Supported*, *Refuted*, *Not Enough Evidence*, or *Conflicting Evidence/Cherry-picking* along with question-answer (QA) pairs grounded in web-based evidence, and a justification explaining how the evidence supports the verdict. The distribution of labels across the training and development splits is summarized in Table 4.2.

The AVeriTeC dataset uses a multi-step annotation process to make each claim easier to verify. First, the claims are cleaned and simplified for clarity. Then, annotators generate questions that reflect the core factual components of the claim. For each question, they retrieve supporting or refuting information from the web and record multiple answers along with the source URLs.

In the 2024 AVeriTeC shared task Schlichtkrull et al. [52] introduced a knowledge store, a curated collection of pre-retrieved web documents, to assist people using the dataset in evidence retrieval, eliminating the need for independent web scraping. In order to evaluate the systems which were submitted in the shared task, Schlichtkrull et al. also released a test set in 2024 [53] which consisted of 1000 carefully curated claims. Importantly, this test set was distributed without accompanying veracity labels. Once a system was developed using the training and development sets, it was submitted to the AVeriTeC shared task organizers, who evaluated it on the test set using their own evaluation metric, called the *AVeriTeC score*, which we will discuss in the next section.

Table 4.2: Class-wise distribution of train and dev sets of the AVeriTeC dataset.

Class	Train	Dev
Supported	847	122
Refuted	1743	305
Conflicting evidence/Cherrypicking	196	38
Not enough evidence	282	35
Total	3068	500

4.2.1 AVeriTeC Score

In the 2024 AVeriTeC shared task, participants were required to generate questions (Q) and question-answer pairs (Q+A) for each claim, with evaluation based on the Hungarian METEOR score to assess how closely generated Q+A matched reference Q+A. METEOR provides a simple similarity measure by matching words, stems, and synonyms while penalizing missing or extra content [15]. The Hungarian algorithm is then used to find the optimal one-to-one alignment between up to the first 10 generated Q+A pairs and the reference Q+A pairs, maximizing the total METEOR similarity across all matches [35]. A Q+A similarity score is computed as the average METEOR over matched pairs (using the Hungarian algorithm to align up to the first 10 generated Q+A pairs with the reference Q+A set), and then compared against a cutoff of 0.25, only when the resulting average METEOR score meets or exceeds 0.25 is the evidence considered adequate and the veracity prediction evaluated, ensuring that only well-aligned, evidence-backed QA contributes to the final *AVeriTeC score*.

However, this method of calculating the *AVeriTeC score* had some obvious limitations. It relies on a fixed METEOR cutoff that may misclassify semantically correct but lexically divergent QA pairs, potentially excluding valid evidence. Essentially, it is based on surface-level word overlap and may fail to capture deeper semantic equivalence in QA content, which can cause genuinely supportive evidence to be disregarded if phrased differently from the reference. This limitation was also high-

lighted by Rothermel et al.’s InFact system [51], submitted to the AVeriTeC Shared Task 2024, which reported lower Hungarian METEOR scores despite generating Q+A pairs that were semantically aligned with the reference. This shortcoming motivated the move to a finer-grained, fact-centric evaluation in 2025.

In 2025, the *AVeriTeC score* transitioned from Hungarian METEOR to the Ev2R atomic reference scorer [9]. Rather than comparing whole QA pairs by surface similarity, Ev2R first uses an LLM to break both the predicted and reference Q+A pairs into sets of “atomic facts”. The metric then measures $Q + A$ (Ev2R Recall) as the fraction of these reference atomic facts for which there is matching, evidence-supported information in the prediction. In simpler terms, one can think of listing out the key statements in the reference answer and checking off which of them the retrieved evidence actually confirms. To ensure reliability, the predicted veracity label is only evaluated and considered when at least half of these key statements (Recall > 0.50) are covered by the retrieved evidence. If fewer than 50% of the atomic facts are supported, the evidence is deemed insufficient and the claim is not scored. This threshold ensures that a majority of the core facts must be grounded in evidence before trusting a veracity decision. Overall, the Ev2R-based AVeriTeC score provides a more semantically precise and robust way to link evidence retrieval to veracity prediction by focusing on the coverage of individual atomic facts rather than on surface text overlap.

4.3 FactCellar

Fact-checking datasets which have been proposed in the past years such as the AVeriTeC [54] or FEVEROUS [10] have played a crucial role in training and evaluating fact-checking models, enabling the development of more sophisticated and

scalable verification pipelines. However, a major limitation of these datasets is that they are constrained by the context and credibility of sources available at the time of collection. In many cases, the supporting or refuting evidence comes from a wide range of web sources, some of which may be outdated, misleading, or lack authority. This poses a challenge for fact checking systems that treat all sources equally without assessing their reliability. For instance, when verifying a statistical claim about a country, the most trustworthy evidence would typically come from official sources, such as government (.gov) websites. As such, domain-based cues like top-level domains (TLDs) can serve as important indicators of credibility and should be weighted accordingly during the retrieval and reasoning stages of automated fact-checking. To address these limitations, we introduce FactCellar a novel dataset that pairs real-world claims with web-retrieved evidence and other metadata while explicitly incorporating source credibility.

FactCellar dataset consists of 5,145 real-world claims sourced from PolitiFact and Snopes, each accompanied by corresponding evidence. Additionally, each claim is annotated with source credibility information, including features such as top level domain (TLD), external credibility ratings, factuality ratings, political bias scores, and domain age, allowing models to assess the reliability of attached evidence more effectively. We also add an impact score with an accompanying justification for each claim. This metadata is generated using a structured prompt to an LLM that evaluates potential harm, emotional provocation, topic sensitivity, and broader societal implications.

In the following subsections, we first detail our data collection methodology covering claim sourcing from PolitiFact/Snopes and web evidence retrieval and then describe our annotation procedures for source credibility and impact scoring. Finally, we present an in-depth analysis of the resulting FactCellar dataset which highlights its characteristics.

4.3.1 Data Construction

We collected claims from two prominent fact-checking resources: PolitiFact [3] and Snopes [5], comprising 2,557 and 2,588 claims, respectively. To maintain relevance and ensure the availability of evidence, we restricted our data extraction to claims published from the year 2021 to 2024. The initial metadata provided by the PolitiFact dataset included the following fields: *Claim*, *Claim Author*, *Claim Source*, *Claim Date*, *Fact Check Date*, *Justification*, *Issue*, *Fact-checking Sources*, and *Label*. In contrast, the Snopes dataset initially comprised a limited subset of these metadata fields, specifically: *Claim*, *Fact Check Date*, *Justification*, *Fact-checking Sources*, *Issue*, and *Label*.

To ensure consistency and alignment across both datasets, additional annotation was necessary for the Snopes claims. For this purpose, we first scraped the original textual content associated with each Snopes claim. Subsequently, we used Google’s Gemini 2.0 Flash Experimental [26] model to systematically extract and annotate the missing metadata fields, namely *Claim Author*, *Claim Source*, *Claim Date*, and *Justification*. This annotation step was necessary to align and unify both datasets. To improve extraction accuracy, we manually inspected up to 50 examples and iteratively refined our prompting strategy encouraging the model to return "N/A" when metadata was absent, and providing clear examples to guide its behavior. Despite these precautions, we acknowledge the possibility that some outputs may still contain hallucinated or imprecise data generated by the model. After constructing the combined dataset consisting of 5,145 claims, several additional processing steps were taken to enhance the dataset, namely gathering an evidence knowledge store, extracting auxiliary metadata such as source credibility, and claim impact assessment. The detailed process is described below:

1. **Claim Decomposition:** This process involves breaking down each claim into

a set of fine-grained, individually verifiable facts [56]. The primary idea behind claim decomposition is to convert these granular facts into effective search queries, facilitating the retrieval of precise and relevant information when constructing the knowledge store. To achieve accurate and efficient decomposition, we leverage Gemini 2.0 Flash experimental, which provides nuanced, context-aware decomposition of complex claims. Specifically, we utilize, employing few-shot examples to guide the Large Language Model in generating accurate and consistent decompositions.

- 2. Knowledge Store Construction:** To construct the knowledge store, we leveraged the queries generated from the claim decomposition step. We utilized Serper [4] with these queries to retrieve the first page of search results, specifically capturing the top 10 results per query. For instance, if a single claim was decomposed into three distinct queries, we performed searches with each of these three queries as well as the original claim itself, resulting in four separate search queries. Consequently, we extracted a total of 40 Uniform Resource Locators(URLs) from these combined searches. Duplicate URLs were identified and removed to ensure that the knowledge store contains only unique URLs.

Next, URLs from social media platforms and well-known fact-checking websites such as Politifact, Snopes, and factcheck.org were filtered out. This step was taken in order to avoid retrieving articles directly addressing the claim itself. Finally, we crawled each remaining URL for every claim using Trifaltura [16], with headless Selenium as a fallback in case of failure, systematically populating the knowledge store with the retrieved content.

- 3. Domain Credibility:** To quantify the reliability of each source URL in our knowledge store, we compute a comprehensive set of credibility metrics:

- **Domain Age:** We determine domain age using WHOIS [6], reflecting the longevity and stability of the domain.
- **Top-Level Domain (TLD) Score:** A predefined mapping was created assigning authority scores based on the type of TLD (e.g., .gov, .edu, .com), reflecting perceived trustworthiness and authority. Highly authoritative domains (e.g., .gov, .mil, .edu) score highest, while free, low-cost and easy-to-obtain domains (e.g., .tk, .ml) score lowest.
- **PageRank:** PageRank values were retrieved using the OpenPageRank API [2], providing an indicator of the page’s significance based on web link structure.
- **Bias Rating, Credibility Rating, and Factual Rating Scores:** We scrape these metrics from Media Bias/Fact Check [1] for each URL when available. Each category has a specific numerical mapping, summarized as follows:
 - **Credibility Rating:** Low (1), Mixed (2), Medium (3), High (4)
 - **Factual Rating:** Very High (6), High (5), Mostly Factual (4), Mixed (3), Very Low (2), Low (1)
 - **Bias Rating:** Left (1), Left-Center (2), Center/Pro-Science (3), Right-Center (4), Right/Conspiracy/Fake News (5)

Finally for each claim, we aggregate the source credibility metrics across all URLs, computing an average arithmetic score. This source credibility score serves as an indicator of the overall trustworthiness and reliability of the evidence supporting the claim.

4. **Impact Analysis:** For each claim, we use Gemini 2.0 Flash experimental to assess its potential impact, if widely believed and disseminated. Specifically, Gemini 2.0 Flash Experimental generates an *impact score* (ranging from 1,

minimal impact, to 10, very high impact) along with an *impact justification*. The prompt as shown in Figure 4.1 includes clear evaluation criteria such as potential harm, topic sensitivity, emotional provocation, and broader context to ensure consistency in assessment, supported by few-shot examples.

4.3.2 Dataset Analysis

4.3.2.1 Snopes

Snopes is one of the two datasets used in our analysis, comprising 2,588 claims spanning from January 2022 to September 2024. The distribution of claims across different labels is as follows:

- **True:** 847 (True), 53 (Mostly True)
- **False:** 1,657 (False), 31 (Mostly False)

The dataset addresses 63 distinct issues, including areas such as science, crime, climate change, abortion, health, and education. The top five most frequent issues are entertainment, politics, news, history, and sports.

The average number of queries per claim is 3.78, with a maximum of 14 queries.

Impact Scores We assess the impact scores for each label category by averaging the scores across all claims assigned to that label. The statistics are as follows:

- **True:** Mean = 4.68, Min = 1, Max = 9
- **False:** Mean = 5.27, Min = 1, Max = 10

The histograms in Figure 4.2 and Figure 4.3 illustrate the distribution of impact scores (ranging from 1 to 10) for claims categorized as True and False, respectively. False claims tend to have a slightly higher average societal impact than true claims, though the gap is less pronounced than in PolitiFact. True claims show a split distribution, with many being low-impact (around score 3) and others moderately impactful (scores 6–7).

4.3.2.2 Politifact

Politifact is the second dataset used in our analysis, comprising 2,557 claims made between January 2021 and September 2024. The distribution of claims by label is as follows:

- **True:** 117 claims
- **Mixed:** 203 (Mostly True), 295 (Half True), 415 (Mostly False)
- **False:** 1,214 (False), 313 (Pants on Fire)

The dataset covers 30 distinct issues, including topics such as environment, military, drugs, animals, and housing. The top five most prevalent issues are coronavirus, economy, crime, healthcare, and education.

The average number of queries per claim is 3.67, with a maximum of 18 queries.

Impact Scores Similar to Snopes, we calculate impact scores for each label category, summarized as follows:

- **True:** Mean = 6.03, Min = 2, Max = 9
- **Mixed:** Mean = 6.85, Min = 2, Max = 10

- **False:** Mean = 7.26, Min = 1, Max = 10

Similar to Snopes, we plot histograms for Politifact as well, as shown in Figure 4.4, Figure 4.6, and Figure 4.5, for claims categorized as True, Mixed, and False, respectively. On Politifact, False claims appear to be more impactful on average than both true and mixed claims, with a significant concentration of high impact scores peaking around 8. In contrast, True claims, while still impactful, tend to exhibit a more moderate distribution and rarely reach the highest levels of the impact scale. The Mixed category falls between the two, showing an impact score distribution that is generally higher than True claims but lower than False claims.

We generate this metric to analyze the extent to which misinformation contributes to societal disruption compared to accurate information. The higher mean impact score for false claims suggests that misinformation/disinformation often garners more attention, possibly due to its sensational or emotionally charged nature. This has important implications for fact-checking initiatives and content moderation strategies, as it highlights the need to prioritize and allocate resources to high impact claims, which essentially have more probability of being false. This also helps policymakers and researchers better allocate resources to combat the most harmful types of misinformation.

4.4 Concluding Remarks

In this chapter we have detailed the use of the AVeriTeC dataset as an initial benchmark for Fathom and motivated the creation of FactCellar to address limitations of prior resources by incorporating the recent real-world claims alongside source credibility metrics and impact assessments. By combining PolitiFact and Snopes data

with structured evidence retrieval, credibility scoring, and impact analysis, *FactCellar* enables a more robust evaluation of *Fathom*'s fact-checking ability. In the next chapter, we will present the experimental setup, evaluation metrics, and results of applying Fathom to both AVeriTeC and FactCellar, highlighting its performance, strengths, and areas for further improvement.

Task: Evaluate the societal impact of a given claim if it were widely believed and shared. You are provided only with the exact text of the claim.

Your response must include:

1. **Justification:** A short paragraph (2–3 sentences) explaining *why* you assigned that specific score (considering potential harm, topic sensitivity, emotional provocation).
2. **Impact Score:** A single integer from 1 to 10.

Impact Score Definitions:

- 1–2: Minimal-impact — negligible effect on public opinion or societal processes.
- 3–4: Low-impact — may slightly misinform or spark mild debate, unlikely to cause significant harm.
- 5–6: Moderate-impact — potential for moderate misinformation or controversy, possibly affecting health behaviors, local politics, or causing minor financial harm.
- 7–8: High-impact — could cause substantial harm, spark intense political or social conflict, or affect key institutions.
- 9–10: Very high-impact — extreme potential for widespread harm, severe financial or public health consequences, large-scale unrest, or major institutional damage.

Evaluation Criteria:

Consider the following when assigning a score:

- **Potential Harm:** Can it lead to health issues, financial loss, unrest, or reputational damage?
- **Topic Sensitivity:** Does it involve health, politics, social issues, or marginalized groups?
- **Emotional Provocation:** Does it amplify fear, anger, or outrage?
- **Wider Context:** How far-reaching could its influence be?

Scoring Guideline: If any single factor is especially extreme (e.g., incites violence, serious health risk), the score should be in the 7–10 range.

Examples:

Input: *Claim: "My homemade cookies are the best in the neighborhood."*

Output: { *Justification: "This is a benign, local claim with no societal impact."*, *Impact Score: 1* }

Figure 4.1: Prompt instructions for societal impact annotation task.

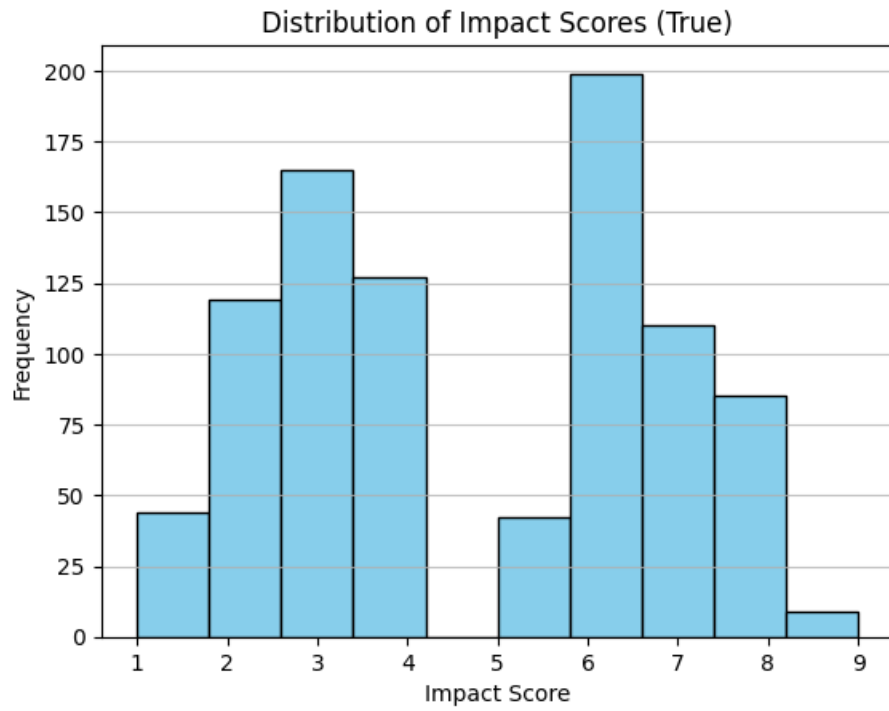


Figure 4.2: Snopes: Impact of True claims

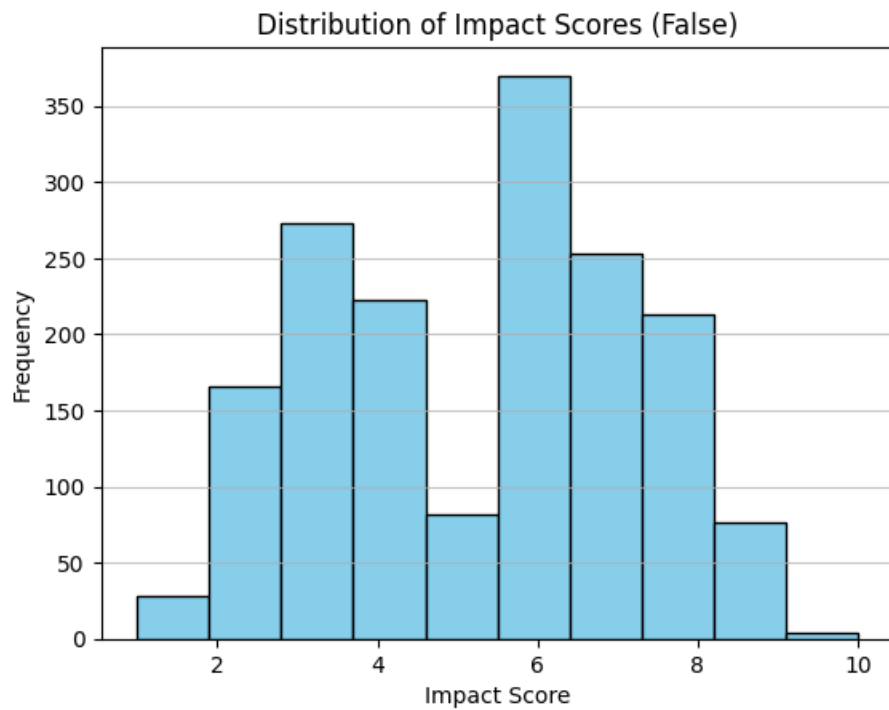


Figure 4.3: Snopes: Impact of False claims

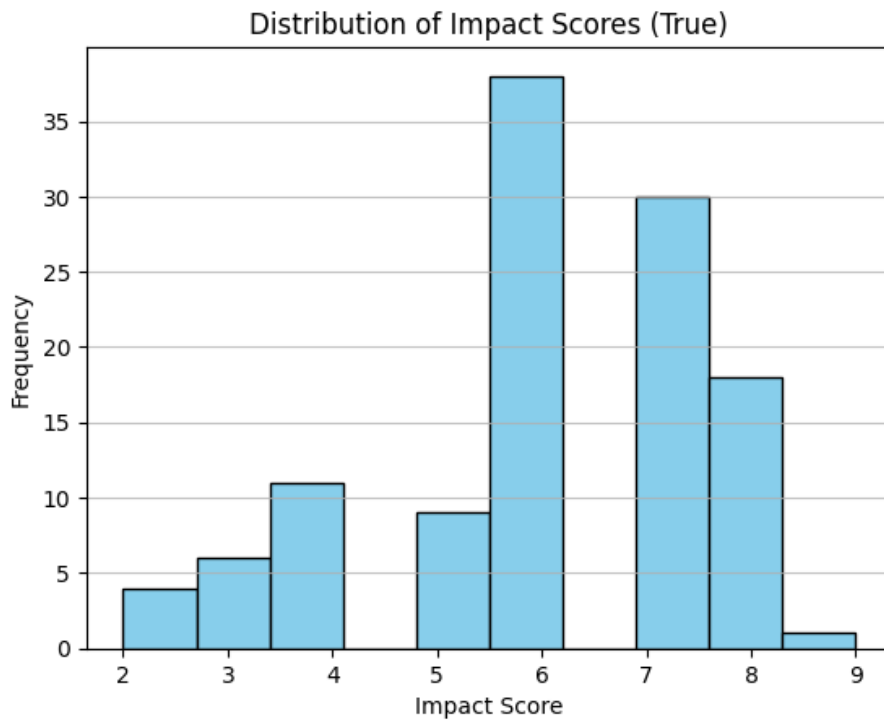


Figure 4.4: Politifact: Impact of True claims

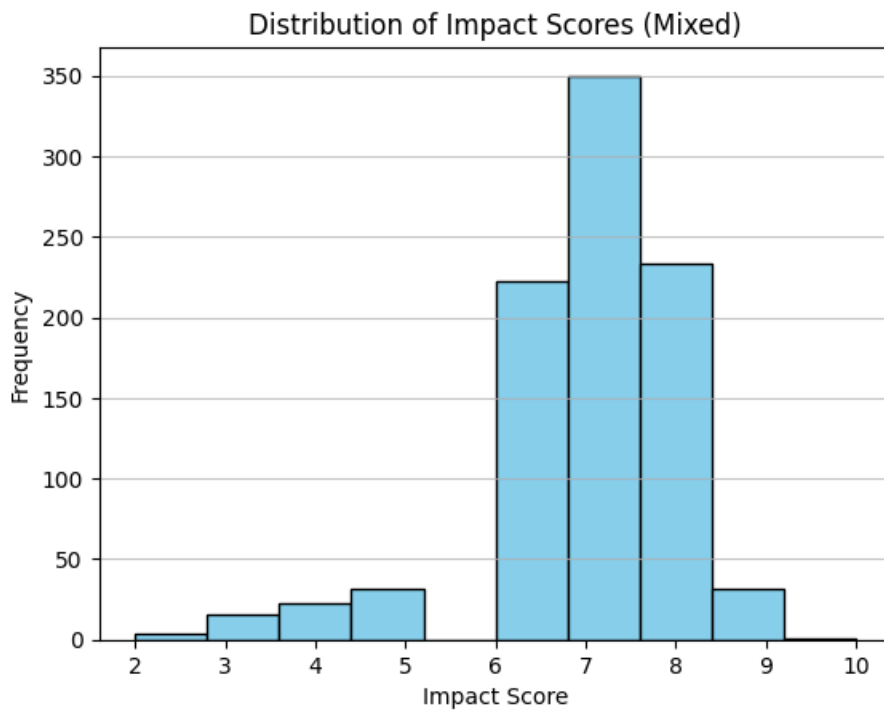


Figure 4.5: Politifact: Impact of Mixed claims

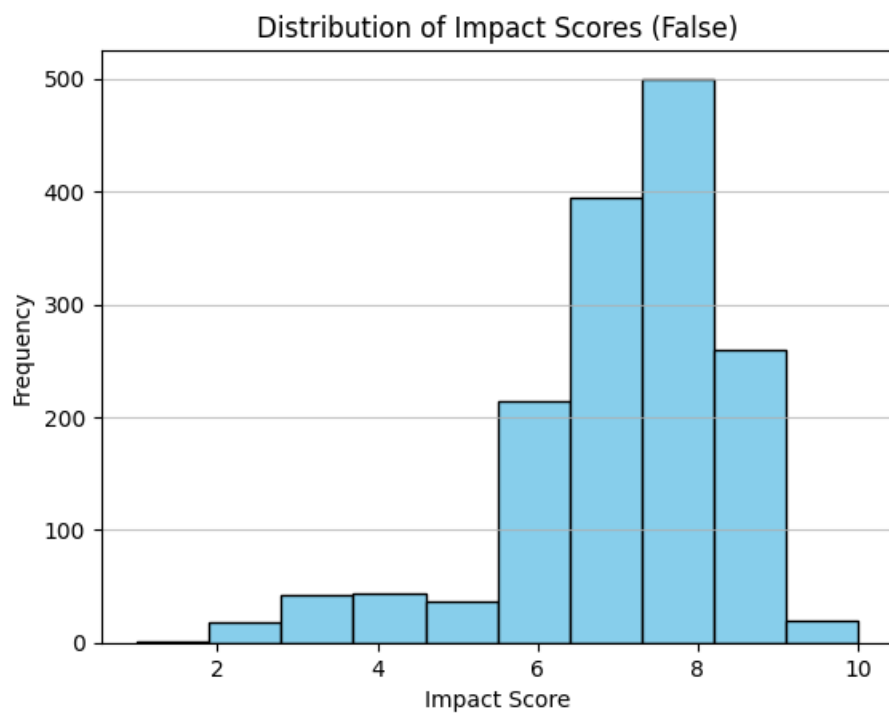


Figure 4.6: Politifact: Impact of False claims

Chapter 5

Experiments and Results

5.1 Overview

In this chapter, we present the experimental setup used to evaluate Fathom on both AVeriTeC and FactCellar. We also present the selection of Large Language Models (LLMs), prompt design and any other strategies. Furthermore, we outline the specific experiments conducted on each dataset, detailing the tasks. Finally, we summarize the main results, analyze improvements over baselines and discuss observed strengths and limitations, pointing to lessons learned and directions for future work.

5.2 Experimental Setup

All experiments were conducted using the NVIDIA Quadro RTX 8000 GPU, which provided ample capacity for both dense retrieval and large language model inference. The system was equipped with a 32-core processor (NUMA node0 CPU(s): 0–31) and approximately 125 GB of RAM. This configuration was particularly beneficial during the BM25 retrieval stage, where multiple processes could be executed concurrently

to accelerate indexing and scoring.

For the AVeriTeC test set, the experiment was conducted on an AWS EC2 g5.2xlarge virtual machine. This configuration included an NVIDIA A10G GPU with 23 GB of memory, 8 vCPUs, and 32 GB of RAM. The setup was provided and operated by the AVeriTeC team, as the test set labels were not publicly released. Throughout our system, we make design choices aimed at maximizing time efficiency without compromising output quality. In the first stage HyDE-QA step we use the open-source `Qwen2.5-7B-Instruct` [57] model. To reduce latency and GPU memory load, claims are processed with a batch size of 2. In the second stage, we apply parallelism to the BM25-based sparse retrieval step. Claims are distributed across multiple CPU processes, and queries within each claim are further parallelized using multi-threading. This dual-level parallelism allows the system to retrieve evidence for many claims simultaneously, significantly reducing total time.

For semantic re-ranking, we use the `Snowflake/snowflake-arctic-embed-m-v2.0` embedding model. This model was chosen based on its strong performance on the Massive Text Embedding Benchmark (MTEB), particularly on the Semantic Textual Similarity (STS) benchmark, a task that evaluates how effectively a model captures the semantic similarity between sentence pairs. At the time of system development, we specifically sought models with fewer than 500 million parameters, to ensure efficient inference and deployment in resource-constrained environments. We went for this model particularly because it supports input sequences up to 8,192 tokens, significantly exceeding the context window of many comparable models which supported input sequences of upto 1024 tokens. This extended input capacity allows for richer semantic representation of longer queries and documents, which is particularly valuable in our re-ranking pipeline. Its combination of compact size, high semantic accuracy, and long-context capability made it the ideal choice for our semantic re-ranking stage.

In the final veracity prediction step, we process claims in batches of 4 using a 4-bit quantized version of Microsoft’s Phi-4 model (jakiAJK/microsoft-phi4-GPTQ-int4) to ensure faster inference and reduced memory usage. We select Phi-4 for its advanced reasoning capabilities, as it outperforms much larger models including LLaMA-3.3 70B and GPT-4o-mini on several reasoning-focused benchmarks such as MATH, GPQA, and HumanEval [7].

5.3 Results

We conduct all development and initial evaluation on the AVeriTeC development set. As shown in Table 5.1, our system achieves a higher New (Ev2R) AVeriTeC score of 0.3780, outperforming the baseline. This gain is primarily driven by stronger semantic evidence retrieval, as reflected in the higher **Q+A (Ev2R)** recall score (quantifies the extent to which the retrieved evidence semantically matches the ground-truth reference evidence) of **0.5137**.

Table 5.2 reports the F1 scores for each class in the veracity classification task. Our final model, based on the 4-bit quantized **Phi-4**, performs well on the Refuted (0.8436) and Supported (0.6877) classes, which are also the most frequent in the development set. However, it struggles significantly with the underrepresented classes: Not Enough Evidence (NEI) (0.1455) and especially Conflicting/Cherry-picking (C/CP), where it fails to classify any instances correctly. This exposes a key limitation of the system in handling nuanced or ambiguous cases, where robust reasoning is required despite limited or contradictory evidence.

The model’s stronger performance on Supported and Refuted can be attributed to the nature of the evidence—these classes often contain explicit lexical cues that clearly indicate support or contradiction. In contrast, Not Enough Evidence is inher-

ently more difficult, as it requires recognizing when no sufficient evidence is present. Conflicting/Cherry-picking is even more challenging, as it involves evidence that is both supporting and refuting, demanding finer-grained analysis and contextual understanding. One idea we believe could improve performance though we were unable to implement it due to time constraints, is introducing a heuristic to directly assign the Not Enough Evidence label when the retrieved evidence cosine similarity falls below a certain relevance threshold. For the Conflicting/Cherry-picking class, computing the standard deviation of scores between individual evidence-QA pairs could help identify high variance situations that suggest conflicting interpretations. We also hypothesize that the use of a zero-shot prompting strategy in the veracity prediction stage contributes to the poor performance on NEI and Conflicting/Cherry-picking. Moving to a few-shot prompting setup, especially incorporating examples from these underrepresented categories, may help the model better generalize to ambiguous or low-resource cases.

Despite these limitations, the system achieves a strong overall accuracy of 72%, demonstrating reliable performance on majority classes while highlighting promising directions for future enhancement.

Table 5.1: AVeriTeC score on the **dev set**

	Q (Ev2R)	Q + A (Ev2R)	New (Ev2R)	Time/claim (s)
Fathom	0.2488	0.5137	0.3780	20
Baseline	0.3392	0.4404	0.2960	50

To evaluate the generalization of our system, we submit predictions on the unseen AVeriTeC test set, where ground-truth labels are hidden. As shown in Table 5.3, our system slightly outperforms (0.99%) the baseline while maintaining significantly lower average runtime per claim. However, in contrast to the substantial improvement observed on the dev set, the margin over the baseline on the test set is noticeably smaller.

Table 5.2: Per-class F1 scores, overall accuracy, and macro-averaged F1 score on the development set.

Class	F1 Score
Supported (S)	0.6877
Refuted (R)	0.8436
Not Enough Evidence (NEI)	0.1455
Conflicting/Cherry-Picking (CP/CE)	0.0000
Accuracy	0.7200
Macro Avg F1	0.4192

We hypothesize that this difference is because of our fixed chunking strategy (in step 2 of the pipeline), which segments documents using a constant token size without adopting to the semantic boundaries of the document. While this approach proved effective on the dev set, it may fail to generalize across more diverse or structurally varied claims in the test set. This is another direction for future work, to explore semantic aware chunking methods, potentially improving retrieval precision and final veracity scores.

Table 5.3: AVeriTeC Scores on the **test set**

	Q (Ev2R)	Q + A (Ev2R)	New (Ev2R)	Time/claim (s)
Fathom	0.1848	0.3368	0.2043	22.73
Baseline	0.2723	0.3362	0.2023	33.88

In addition to evaluating our system on the AVeriTeC dataset, we also assess its performance on the FactCellar dataset, which includes both the Snopes and Politifact subsets. The results are presented in Table 5.4. The system achieves an accuracy of 87% on Snopes, but only 68% on Politifact. We hypothesize that the lower performance on Politifact is due to the increased complexity of the claims, particularly those labeled as True and Mixed, where the system tends to struggle. This performance gap highlights the need for further work to improve the system’s robustness across diverse claim types.

Consistent with our observations on the AVeriTeC dataset, the system performs best

on claims labeled False or Refuted, which often include more direct and conclusive evidence. These clearer cases make it easier for the model to render confident veracity judgments. In contrast, Mixed claims specifically in the Politifact subset pose a significant challenge. These often involve retrieved evidence that is both supporting and contradicting, requiring the system to reconcile conflicting signals. This underscores the model’s current limitations in nuanced reasoning and suggests that improving its ability to interpret and balance contradictory information is a key direction for future research.

Table 5.4: Results for the FactCellar dataset

Dataset	Accuracy	F1 Macro	F1 True	F1 Mixed	F1 False
Snopes	0.8780	0.8780	0.8270	–	0.9060
Politifact	0.6810	0.5830	0.4550	0.4770	0.8160

5.4 Concluding Remarks

This chapter began by outlining the experiments conducted to evaluate our system, *Fathom*. We presented and discussed the results obtained from both the AVeriTeC and FactCellar datasets. The experiments demonstrate that *Fathom* performs well overall, particularly on claims labeled as False. However, the system shows limitations when handling a broader range of claim types, especially within the FactCellar - Politifact subset. These findings indicate challenges in generalizing to more diverse claims. In the next chapter, we present our conclusions and outline directions for future work.

Chapter 6

Conclusion and Future Work

6.1 Conclusion

In the era of Large Language Models (LLMs), detecting disinformation and misinformation based solely on content has become increasingly challenging. The sophistication of generative models enables the creation of highly convincing falsehoods, making traditional content-based detection methods less effective. As a result, researchers have shifted their focus toward incorporating contextual signals to verify information. This context includes user metadata, propagation patterns across social networks, the broader social environment in which content spreads, and supporting or contradicting textual evidence.

To address these challenges, several fact-checking systems leveraging LLMs have been developed and deployed. These systems often rely on extracting external evidence from the web to validate claims. While effective, such approaches typically involve substantial computational costs, reliance on external APIs and latency issues making them difficult to scale for real-time or large-scale applications.

Recent research has begun to prioritize efficiency, aiming to reduce both the compu-

tational resources and time required for verification without compromising accuracy. This shift reflects a growing demand for practical, production-ready fact-checking solutions that can operate under tight resource and time constraints.

Fathom is our proposed system designed in response to this need. It serves as a proof of concept that claim verification can be performed with significantly lower resource consumption using a much smaller GPU than other proposed systems while still achieving comparable results and reduced latency, while still maintaining strong performance. We believe that *Fathom* paves the way for next-generation fact-checking systems. Such systems can be integrated into real-world workflows to support professional fact-checkers and contribute to the broader effort of combating misinformation in a scalable and sustainable manner.

6.2 Future Work

- **Expand Evaluation Across Diverse Datasets:** We plan to evaluate our system on a wider range of fact-checking datasets to better understand its generalization and performance across various domains and claim types.
- **Support for Multilingual Claims:** A key direction is extending the system’s capabilities to handle multilingual claims. This involves enabling reasoning across different languages and ensuring robust performance in diverse linguistic contexts.
- **Production-Grade Deployment:** We aim to transform the system into a production-ready application capable of verifying claims in real-world settings with low latency and minimal resource usage.
- **Leveraging Metadata from FactCellar:** The FactCellar dataset includes rich metadata such as source credibility and impact scores. In future work, we

plan to explore how incorporating this metadata can enhance claim verification and propose methods for effectively integrating these signals into the system.

Bibliography

- [1] *Media Bias/Fact Check — Search and Learn the Bias of News Media*, <https://mediabiasfactcheck.com>, [Accessed January 2025].
- [2] *Open PageRank — Free Domain PageRank Metrics and API*, <https://www.domcop.com/openpagerank/>, [Accessed January 2025].
- [3] *PolitiFact — politifact.com*, <https://www.politifact.com/>, [Accessed January 2025].
- [4] *Serper - The World's Fastest and Cheapest Google Search API — serper.dev*, <https://serper.dev/>, [Accessed January 2025].
- [5] *Snopes.com — snopes.com*, <https://www.snopes.com/>, [Accessed January 2025].
- [6] *Who.is — Domain Name Lookup and Whois Tool*, <https://who.is>, [Accessed January 2025].
- [7] Marah Abdin, Jyoti Aneja, Harkirat Behl, Sébastien Bubeck, Ronen Eldan, Suriya Gunasekar, Michael Harrison, Russell J Hewett, Mojan Javaheripi, Piero Kauffmann, et al., *Phi-4 technical report*, arXiv preprint arXiv:2412.08905 (2024).
- [8] Akshay Aggarwal, Aniruddha Chauhan, Deepika Kumar, Mamta Mittal, and Sharad Verma, *Classification of fake news by fine-tuning deep bidirectional*

- transformers based language model.*, EAI Endorsed Trans. Scalable Inf. Syst. **7** (2020), no. 27, e10.
- [9] Mubashara Akhtar, Michael Schlichtkrull, and Andreas Vlachos, *Ev2r: Evaluating evidence retrieval in automated fact-checking*, arXiv preprint arXiv:2411.05375 (2024).
- [10] Rami Aly, Zhijiang Guo, Michael Schlichtkrull, James Thorne, Andreas Vlachos, Christos Christodoulopoulos, Oana Cocarascu, and Arpit Mittal, *Feverous: Fact extraction and verification over unstructured and structured information*, arXiv preprint arXiv:2106.05707 (2021).
- [11] Anthropic, *Introducing Contextual Retrieval*, <https://www.anthropic.com/news/contextual-retrieval>, September 2024, Accessed: 2025-06-19.
- [12] Isabelle Augenstein, Christina Lioma, Dongsheng Wang, Lucas Chaves Lima, Casper Hansen, Christian Hansen, and Jakob Grue Simonsen, *Multifc: A real-world multi-domain dataset for evidence-based fact checking of claims*, arXiv preprint arXiv:1909.03242 (2019).
- [13] Pieter Balcaen, Caroline Buts, Cind Du Bois, and Olesya Tkacheva, *The effect of disinformation about covid-19 on consumer confidence: Insights from a survey experiment*, Journal of Behavioral and Experimental Economics **102** (2022), 101968, PMID: PMC9733969.
- [14] Arvinder Pal Singh Bali, Mexson Fernandes, Sourabh Choubey, and Mahima Goel, *Comparative performance of machine learning algorithms for fake news detection*, Advances in Computing and Data Sciences: Third International Conference, ICACDS 2019, Ghaziabad, India, April 12–13, 2019, Revised Selected Papers, Part II 3, Springer, 2019, pp. 420–430.

- [15] Satanjeev Banerjee and Alon Lavie, *Meteor: An automatic metric for mt evaluation with improved correlation with human judgments*, Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization, 2005, pp. 65–72.
- [16] Adrien Barbaresi, *Trafilatura: A web scraping library and command-line tool for text discovery and extraction*, Proceedings of the 59th annual meeting of the association for computational linguistics and the 11th international joint conference on natural language processing: System demonstrations, 2021, pp. 122–131.
- [17] Alessandro Bondielli and Francesco Marcelloni, *A survey on fake news and rumour detection techniques*, Information sciences **497** (2019), 38–55.
- [18] Maria Briceño, *Viral images of ‘free palestine’ ben & jerry’s ice cream tubs aren’t real*, PolitiFact (2025), Accessed 21 July 2025.
- [19] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al., *Language models are few-shot learners*, Advances in neural information processing systems **33** (2020), 1877–1901.
- [20] Michelle Castillo, *Exclusive: Fake news is costing the world \$78 billion a year*, November 2019, Accessed: 2025-05-31.
- [21] Roberto J. Cavazos, *The economic cost of bad actors on the internet: Fake news*, Tech. report, University of Baltimore and CHEQ, 2019.
- [22] CyberScoop Staff, *Israeli influence operations are powered by a commercial ecosystem*, July 2023, Accessed: 2025-05-31.
- [23] Luyu Gao, Xueguang Ma, Jimmy Lin, and Jamie Callan, *Precise zero-shot dense retrieval without relevance labels*, Proceedings of the 61st Annual Meeting of

- the Association for Computational Linguistics (Volume 1: Long Papers), 2023, pp. 1762–1777.
- [24] Sherry Girgis, Eslam Amer, and Mahmoud Gadallah, *Deep learning algorithms for detecting fake news in online text*, 2018 13th international conference on computer engineering and systems (ICCES), IEEE, 2018, pp. 93–97.
- [25] Mohammad Hadi Goldani, Saeedeh Momtazi, and Reza Safabakhsh, *Detecting fake news with capsule neural networks*, *Applied Soft Computing* **101** (2021), 106991.
- [26] Google AI, *Gemini 2.5 flash*, <https://ai.google.dev/gemini-api/docs/models#gemini-2.5-flash>, Jun 2025, Latest update June 2025; accessed June 21, 2025.
- [27] Ashim Gupta and Vivek Srikumar, *X-fact: A new benchmark dataset for multilingual fact checking*, arXiv preprint arXiv:2106.09248 (2021).
- [28] Andreas Hanselowski, *A machine-learning-based pipeline approach to automated fact-checking*, (2020).
- [29] Naeemul Hassan, Fatma Arslan, Chengkai Li, and Mark Tremayne, *Toward automated fact-checking: Detecting check-worthy factual claims by claimbuster*, Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining, 2017, pp. 1803–1812.
- [30] Naeemul Hassan, Gensheng Zhang, Fatma Arslan, Josue Caraballo, Damian Jimenez, Siddhant Gawsane, Shohedul Hasan, Minumol Joseph, Aaditya Kulkarni, Anil Kumar Nayak, et al., *Claimbuster: The first-ever end-to-end fact-checking system*, Proceedings of the VLDB Endowment **10** (2017), no. 12, 1945–1948.

- [31] Yichen Jiang, Shikha Bordia, Zheng Zhong, Charles Dognin, Maneesh Singh, and Mohit Bansal, *Hover: A dataset for many-hop fact extraction and claim verification*, arXiv preprint arXiv:2011.03088 (2020).
- [32] Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick SH Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih, *Dense passage retrieval for open-domain question answering.*, EMNLP (1), 2020, pp. 6769–6781.
- [33] Kepios, *Global social media statistics*, 2025, Accessed: 2025-05-29.
- [34] Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa, *Large language models are zero-shot reasoners*, Advances in neural information processing systems **35** (2022), 22199–22213.
- [35] Harold W Kuhn, *The hungarian method for the assignment problem*, Naval research logistics quarterly **2** (1955), no. 1-2, 83–97.
- [36] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al., *Retrieval-augmented generation for knowledge-intensive nlp tasks*, Advances in neural information processing systems **33** (2020), 9459–9474.
- [37] Christopher Malon, *Multi-hop evidence pursuit meets the web: Team papelo at fever 2024*, arXiv preprint arXiv:2411.05762 (2024).
- [38] Philip Marcelo, *Fact focus: Fake image of pentagon explosion briefly sends jitters through stock market*, Associated Press (2023).
- [39] Mohit Mayank, Shakshi Sharma, and Rajesh Sharma, *Deep-faked: Knowledge graph based approach for fake news detection*, 2022 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM), IEEE, 2022, pp. 47–51.

- [40] Luke Merrick, Danmei Xu, Gaurav Nuti, and Daniel Campos, *Arctic-embed 2.0: Multilingual retrieval without compromise*, 2024.
- [41] mixedbread-ai, *mxbai-embed-large-v1*, <https://huggingface.co/mixedbread-ai/mxbai-embed-large-v1>, 2024, Hugging Face model card.
- [42] Niklas Muennighoff, Nouamane Tazi, Loïc Magne, and Nils Reimers, *Mteb: Massive text embedding benchmark*, arXiv preprint arXiv:2210.07316 (2022).
- [43] Tobias A Opsahl, *Fact or fiction? improving fact verification with knowledge graphs through simplified subgraph retrievals*, arXiv preprint arXiv:2408.07453 (2024).
- [44] Esteban Ortiz-Ospina, *The rise of social media*, 2019, Accessed: 2025-05-29.
- [45] Heesoo Park, Dongjun Lee, Jaehyuk Kim, ChoongWon Park, and Changhwa Park, *Dunamu-ml’s submissions on AVERITEC shared task*, Proceedings of the Seventh Fact Extraction and VERification Workshop (FEVER) (Miami, Florida, USA) (Michael Schlichtkrull, Yulong Chen, Chenxi Whitehouse, Zhenyun Deng, Mubashara Akhtar, Rami Aly, Zhijiang Guo, Christos Christodoulopoulos, Oana Cocarascu, Arpit Mittal, James Thorne, and Andreas Vlachos, eds.), Association for Computational Linguistics, November 2024, pp. 71–76.
- [46] ———, *Dunamu-ml’s submissions on averitec shared task*, Proceedings of the Seventh Fact Extraction and VERification Workshop (FEVER), 2024, pp. 71–76.
- [47] Jeffrey Pennington, Richard Socher, and Christopher Manning, *GloVe: Global vectors for word representation*, Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP) (Doha, Qatar) (Alessan-

- dro Moschitti, Bo Pang, and Walter Daelemans, eds.), Association for Computational Linguistics, October 2014, pp. 1532–1543.
- [48] Verónica Pérez-Rosas, Bennett Kleinberg, Alexandra Lefevre, and Rada Mihalcea, *Automatic detection of fake news*, arXiv preprint arXiv:1708.07104 (2017).
- [49] Kashyap Papat, Subhabrata Mukherjee, Andrew Yates, and Gerhard Weikum, *Declare: Debunking fake news and false claims using evidence-aware deep learning*, arXiv preprint arXiv:1809.06416 (2018).
- [50] Stephen Robertson, Hugo Zaragoza, et al., *The probabilistic relevance framework: Bm25 and beyond*, Foundations and Trends® in Information Retrieval **3** (2009), no. 4, 333–389.
- [51] Mark Rothermel, Tobias Braun, Marcus Rohrbach, and Anna Rohrbach, *Infact: A strong baseline for automated fact-checking*, Proceedings of the Seventh Fact Extraction and VERification Workshop (FEVER), 2024, pp. 108–112.
- [52] Michael Schlichtkrull, Yulong Chen, Chenxi Whitehouse, Zhenyun Deng, Mubashara Akhtar, Rami Aly, Zhijiang Guo, Christos Christodoulopoulos, Oana Cocarascu, Arpit Mittal, et al., *The automated verification of textual claims (averitec) shared task*, arXiv preprint arXiv:2410.23850 (2024).
- [53] Michael Schlichtkrull, Yulong Chen, Chenxi Whitehouse, Zhenyun Deng, Mubashara Akhtar, Rami Aly, Zhijiang Guo, Christos Christodoulopoulos, Oana Cocarascu, Arpit Mittal, James Thorne, and Andreas Vlachos, *The automated verification of textual claims (AVeriTeC) shared task*, Proceedings of the Seventh Fact Extraction and VERification Workshop (FEVER) (Miami, Florida, USA) (Michael Schlichtkrull, Yulong Chen, Chenxi Whitehouse, Zhenyun Deng, Mubashara Akhtar, Rami Aly, Zhijiang Guo, Christos Christodoulopoulos, Oana Cocarascu, Arpit Mittal, James Thorne, and

- Andreas Vlachos, eds.), Association for Computational Linguistics, November 2024, pp. 1–26.
- [54] Michael Schlichtkrull, Zhijiang Guo, and Andreas Vlachos, *Averitec: A dataset for real-world claim verification with evidence from the web*, Advances in Neural Information Processing Systems **36** (2023), 65128–65167.
- [55] Kai Shu, Amy Sliva, Suhang Wang, Jiliang Tang, and Huan Liu, *Fake news detection on social media: A data mining perspective*, ACM SIGKDD explorations newsletter **19** (2017), no. 1, 22–36.
- [56] Yixiao Song, Yekyung Kim, and Mohit Iyyer, *Veriscore: Evaluating the factuality of verifiable claims in long-form text generation*, arXiv preprint arXiv:2406.19276 (2024).
- [57] Qwen Team, *Qwen2.5-7b-instruct*, <https://huggingface.co/Qwen/Qwen2.5-7B-Instruct>, 2024, Accessed: 2025-07-21.
- [58] James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal, *Fever: a large-scale dataset for fact extraction and verification*, arXiv preprint arXiv:1803.05355 (2018).
- [59] Maurice K.S. Tse and Clive Ho, *Combat misinformation for a stable economy*, August 2024, Accessed: 2025-05-31.
- [60] Herbert Ullrich, Tomáš Mlynář, and Jan Drchal, *Aic ctu system at averitec: Re-framing automated fact-checking as a simple rag task*, arXiv preprint arXiv:2410.11446 (2024).
- [61] Andreas Vlachos and Sebastian Riedel, *Fact checking: Task definition and dataset construction*, Proceedings of the ACL 2014 workshop on language technologies and computational social science, 2014, pp. 18–22.

- [62] Soroush Vosoughi, Deb Roy, and Sinan Aral, *The spread of true and false news online*, science **359** (2018), no. 6380, 1146–1151.
- [63] Monica L. Wang, *Pov: Health misinformation is rampant on social media*, February 2024, Accessed: 2025-05-31.
- [64] William Yang Wang, ” *liar, liar pants on fire*”: *A new benchmark dataset for fake news detection*, arXiv preprint arXiv:1705.00648 (2017).
- [65] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al., *Chain-of-thought prompting elicits reasoning in large language models*, Advances in neural information processing systems **35** (2022), 24824–24837.
- [66] Junfei Wu, Weizhi Xu, Qiang Liu, Shu Wu, and Liang Wang, *Adversarial contrastive learning for evidence-aware fake news detection with graph neural networks*, IEEE Transactions on Knowledge and Data Engineering **36** (2023), no. 11, 5591–5604.
- [67] Yejun Yoon, Jaeyoon Jung, Seunghyun Yoon, and Kunwoo Park, *Hero at averitec: The herd of open large language models for verifying real-world claims*, arXiv preprint arXiv:2410.12377 (2024).
- [68] Xinyi Zhou and Reza Zafarani, *A survey of fake news: Fundamental theories, detection methods, and opportunities*, ACM Computing Surveys (CSUR) **53** (2020), no. 5, 1–40.

Vita

Candidate's full name: Farrukh Bin Rashid

University attended (with dates and degrees obtained): National University of Sciences and Technology (NUST) 2016-2020

Publications:

Farrukh Bin Rashid, Saqib Hakak: **“Fathom A Fast and Modular RAG Pipeline for Fact-Checking”**: Eighth FEVER Workshop, Association for Computational Linguistics (ACL); June 2025; Vienna, Austria.

Arbaaz Dharmavaram, Farrukh Bin Rashid, Saqib Hakak: **FactCellar: An Evidence based Dataset for Automated Fact-Checking**”: Privacy Security & Trust; June 2025; Fredericton, New Brunswick, Canada.

Conference Presentations:

Arbaaz Dharmavaram, Farrukh Bin Rashid, Saqib Hakak; **”FactCellar: An Evidence based Dataset for Automated Fact-Checking”** UNB Research Expo; April 2025; Fredericton, New Brunswick, Canada.