University of
Sheffield

Large Language Models for Identifying Trigger Events in Disinformation Analysis

William Jeynes

Supervisors: Carolina Scarton, Olesya Razuvayevskaya

A report submitted in fulfilment of the requirements
for the degree of BSc Computer Science (with a Year in Industry)

in the

School of Computer Science

May 12, 2026

Declaration

All sentences or passages quoted in this report from other people's work have been specifically acknowledged by clear cross-referencing to author, work and page(s). Any illustrations that are not the work of the author of this report have been used with the explicit permission of the originator and are specifically acknowledged. I understand that failure to do this amounts to plagiarism and will be considered grounds for failure in this project and the degree examination as a whole.

Name: William Jeynes

Signature: *W. Jeynes*

Date: May 12, 2026

Abstract

Disinformation on the internet has become a significant and growing challenge, driven by the sheer volume and speed of content generation across digital platforms.

While prior work has thoroughly investigated tasks such as automatic disinformation detection, especially using large language models, comparatively little attention has been paid to applying these techniques to generative analyses of claims. In particular, the retrieval of structured contextual background information that explains why disinformation spreads remains underexplored.

This paper introduces a dataset of trigger events, defined as actions or statements that influence the creation, spread or believability of disinformation claims. The dataset is created using large language models as few-shot retrievers and subsequently validated through an ensemble classifier to ensure consistency.

We find that we can successfully retrieve a large dataset of trigger events with high accuracy, and that the dataset proves useful for real-world debunking and prebunking applications.

Acknowledgements

I would first like to thank my supervisors Carolina Scarton and Olesya Razuvayevskaya. Their unwavering support and help was instrumental to this project, particularly during the writing process.

I would also like to thank those around me for their support, Tom, my other flatmates and all of Josh's special people.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Disinformation is an ever-present problem on the internet, becoming much more prevalent especially in the last decade (Broda et al. 2024). This dissertation provides additional support for two disinformation countermeasures, debunking and prebunking. The novelty of this work is the first dataset of trigger events, linked to their source disinformation claims. We refine the dataset generation pipeline by training classifiers on a human-labelled dataset. The practical value of the dataset is demonstrated through two use cases.

Debunking aims to refute false claims after they have spread, whereas prebunking seeks to inoculate users against misleading narratives before exposure. Both approaches face significant challenges. First, truthful information tends to propagate more slowly than false information (Shao et al. 2016), creating a substantial period during which false claims may circulate without effective correction. Similarly, prebunking relies on early identification of emerging disinformation claims, since warnings issued about disinformation after it spreads are known to have little effect (Traberg et al. 2023).

We introduce a task designed to operate within the gap between claim propagation and debunking. Disinformation evolves to encompass real-world events and conspiracy theories (Gerts et al. 2021), therefore, identifying such events could help support proactive defences against potential disinformation claims. To exploit this relationship, we define a trigger event as a real-world action or statement that influences the spread of a given disinformation claim. As an example, a trigger event for the false claim "COVID-19 vaccines are causing a rise in heart attacks among young people" spread in February 2023 could be the on-field cardiac arrest of NFL player Damar Hamlin in January[1].

We propose an automated technique that uses large language models (LLMs) to retrieve these trigger events, implemented in TypeScript and the LangChain framework. This pipeline first normalises input claims into a consistent format. The normalised claim, together with a prompt, is then submitted to a general-purpose large language model capable of using external tools to retrieve relevant background information before outputting a set of candidate events. An ensemble of classifiers is subsequently used to filter these events to ensure quality. Using this pipeline, we extracted a dataset of 4,468 trigger events from 2,000 input disinformation

[1] `https://www.bbc.co.uk/news/world-us-canada-64149300`

claims, which is publicly available online[2].

To demonstrate the dataset's quality and utility, we propose two applications to support disinformation countermeasures. To aid the analysis of disinformation spread and the creation of debunking content, we introduce a graph-based visualisation tool that leverages the dataset's structured nature. It allows for plotting the relationships between trigger events and claims on a graph, where they form connected components of varying sizes and reveal previously unseen patterns.

Secondly, we investigate whether currently occurring trigger events can be used to predict upcoming disinformation claims to support the prebunking process. By training a large language model on a dataset of event-claim pairs, we aim to develop a model capable of predicting upcoming claims better than a zero-shot baseline.

1.1 Research Questions

This project aims to implement an automatic process of retrieving trigger events for disinformation claims. More specifically, we aim to answer the following research questions:

RQ1
How can LLMs be effectively used to retrieve structured, coherent trigger events for disinformation claims?

RQ2
Can a robust verification framework be implemented to enforce the relevance and style of the retrieved trigger events?

RQ3
How can the automatically generated dataset of trigger events be applied in disinformation visualisation and pre-bunking methods?

1.2 Overview of the Report

The report is split into five chapters. First, chapter 2 details the background for this dissertation, from the required technical prerequisites to the background on disinformation. Next, chapter 3 details the approach to generating a dataset of trigger events, covering data preparation, trigger event extraction, and classification. This section also details experiments conducted on the dataset to evaluate its usefulness. Following on from this, chapter 4 presents

[2] `https://huggingface.co/datasets/WillJeynes/LLMsForDisinformationAnalysis-Dataset`

the results of experiments in relation to the three research questions addressed in this project. Finally, in chapter 5 we discuss the overall contributions of this project, as well as directions for future work.

Chapter 2

Literature Survey

This chapter presents a review of prior research relevant to the task of extracting trigger events outlined in the dissertation. Since this is the first study to target trigger event retrieval, there is no prior work on this problem. Therefore, this chapter rather ties related tasks and approaches together, which can be used in conjunction to create the analysis. The discussion starts with an overview of online disinformation (section 2.1), then dives into technologies currently used for disinformation countering, including the use of LLMs (section 2.2). The chapter ends with more advanced techniques for prompting and retrieving information from LLMs (section 2.3), with the intention of assessing the suitability of advanced techniques such as RAG for our retreival process.

2.1 Fake News, Disinformation and Misinformation

Throughout this dissertation, although we will be using the terms misinformation and disinformation interchangeably, they represent two distinct types of fake news (Lecheler et al. 2022). The term *misinformation* refers to the spread of false or misleading information without intent to deceive. *Disinformation*, on the other hand, is the spread of false information deliberately created with the intention to manipulate people. Unlike misinformation that uninformed individuals typically share, disinformation is often a coordinated form of propaganda (El Mikati et al. 2023). Figure 2.1 illustrates the dynamic relationship between disinformation and misinformation. Individuals exposed to disinformation may unknowingly disseminate it as factual information, thereby contributing to its spread as misinformation. Conversely, malicious actors may exploit existing misinformation by reframing it to reinforce disinformation narratives that advance their objectives. This dissertation does not attempt to distinguish between misinformation and disinformation, but rather uses the analysis techniques for one or both as part of the same cohesive experiment.

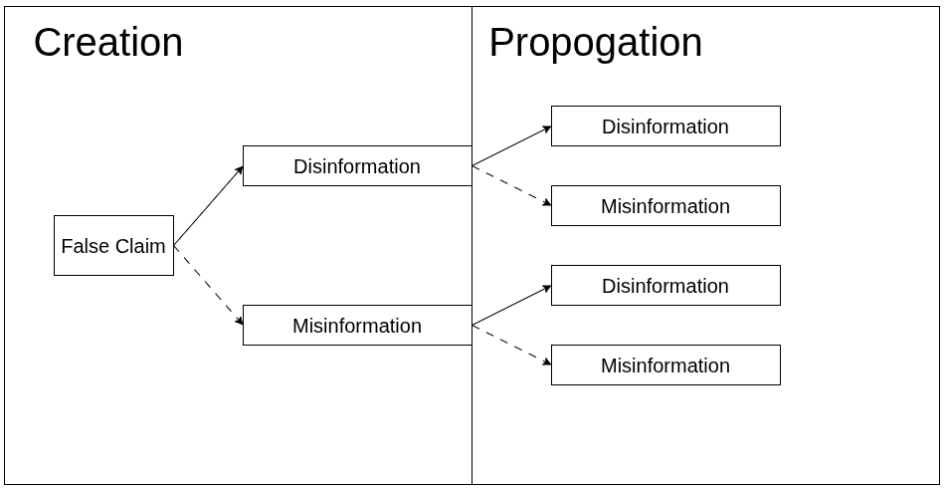


Figure 2.1: Creation/Propagation Diagram (Lecheler et al. 2022)

Misinformation can evolve over time, taking in details from other false narratives or real-world background events (Gerts et al. 2021). A famous example of this is Donald Trump citing hydroxychloroquine as a "miracle cure" for COVID-19 in the early days of the pandemic[1]. This false claim fueled an explosion of misinformation, ultimately culminating in the recent withdrawal of a highly cited paper on the effects of hydroxychloroquine from the journal Elsevier due to ethical concerns and flawed results (Gautret et al. 2020). Deliberately or not, false narratives around COVID-19 quickly began to centre on hydroxychloroquine, capitalising on the real-world press conference by the then US President.

Examining instances such as this is important, since COVID-19 was the world's most severe global pandemic in the information age. The advent of health-related misinformation created a massive problem for governments trying to roll out schemes such as vaccination and contact tracing, with some branding it an "infodemic". Studies have shown that exposure to this misinformation had a measurable negative impact on people's health and economic outlooks for the future (Balcaen et al. 2023).

Similarly, after Russia's invasion of Ukraine, disinformation has been actively used as a tool of information warfare by the Russian government (OECD 2022). This presents a slightly different set of challenges from COVID-19 disinformation, since, instead of the accidental spread of false stories, the spread of anti-Ukrainian propaganda is deliberate and planned. Overall, the sheer volume of misinformation is becoming impossible to manage, and this is met with a significant body of research on the topic (figure 2.2).

[1] `https://www.nytimes.com/2020/05/18/us/politics/trump-hydroxychloroquine-covid-coronavirus.html`

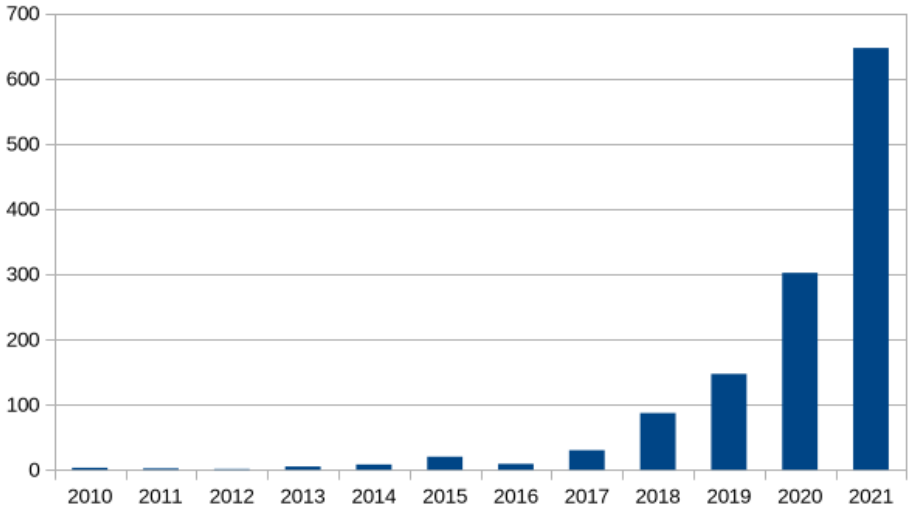


Figure 2.2: Publications Focussing on Disinformation Over Time
Data from Broda et al. 2024

2.1.1 Debunking and Prebunking

To counter disinformation, there are many methods, with debunking and prebunking being particularly central to this project. *Debunking* is the process of providing a set of corrective messages that convince the user that the preceding messages were disinformation (Chan et al. 2017). Whilst extensively used, current implementations have several issues (Swire-Thompson et al. 2020, Walter et al. 2019, Tay et al. 2024). One of the key challenges in effective disinformation debunking is the delayed response time. Disinformation spreads faster and reaches more people than true news (Vosoughi et al. 2018). The majority of susceptible users who would see a false news article shared to Twitter do so within the first 6 hours of the post's existence (Tokita et al. 2024). In comparison, the widespread sharing of debunking content takes up to 20 hours (Shao et al. 2016). It is therefore vitally important to be able to react to disinformation instantaneously, or, where possible, predict it before it receives widespread attention.

As an alternative technique for countering disinformation, *prebunking*, or psychological inoculation, is the process of providing a warning with details of an upcoming disinformation campaign, which has been shown to decrease the believability of fake news (Linden et al. 2017). As a result, disinformation becomes ineffective if all susceptible people scroll past and ignore the post. Prebunking has been used in practical studies as recently as 2026 (Linden et al. 2026), proving that even short advertisements on social media can have a real-world reduction in the efficacy of disinformation spread. Despite this, any prebunking effort that occurs after a given piece of disinformation has been manually detected and analysed risks being too late; a large portion of inoculation will occur after exposure (Traberg et al. 2023), significantly reducing effectiveness.

This dissertation aims to create an automated analysis pipeline that operates during the

time gap between background events and extensive claim propagation, laying the groundwork for preventative methods that act much earlier in the disinformation lifecycle.

2.2 Automated Approaches to Disinformation Processing and Analysis

The methodological design for this project considered existing cutting-edge techniques for countering misinformation. As in many research areas, LLMs represent the current state-of-the-art. This section covers the foundational models used in disinformation detection and debunking, such as transformer models and their application to countering disinformation. It ends by introducing LLMs for disinformation analysis and their myriad task-specific uses, including generation and normalisation.

2.2.1 Neural Networks

Neural networks are a class of machine learning models inspired by the biological model of the brain, with nodes acting as neurons and the connections between representing synapses. Multilayer Neural Networks consist of several interconnected layers of artificial neurons. The input layer receives node values from the external environment or dataset; these could be anything that can be transformed into a series of numbers, be it sensor values, images, or even text. Next, the values pass through one or more hidden layers, which transform the data via weighted connections between nodes and activation functions. Finally, the values of the output layer neurons can be used to determine the model's classification or decision. By updating the weights between these nodes, different outputs can be obtained at the final layer (Rosenblatt 1958).

Classification is the process of grouping data into predetermined groups, based on shared characteristics or criteria (Hastie et al. 2009). In the context of disinformation processing, classifiers often operate on text inputs to determine whether a given statement is disinformation.

Al-Tarawneh et al. 2024 details an approach to using neural networks (multiplayer perceptrons) as a disinformation classifier. Transformer models (section 2.2.2) embed words, and the resulting vectors serve as input to various types of neural networks. In the paper's experiment, multilayer neural networks correctly identified disinformation with an accuracy of over 98%, while remaining reasonably small and quickly trainable.

2.2.2 Transformer Models

Transformer models are a type of neural network architecture, specifically designed to better process sequences of data, most commonly groups of words / text (Vaswani et al. 2017). Earlier neural network approaches to language processing often handled words one at a time, making it difficult to efficiently capture long-range relationships and the wider context within a sentence. Transformer models extend neural networks with the multi-head self-attention

mechanism, which processes words in relation to other content in the sentence, rather than naively in a left-to-right manner. This enables the model to access both the left and right context of a word, leading to a better understanding of meaning, grammar, and dependencies within text.

Bidirectional encoder representations from transformers (BERT, Devlin et al. 2018) is a transformer-based language model, designed to learn contextual representations of text by processing words bidirectionally in a sequence. BERT processes both left and right contexts of each word simultaneously, enabling stronger performance on natural language understanding tasks such as text classification, question answering, and sentiment analysis.

RoBERTa (Y. Liu et al. 2019) builds upon the BERT architecture but was trained on substantially larger datasets for longer than the original BERT model, improving representation performance. The RoBERTa-base model contains approximately 125 million parameters and was trained on over 160GB of text data. Prior research has shown that finetuned versions of RoBERTa can classify misinformation (Truică et al. 2022), achieving 91% accuracy, even with the base version.

Similarly, fine-tuned language nets (FLAN, Chung et al. 2022) are a series of models designed to be more efficient at executing instructions. Typically, FLAN models are based on a Text-to-Text Transfer Transformer model (T5, Raffel et al. 2019). T5 models are designed for natural language processing tasks, in which each task is formulated as a text generation problem, where the model receives a textual prompt or question as input and generates a textual response as output. For example, a classification task can be represented as a question where the generated answer corresponds to a class label. Unlike BERT models, FLAN-T5 uses an encoder-decoder architecture, allowing for full text input and output. For detecting medical disinformation, a finetuned FLAN model outperformed large models, including the 7 billion parameter Llama-2, and smaller models such as BERT (Robles-Pagán et al. 2025).

2.2.3 Ensembling

For tasks such as classification, the outputs of multiple models can be combined to make the final prediction, potentially with a higher accuracy (Breiman 1996). This process is demonstrated by Truică et al. 2022, where two models with completely different architectures, trained at different times, are combined into a single "ensemble" model that improves performance on classification tasks compared to each model individually. According to Maclin et al. 1998, theoretically, an ensemble model can achieve a near-zero error rate if its component models are individually highly accurate. To maximise this effect, differences between training sets and alternative architectures in the component models should account for disagreement across different sets of samples as much as possible.

2.2.4 Large Language Models

LLMs excel in text processing and logical reasoning, making them well-suited for complex, nuanced disinformation-related tasks, as evidenced by extensive literature on the topic

(Radford et al. 2018). This makes these models directly applicable to our task of disinformation analysis, and their effectiveness - particularly in terms of accuracy, robustness, and structured information extraction - can be further improved using the cutting-edge techniques defined in section 2.3.

One caveat for LLMs in the world of disinformation is that most models have safeguards that prevent them from responding to politically sensitive or divisive topics, even when discussed academically. These safeguards, however, are inconsistent and can often be reliably broken with "jailbreaking" prompts (Souly et al. 2024). These techniques open the door for LLMs to generate large databases of disinformation, such as AI-TRAITS (Leite et al. 2025). AI-TRAITS is a million-plus-record set of personalised AI-generated multilingual disinformation claims. Although most attempts to generate misinformation are malicious in intent, research in this area can serve important legitimate purposes. In controlled scenarios, generated claims can help improve disinformation detection systems and support the development of LLM safeguards by demonstrating the capabilities of public models.

Disinformation claims are very hard to work with when taken directly from the source. This is because many false claims are worded completely differently and are often embedded within a long chain of discourse alongside other irrelevant information. To counter this, the process of *normalisation* converts long and differently worded statements into common, concise claims. The CLAN database (Sundriyal et al. 2023) is a large multilingual dataset of normalised claim pairs, generated using generic LLMs and few-shot prompting (see section 2.3.2). The few-shot prompting approach has since been improved by prioritising the examples in the prompt based on their similarity to the input claim (Pramov et al. 2025). This step means the examples provided are more relevant to the subject area, making inference easier and potentially leading to significant improvements in the model's performance.

Recent approaches to misinformation detection are following similar trends to claim normalisation. Kramer et al. 2025 uses a similar few-shot approach on a labelled dataset of pro-Russian tweets against unbiased tweets, and achieves a 90% accuracy on the test set. Despite its closed-source architecture and unknown training parameters, Kramer et al. found that GPT-4 performs considerably better in this task than open-source models such as Llama. While it is important not to focus experiments on a single language model, this and similar research show that modern GPT models can avoid bias and make inferences even on politically divisive topics.

2.3 Task Adaptation Strategies

While the previous section goes into detail about existing strategies for completing long-standing disinformation-processing tasks, such as classification, this dissertation proposes a new task: extracting disinformation trigger events using LLMs. This shift requires drawing on advances from related research areas, including information retrieval and prompt-based learning. In particular, we consider techniques such as retrieval augmented generation and few-shot prompting.

Many of the papers in this chapter focus on increasing LLM accuracy against general question-answering benchmarks. Research has shown, however, that a model or technique that performs better on these benchmarks will also perform better on any context-heavy use case (N. F. Liu et al. 2023).

2.3.1 Retrieval Augmented Generation

As explained in section 2.1, disinformation narratives are constantly evolving, taking on details from current events as they happen. For any efficient pipeline of trigger event retrieval, it must have a detailed context for events that occurred hours or weeks ago. LLMs are pre-trained on large text datasets, and the point in time when the pre-training data was last updated for an LLM is called the *cutoff point*. Any events that happen after this cutoff point become a gap in its knowledge base.

To provide the required context beyond a model's cutoff point, we can consider the context grounding technique called Retrieval-Augmented Generation or RAG (Lewis et al. 2020). RAG represents a group of techniques that, given a large collection of documents relevant to a specific task, retrieve documents relevant to a given query and feed them into an LLM, along with the user prompt. This allows the model to access non-public or post-cutoff information while also improving reasoning ability.

Gao et al. 2023 details all current approaches to RAG, from naive generation to more advanced modular RAG set-ups. In a naive generation scenario, documents are initially retrieved based on the input query and then fed into the agent along with the unmodified query, which the language model uses to produce an answer. This technique struggles with precision, since the retrieved documents often fail to capture the nuance of the question, and results are matched only by simple text-similarity metrics. Advanced RAG addresses some of these issues by optimising the generated search query and by using a more computationally intensive ranking method to further refine the returned results post-initial retrieval. While this yields better results on QA datasets, it still lacks support for the multi-step reasoning that may be required when analysing complex disinformation cases. To overcome these issues, researchers are investigating more advanced RAG patterns, including Iterative, Recursive, and Adaptive Generation (Gao et al. 2023). Each of these methods relies on the concept of multiple generation steps, with differing approaches to how this should be executed. Iterative RAG completes multistep-generation naively, wherein data is retrieved based on the current context, and then LLM generation occurs, and the process repeats. It has benefits over standard RAG; however, it can suffer from a gradual accumulation of irrelevant documents. Recursive RAG is another framework for solving complex problems, where at each step, instructions are given to the LLM to split the task into solvable subproblems, up to a maximum depth. Each step retrieves relevant documents, which are summarised and passed back to the query that requested them. It has been shown to perform well with Chain-of-Thought prompting (section 2.3.2) and on questions where the intended research direction is not immediately obvious (Trivedi et al. 2022). Finally, Adaptive RAG is similar to Iterative, however LLM judgment is used for how and when the extra content is required,

to solve issues around document accumulation.

In user-facing AI tools, the concept of Iterative RAG has been combined with web and search-engine data to improve the accuracy of language models and enable citations (Nakano et al. 2021). Combining Iterative RAG with properly filtered internet data has been shown to answer questions on almost any topic accurately.

When adding context to an LLM, it is important to consider the amount of text provided to the model. A given LLM has a context window, which is the amount of text it can use at any given time. Exceeding this window significantly reduces the effectiveness of the initial prompt (N. F. Liu et al. 2024). We also have to balance the accuracy and quantity of additional context, since adding irrelevant text increases the chances of hallucination (Z. Wei et al. 2025). The task of finding only the most relevant documents from a large corpus is a specific case of an Information Retrieval (IR) task. One option is BM25 (Robertson et al. 2009), an algorithm for finding relevant documents in a search corpus that takes into account relative keyword frequency and document length. Alternatively, embeddings allow us to convert a sentence into a vector representation using a language model such as Word2Vec (Mikolov et al. 2013) or miniLM (W. Wang et al. 2020). The cosine similarities between the search query and the vectors in the search corpus are used to rank the retrieved documents. Sawarkar et al. 2024 proposes a hybrid approach in which the scores from a keyword-matching algorithm like BM25 (sparse) and a semantic-similarity metric like cosine similarity (dense) are combined into a single score, used to rank and select relevant examples. This approach, called *Blended RAG*, provides incremental improvement over one of these techniques individually.

2.3.2 Prompt Engineering

To elicit good results from an LLM and encourage efficient utilisation of extra context, we have to provide a well-crafted natural-language prompt to the model. Prompt engineering is an emerging area of research, however, there are currently no standardised approaches to creating the best prompt for a given LLM task, as studies are task-dependent. There is, however, a diverse range of techniques that can be experimented with (Sahoo et al. 2024). *In-context-learning* is a process in which additional information is provided to the model as part of the prompt. The process is utilised in *few-shot prompting* (Brown et al. 2020), in which a model is given a series of perfect examples to similar problems. The *shot* of a given prompt refers to how many examples are given. Zero-shot relies on instruction alone, one-shot is one example, and few-shot is two or more. Brown et al. show that even for the models of large sizes, adding extra examples to a prompt yields incremental improvements in model performance, demonstrating the ability to tune a model to a specific use case.

Chain-of-Thought (CoT) prompting (J. Wei et al. 2022) is a process designed to elicit multistep reasoning from an LLM. This can be used both in isolation (zero-shot) and in combination with few-shot prompting.

In the zero-shot variant, a trigger phrase, for example “Let’s think step by step”, is added to a typical prompt. This method sees incredible performance boosts compared to zero-shot prompts without the phrase (Kojima et al. 2022). This technique is not model-specific and

has proven boosts for both GPT and PALM models.

When used in combination with few-shot prompting, reasoning steps are embedded within the provided examples, explaining how the answer was reached. This strategy implicitly defines sub-problems for complex reasoning tasks. It prompts the model to output its reasoning steps as it solves the problem, making it easier to debug incorrect answers (J. Wei et al. 2022). The technique works well for off-the-shelf models and has been proven to outperform finetuned models in many scenarios (J. Wei et al. 2022, Zhang et al. 2023).

Another direction to elicit reasoning in an LLM is to break down the steps required to solve a multistep problem into the prompt (Plan and Solve, L. Wang et al. 2023). This technique requires fewer resources than few-shot CoT prompting, however, it can be tailored to a specific task, unlike trigger phrases. The technique achieves similar overall accuracy to few-shot CoT, and L. Wang et al. show that it actually beats CoT on some symbolic reasoning question sets. Table 2.1 shows examples of these strategies.

Prompting strategies for the imaginary scenario:
"what is the opposite of the following word?"

Prompt Strategy	**Example**
Few Shot	Q:Black A:White, Q:Bad A:Good, Q:Slow A:Fast ...
Chain of Thought (few-shot)	Q:Black A:Black is the absence of light, therefore we want an abundance of light, so White...
Chain of Thought (zero-shot)	... Think it through step by step.
Plan and Solve	First, write the definition of the word. Then, invert this definition. Finally, find what word corresponds to this definition ...

Table 2.1: Prompt Examples

As well as improving our initial generation, we can consider self-critiquing (Ho et al. 2025), allowing the LLM to critique its own answers and then act on its own recommendations. The approach is a three-step process: initial prompting, critique against some criteria, and then improving the initial response against the generated critique. Consistent gains across all model types demonstrate the adaptability of multi-step processes when working with language models. Some hallucinations and incurabilities are an inevitability, however, in this framework they are caught before causing problems in the final response, assuming, of course, that the extra generation steps do not add errors of their own.

Although few-shot learning and prompting techniques can achieve reasonable performance, for some complex tasks, full fine-tuning of LLMs may be needed (Pavlyshenko 2023, Tian et al. 2023). Due to the size of most LLMs, Parameter-Efficient Finetuning (PEFT) methods are used to reduce computational costs and time (Houlsby et al. 2019, Li et al. 2021). In the next section, we will discuss one specific type of PEFT relevant to this dissertation.

2.3.3 Low-Rank Adaptation

Low-Rank Adaptation (LoRa) is a method for finetuning pre-trained models in which most of the node weights are frozen, while only a few additional injected nodes are actually tuned. Doing this significantly reduces the system resources required for training and often yields results comparable to finetuning the full model (Hu et al. 2021). This process offers several advantages: training takes significantly less time and requires fewer resources, and it can converge faster than the full model, since there is less to tune. This approach can be applied to finetune models for disinformation detection (Pavlyshenko 2023). Pavlyshenko proposes a model based on the large model LLama2, augmented with LoRa, and trained on fewer than 50,000 examples. With this technique and low-cost GPU resources, the model saw a significant increase in its ability to deeply analyse a given piece of disinformation, processing complex narratives whilst still producing usable outputs.

2.3.4 Dataset Augmentation

Even with performance improvements like LoRa, there might still be a need to increase the effective size of a training set. An example of this arises in classification tasks, where label imbalance in the training set significantly impacts model performance (Buda et al. 2017). One can balance label proportions in the training set through sampling; however, this approach reduces the effective dataset size by undersampling frequent labels.

Alternatively, we can consider LLM-powered data augmentation (ValizadehAslani et al. 2024). This technique involves prompting an LLM to generate additional examples of unpopular labels and inserting them into the training set. This technique not only solves class imbalance but also increases model robustness and generalisation by adding previously unseen word structures (Bayer et al. 2022).

A similar data augmentation technique is Back Translation (Sennrich et al. 2015). In this solution, examples of underrepresented labels are translated into another language and then translated back into the source language. While this technique is not commonly used specifically for data balancing, it is an effective method for generating semantically similar sentences with varied phrasing, offering more diversity than simply duplicating the original examples.

2.4 Summary

This chapter first considered the growing issue of disinformation and misinformation online. We then consider the effectiveness and limitations of debunking and prebunking, the methods this dissertation hopes to improve. This chapter then discussed existing approaches to disinformation analysis, including the utilisation of feedforward neural networks, transformer models and LLMs. We also considered the results of merging multiple of these model types together, a technique known as ensembling. Finally, this chapter considered advances from related research areas that could be applied to our novel extraction task. We

first consider techniques for working with pre-trained LLMs, namely RAG and prompting strategies such as Few-Shot, CoT, and self-critiquing. We subsequently examine techniques applicable when prompting alone is insufficient, and finetuning becomes necessary, including the parameter-efficient LoRa approach for training large-scale models, as well as dataset augmentation methods used to improve class balance prior to training. In the following chapter, we will describe how these methods were directly applied to trigger event extraction.

Chapter 3

Methodology

Within this project, the definition of a trigger event is provided in figure 3.1. As described in section 2.1, real-world events and other unrelated false narratives can evolve into misinformation claims over time. This definition restricts the analysis to only include statements and actions, since their existence can be reliably validated using publicly available data.

> *Trigger Event:* A real-world action or public statement that influences the spread or believability of a given misinformation or disinformation claim.

Figure 3.1: Trigger Event Definition

Figure 3.2 outlines the main stages of the proposed pipeline for automatically generating disinformation trigger events. The first stage, shown in green, details the preparation of the input data for analysis, as explained in section 3.2. With this cleaned data, we can then prompt an LLM, with the instruction to generate trigger events (section 3.3, red). We must filter our LLM-generated outputs to ensure only high-quality events are generated in downstream tasks. To achieve this goal, we need to train a classifier to automate the labelling process (section 3.5, yellow). Such a classifier relies on a set of human-labelled responses (section 3.4) to serve as the initial training data. Finally, we explore creating a large dataset of trigger events, and evaluate their real-world usefulness in section 3.6. An example of the full process is displayed in figure 3.3.

For consistency, the following sections are presented in logical order rather than the order they were completed in. To evaluate the dataset generation process at scale, the classifier and, therefore, human labelling were completed first, allowing LLM versions to be quickly quantified.

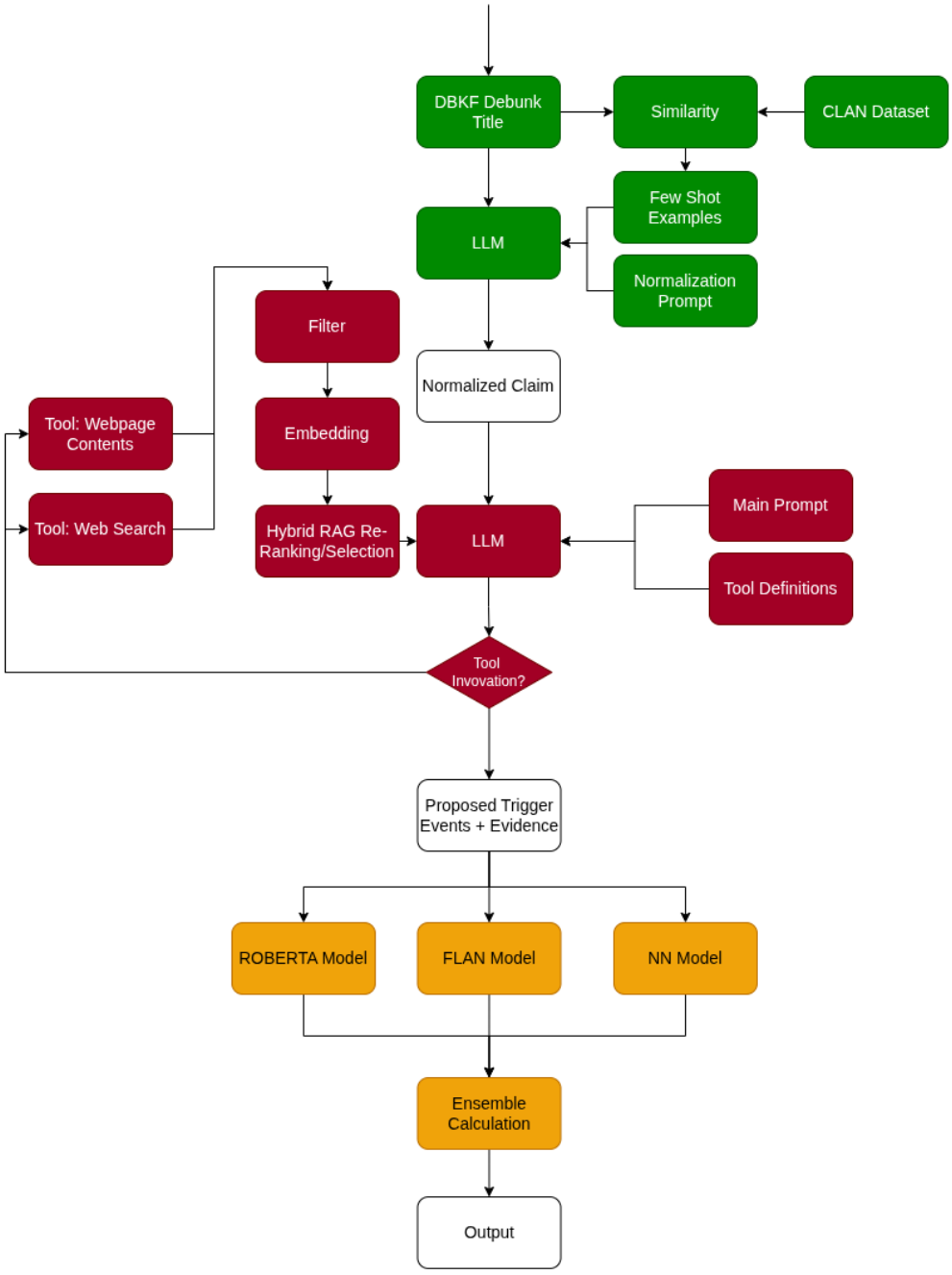


Figure 3.2: System Architecture

Key: Green=Claim Normalization, Red=Claim Generation, Yellow=Claim Classification

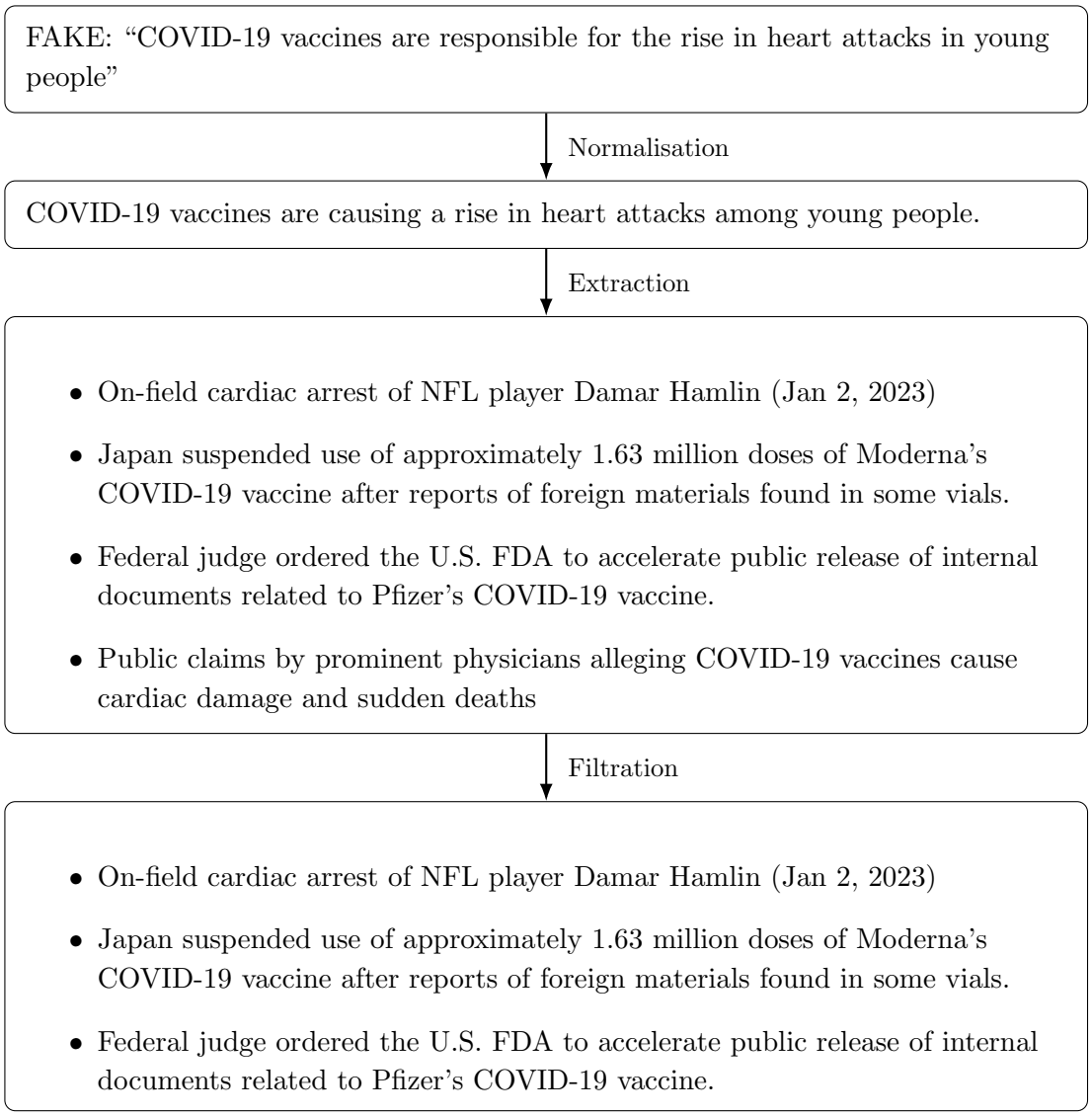


Figure 3.3: Trigger Event Extraction Example
For additional examples, see appendix C

3.1 Technology

The disinformation processing pipeline was written using the framework LangChain (Mavroudis 2024). The motivation for using LangChain lies in its frontend tools, which provide a streamlined developer experience, allowing for simple user input and debugging. The framework is developed in TypeScript[1], enabling type safety and enhanced security. Its production deployment enables highly parallel workloads, which in turn yields significant speed improvements, since a substantial portion of processor time is spent waiting for slow

[1] `https://www.typescriptlang.org/`

external API responses.

The framework allows the developer to model the entire process as a series of interchangeable graph nodes. These can be created at runtime to simplify the development experience further. Figure 3.4 demonstrates a screenshot of the LangChain interface, running the final version of this project's disinformation processing pipeline (source code available on GitHub[2]). It is analogous to our formal system architecture (figure 3.2), demonstrating a graph-based programming framework's ability to model a problem almost exactly as it was designed.

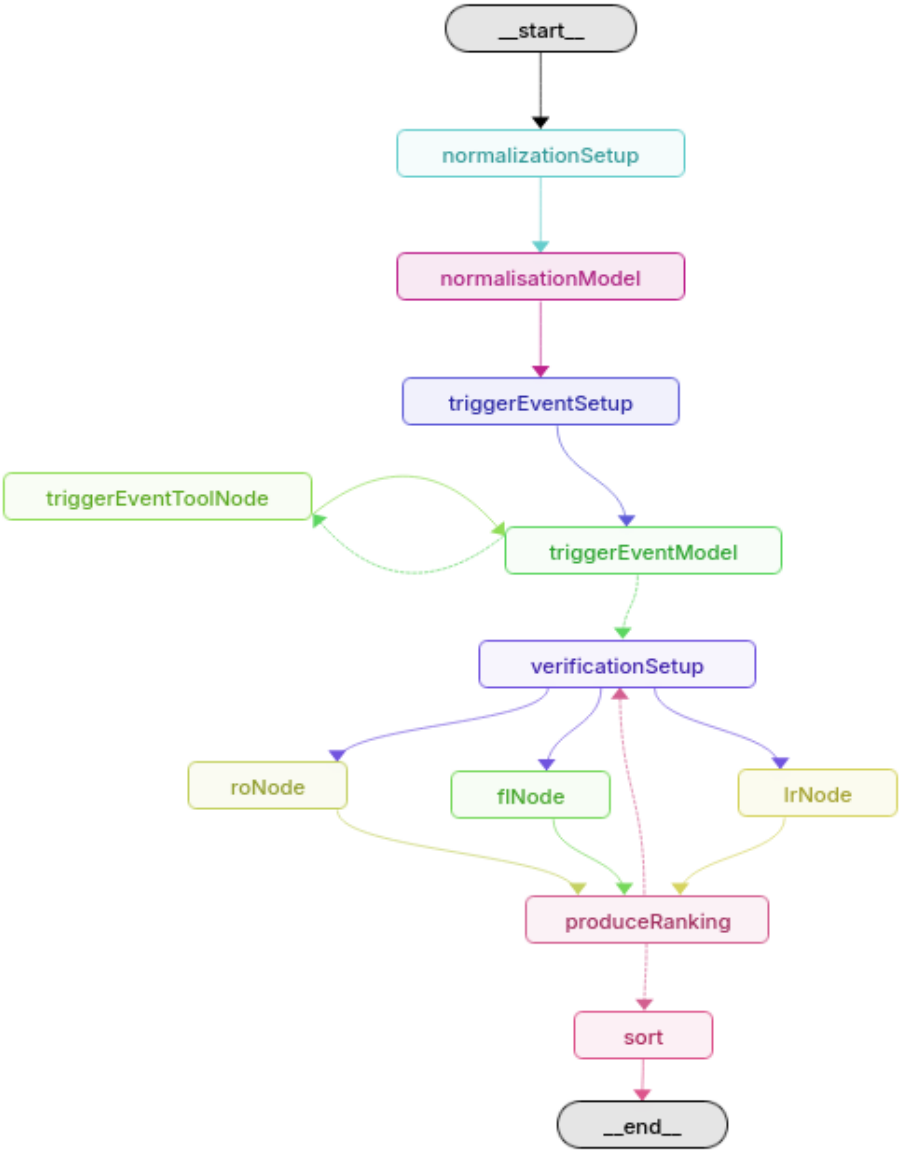


Figure 3.4: Screenshot of the LangChain Interface Running Our Pipeline

[2] https://github.com/WillJeynes/LLMsForDisinformationAnalysis/

3.2 Input Data Selection

As explained in section 2.2.4, through the process of *claim normalisation*, we can convert any piece of disinformation into a concise statement ready for processing. Debunk articles often address impactful and popular disinformation claims as a priority, since these have the greatest potential for harm. Through our *normalisation* process, we extract the underlying claim being addressed from each debunk article headline. We therefore propose that the resulting collection of extracted claims constitutes a curated subset of disinformation that is considered especially important to counter, since human effort has been expended to try to stop its spread.

The Database of Known Fakes is[3] a comprehensive website that collates a large number of debunk articles, grouped by source, type, and more. Its API was used to programmatically fetch debunk articles and metadata from the sites StopFake[4] and ScienceFeedback[5] for disinformation about the Ukraine War and COVID-19, respectively.

As seen in table 3.1, debunk article titles have an inconsistent format, ranging from negating the disinformation claim, to just a "FAKE:" prefix. The process of claim normalisation should therefore be robust enough to turn each of these cases into the single incorrect statement that the debunk article is disproving.

Input Debunk Title	**Output Claim**
Did NATO Undertake 'Operation Justice' To Intervene In Ukraine?	NATO launched "Operation Justice" to intervene in Ukraine.
Fake: Crimean Tatar Mejlis a US Creation	The Crimean Tatar Mejlis was created by the United States.
No, getting a COVID-19 vaccine won't expose you to high amounts of electromagnetic radiation	Getting a COVID-19 vaccine exposes you to high amounts of electromagnetic radiation.

Table 3.1: Examples of Claim Normalisation on Debunk Article Titles

Normalising a debunk article into a concise false statement is therefore the first step in the disinformation analysis pipeline. Since claim normalisation is not the focus of this project, this step applies an existing technique for prompting an LLM with a few-shot prompt, as outlined in Pramov et al. 2025. The process starts with the CLAN dataset (Sundriyal et al. 2023), a large multilingual dataset of post-claim pairs. Each claim is embedded with all-MiniLM-L6-v2[6], as well as the input claim. By comparing cosine similarities, only the 5 most similar claims are provided as examples in a prompt to an LLM, along with detailed instructions of the intended output type. (see B.3)

[3] at the time of writing, it has become unavailable
[4] `https://www.stopfake.org/en/main/`
[5] `https://science.feedback.org/`
[6] `https://huggingface.co/Xenova/all-MiniLM-L6-v2`

3.3 Dataset Generation

The proposed trigger events are generated using the steps outlined in the red nodes of the architecture diagram (figure 3.2). The most critical aspect of this process is the RAG setup (section 3.3.1), providing the vital extra context required to produce high-quality analysis. This is supplemented by the system prompt (section 3.3.2), which provides the model with clear instructions for generating the analysis. Multiple techniques were experimented with for the prompt, as outlined in section 3.3.3.

3.3.1 RAG Setup

This project takes advantage of RAG, combined with the concept of tool calling (see section 2.3.1). This process is illustrated in figure 3.5. In addition to the main prompt, a series of tool specifications are sent to the LLM detailing the tool description and required parameters. Ideally, after prompting, the LLM requests a wide range of web searches for background information instead of using its own pre-training data to provide an instant response. Details of the search results are passed to the LLM, and, if more information is required, the LLM can request a specific webpage to be opened to retrieve the full webpage content. With all of this extra information, a more accurate response can be provided. Specific implementation details are described below.

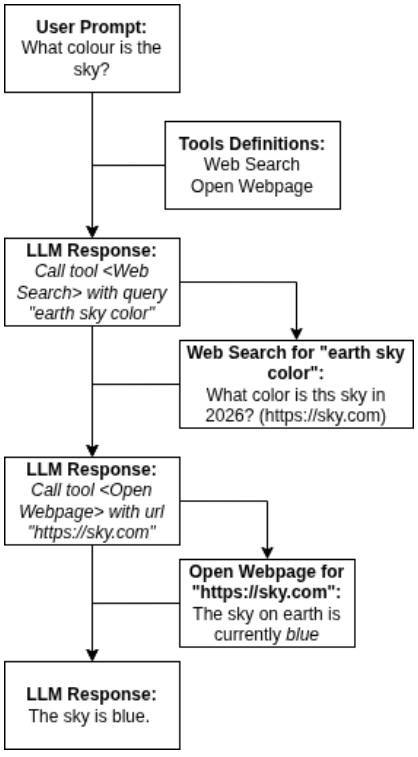


Figure 3.5: Proposed RAG Solution

Tool 1: Web Search

The most basic method of information retrieval is similar to how we, as users, retrieve data: using a search engine. A request for data from the language model is passed as a query to a conventional web search engine, and a filtered list of results is returned to the LLM in a consistent format. This technique is similar to the one outlined by Nakano et al. 2021. It allows us to collect data from a wide range of sources across the internet, enabling any possible research direction. For this implementation, a scraper was used for the free search engine DuckDuckGo[7], and the information was returned to the language model in a markdown-formatted document containing page titles, excerpts and URLs.

Search engines have historically been vulnerable to manipulation of their internal ranking algorithms by malicious actors, often trying to spread misinformation or scams (Williams et al. 2023). Presenting these to the LLM as facts could undermine the quality of the analysis. To mitigate this risk, a filter list was used to remove potentially biased or inaccurate results. For this project, we applied the Iffy index[8] for its extensive coverage and frequent updates.

Tool 2: Open Webpage

If the language model needs context beyond what is available in the page excerpt provided by the search tool, an extra tool is called. It allows snippets of a webpage to be extracted, assuming a valid URL is provided.

To ensure that the content is returned correctly for dynamic webpages or sites with aggressive bot filtering, a headless instance of the popular open-source web browser Firefox[9] is automatically activated by Selenium Web Driver[10]. The code is then injected into the rendered webpage to extract the relevant text snippets and return them to the LLM.

Blended RAG

As described in section 2.3.1, exceeding a context limit degrades performance, with the prompt having significantly less effect. To avoid overwhelming a given context window, there needs to be a method for refining and prioritising the returned tool content, ranking it by the most appropriate to the given query.

To filter context returned to the language model, this project uses Blended RAG (Sawarkar et al. 2024). In this implementation, the embedding model used to create the dense scores was a all-MiniLM-L6-v2[11]. The sparse-term matching to rank the results was performed using BM25 (Robertson et al. 2009).

[7] https://duckduckgo.com/
[8] https://iffy.news/index/
[9] https://www.firefox.com/en-GB/
[10] https://www.selenium.dev/documentation/webdriver/
[11] https://huggingface.co/Xenova/all-MiniLM-L6-v2

3.3.2 Prompt Setup

An example of a core system prompt for trigger-event generation can be found in section A.1. The prompt details a specific response format and instructions for the intended response content.

Before the system prompt, few-shot prompting is applied, with several examples of disinformation claims and trigger event pairs added to the context window. These example claims were taken from the human-labelled dataset defined later in the dissertation (section 3.4). As in our setup for claim normalisation (section 3.2), both the input claim and existing examples were embedded with all-MiniLM-L6-v2[12], and, using cosine similarity, only the 5 most similar examples were selected for the prompt.

To reliably process the data further in the pipeline, an output schema was set up for the LLM to adhere to (figure 3.6). Any output was then parsed into a JSON document, and the schema was verified with the TypeScript library zod[13].

```
[
  {
    "Event": "...",
    "ReasoningWhyRelevant": "...",
    "SearchQuery": "...",
    "Url": "...",
    "Date": "..."
  }
]
```

Figure 3.6: Expected JSON Output Schema

3.3.3 Experimental Setup

Section 3.5 details the process of creating an automatic classifier to assess the quality of generated trigger events. By comparing the accuracy for the same set of input claims, we can assess efficiency before and after a prompt or model change. Improving the proportion of valid events, also referred to as *accuracy*, not only raises efficiency, but it also assists the downstream tasks such as visualisation (section 3.6.2), making it an important metric to attempt to improve.

Table 3.2 details the experiments completed on the main trigger event generation prompt, along with a link to the prompt utilised. Descriptions of the techniques can be found in section 2.3.2. We should compare any prompt changes to the accuracy achieved during the labelling of our initial dataset (33%). An increase in accuracy when using a prompting technique indicates that it is well-suited to extracting trigger events.

[12] https://huggingface.co/Xenova/all-MiniLM-L6-v2

[13] https://zod.dev/

Technique	Prompt
Unedited (baseline)	A.1
Chain of Thought Prompting (Zero-shot)	A.2
Plan and Solve	A.3
Self Critique	A.4

Table 3.2: Experiments on Prompting Techniques

Another set of experiments involves changing the LLM used to extract the trigger events. Training differences and smaller/larger contexts yield different-quality results, and ability is often task-specific. Our experimental list is presented in table 3.3, which shows our experiments with drop-in replacement LLMs.

Model	Link	Year
gpt-5-mini	[1]	2025
gpt-5.4-mini	[2]	2026
gpt-5.4-nano	[3]	2026
gpt-4.1-mini	[4]	2025
gpt-4o-mini	[5]	2024
llama3.1:8b-instruct-q4_K_M	[6]	2025
qwen3.5:9b	[7]	2026

[1] `https://developers.openai.com/api/docs/models/gpt-5-mini`
[2] `https://developers.openai.com/api/docs/models/gpt-5.4-mini`
[3] `https://developers.openai.com/api/docs/models/gpt-5.4-nano`
[4] `https://developers.openai.com/api/docs/models/gpt-4.1-mini`
[5] `https://developers.openai.com/api/docs/models/gpt-4o-mini`
[6] `https://ollama.com/library/llama3.1:8b-instruct-q4_K_M`
[7] `https://ollama.com/library/qwen3.5:9b`

Table 3.3: Model Experiments

When evaluating these model improvements, it is also important to assess the agent's ability to process URLs and RAG data, specifically its ability to accurately retrieve, retrain, and reproduce valid links without hallucination. Some models may bypass the RAG tool calling and generate plausible but non-existent URLs based on the pre-training data. To address this, a verification program was created to check each URL generated during the experiments. The program determines whether the domain is reachable, and the specified page currently exists on the website. This allows us to quantify the number of valid links returned. The URL validity metric is used alongside our classification accuracy metric to provide a more comprehensive evaluation of model performance.

3.4 Dataset Annotation

As described in section 3.3.3, we trained a classifier to evaluate the quality of the LLM-generated responses by labelling them. This allows us to discard low-quality generations

from our dataset and provide examples for the few-shot prompt defined in section 3.3.2. For this reason, we manually labelled the 750 disinformation claim-trigger event pairs (150 unique claims with 5 candidate events each) using a set of criteria detailing output quality. To facilitate this annotation process, a simple interface was created in Streamlit[14], which presents the user with the unlabelled claims in a random order, along with the artificially generated trigger events. Each event can be labelled with one or more of the boolean flags, listed with examples in table 3.4.

Trigger events for the imaginary disinformation "birds can own parking spaces"

Label	**Example**
Perfect	Statements by the mayor falsely proclaiming ...
Story	The false claim was then spread on Facebook ...
Time Incorrect	Debunk articles were created in response to the claim ...
Not Specific	General sentiments of anti-bird skepticism ...
Rewording	Existing false claims that birds can own parking spaces ...
Duplicate	The mayor said ...

Table 3.4: Labels and Examples

Perfect represents the ideal analysis, a specific background event that could have had a plausible link to disinformation spread. *Rewording* is used for trigger events that are just a re-phrase of the input claim, providing no new information or analysis. *Story* labelled events are those that describe generic methods of spread (i.e. "claim shared widely on social media"), rather than thinking about what made people share it in the first place. Sometimes, this is tied to *Time Incorrect*, which represents any proposed trigger events that occurred after the initial spread, such as the creation of debunk articles. *Nonspecific* refers to trigger events that could be true for hundreds of claims rather than event-specific analysis, for example, events mentioning "general sentiment" or "ongoing disinformation around...". Finally, *Duplicate* refers to a trigger event that is almost identical to one already generated, often just worded differently.

The distribution of labels in the human-labelled dataset is shown in figure 3.7. We treat anything labelled *perfect* or with no label as a passing case, and any other label as a failure case.

[14] `https://streamlit.io/`

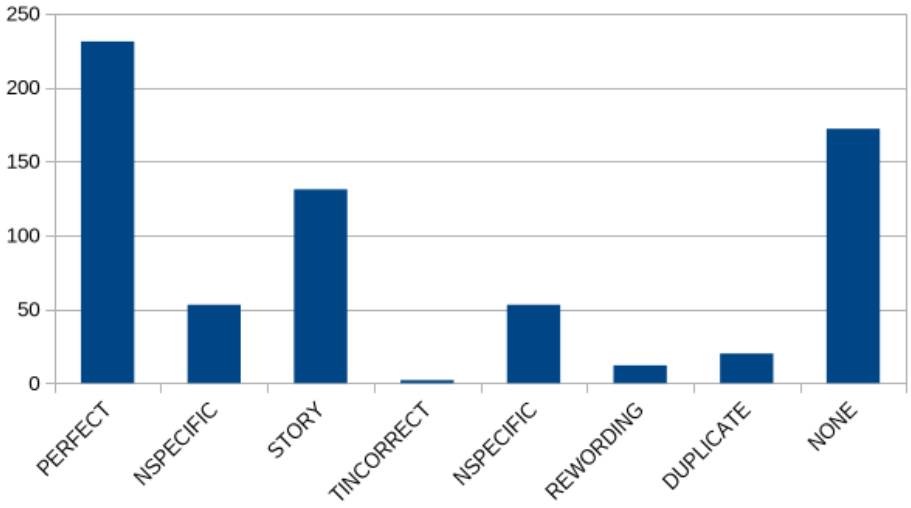


Figure 3.7: Distribution of Labels

3.5 Automation of event classification

An automated event-quality assessment system is required for the pipeline to run at scale. This process automates the removal of low-quality data from the final dataset and the evaluation of models and prompts for the experiments outlined in section 3.3.3. This behaviour can be achieved by training one or more models to classify our proposed trigger events into either a *perfect* category or a failure label. Table 3.5 provides an example of how this process would work. By subtracting the probability of a negative label from the probability of a positive, we can assign each trigger event a score. An ideal trigger event gains a positive score, whereas a *non-specific* and *time-incorrect* event is profoundly negative.

Trigger events and scores for the claim:
"A Lancet study showed COVID-19 vaccines suppress the immune system and vaccinated people have lower immune function than unvaccinated people."

Proposed Trigger Event	**Score**
Publication and publicity of studies showing reduced antibody responses in clinically immunocompromised groups (OCTAVE/related studies reported in The BMJ)	0.61
Fact-checks and debunks documenting viral social-media misinterpretations...	-0.98

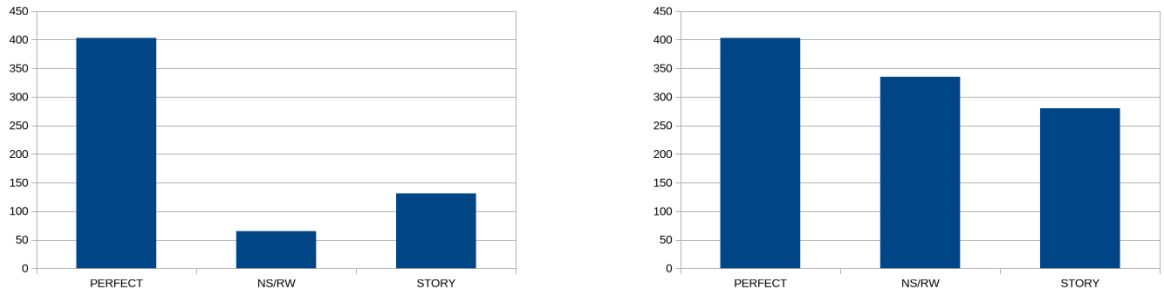
Table 3.5: Proposed Trigger Events and Automated Classifier Score

To account for data imbalance in table 3.4, the final classification task was redefined by merging low-frequency labels into one class. More specifically, *Rewording* and *Not-specific* labels were merged because, in practice, their responses were indistinguishable in the training

set, assuming the initial prompt is unknown. *Bias shown* and *Time Incorrect* were removed from the training set, since even an advanced model does not have the background knowledge to distinguish them. Additionally, with the requirement that a URL and search results have to be returned (figure 3.6), the existence of these labels is exceedingly rare and can be easily discovered by checking the status of the returned URL. *Duplicate* events were also removed, since each claim is evaluated in isolation, therefore there is no knowledge of what each event was duplicated from, making automated detection via classifier impossible. However, such events can be easily detected by checking cosine similarity of the sentence embeddings of the retrieved events. *Story* and *Perfect* were kept as their own separate labels, because of their prevalence in the dataset and clearly distinguishable word patterns. Therefore, the final classifier was trained on three event types: *Perfect*, *Not Specific/Rewording*, and *Story*.

3.5.1 Dataset Balancing

Despite our effort to merge the labels into three groups, there was still a major imbalance in label distribution (figure 3.8, a). As described in section 2.3.4, this impacts model performance unless rectified. To increase the label counts, back translation was used to double the size of the set of *Story* and *Not Specific/Rewording* labelled data, by using opus-mt models (Tiedemann et al. 2023) to translate each trigger event from English to French, and then from French back into English. For *Not Specific/Rewording* claims, an LLM was prompted to augment the dataset with extra synthetic examples (section 2.3.4). More specifically, gpt-5-nano[15] was prompted with a set of disinformation claims, and a simple few-shot prompt to create some nonspecific background information (see B.1). After the results of these two augmentation techniques were added to the dataset, the number of events in each of the three classifier labels became significantly more comparable (figure 3.8, b).



(a) Original Labeled Dataset

(b) Dataset with Augmentation

Figure 3.8: Distribution of Model Training Labels

[15] https://developers.openai.com/api/docs/models/gpt-5-mini

3.5.2 Classification Threshold Tuning

After label consolidation and dataset augmentation, we experimented with calibrating the probability threshold required to classify a label as perfect. While the standard approach is to select the label with the highest probability, we introduce a thresholding technique to enforce classification confidence. Under this method, a positive label is only assigned if its probability exceeds the combined probability of all negative labels by a fixed margin, δ. If this threshold is not met, the classifier defaults to a negative case (figure 3.9). The δ offsets used are listed in table 3.6.

Typical:
$P(y = P|event) > P(y = NS|event) \wedge P(y = P|event) > P(y = S|event)$
Ours:
$P(y = P|event) - (P(y = NS|event) + P(y = S|event)) > \delta$

Figure 3.9: Threshold Shifting
Labels: P = Perfect; NS = Not Specific; S = Story
Key: δ = configurable offset

3.5.3 Model Architecture

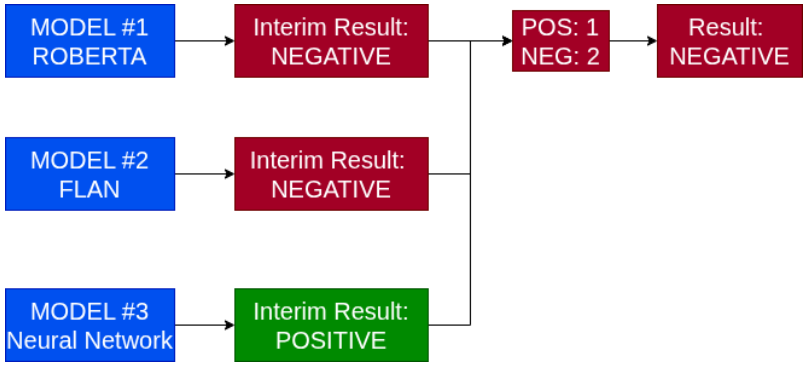
For training and evaluating each model, the human-labelled dataset was split into three sections. 400 trigger events were used to train the model, 100 to validate model performance, and 250 for final model testing.

The final model for event quality assessment represents an ensemble (see section 2.2.3) of three models: RoBERTa, FLAN, and a Feedforward Neural Network. The following sections outline how the component models were individually trained.

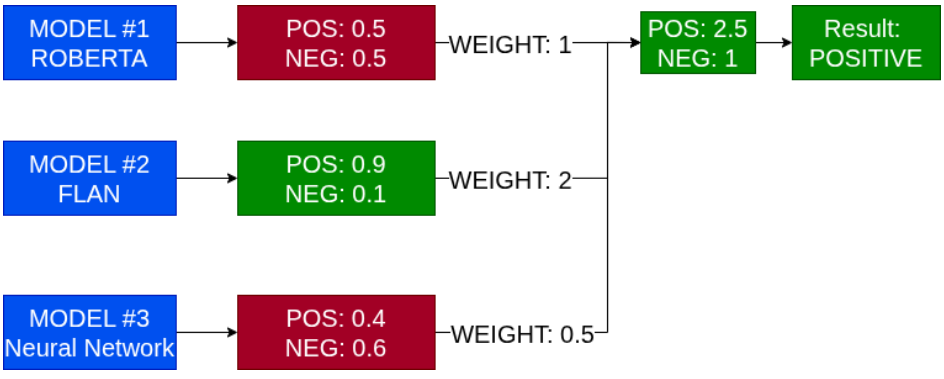
This study experimented with two methods of aggregating individual results into the final label (figure 3.10). The first option was majority voting, in which each component model is run independently, and their results are used to vote on the classification result. The second option involved retaining the scores from each run and using them to compute a weighted sum, which should be more accurate than any of the models individually. In this technique, weightings were first derived from individual classifier accuracies. Additionally, the weights were refined with a brute-force approach. The final weights used are listed in table 3.6.

Model	δ	**Weighting**
RoBERTa	0.4	0.3
FLAN	0.94	0.5
Feedforward	0.75	0.3
Sum Ensemble	0.4	N/A

Table 3.6: Model Evaluation Hyperparamters



(a) Majority Voting



(b) Weighted Sum

Figure 3.10: Ensemble Classifier Design

3.5.4 RoBERTa Model

As the first classifier, finetuned RoBERTa-base[16] was used to predict the event class. The transformers Python library[17] has built-in support for classifier refinement with RoBERTa. Once the three labels are assigned numbers from 0-2, the classification system can be integrated with minimal effort. See table 3.7 for the optimal training hyperparameters inferred on the validation set. The finetuned model is available on HuggingFace[18].

Hyperparameter	**Value**
Learning Rate	2e-5
Weight Decay	0.1
Dropout	0
Epochs	5
Batch Size	32
Loss Function	Cross-Entropy
Maximum Gradient	None
Unfrozen Layers	Last 6
Dataset Used	Balanced

Table 3.7: RoBERTa Hyperparameters

3.5.5 FLAN Model

The next model to fine-tune was FLAN. Unlike in the case of RoBERTa, we not only have to prompt (see B.2) FLAN-T5-base[19], but also decode output tokens to check whether it matches one of our pre-defined label names. This model was trained using the original, non-augmented dataset. This is to provide the training variation required across the component models to achieve a more accurate ensemble classifier (see section 2.2.3). The hope is that the large FLAN base model will reduce bias due to label prevalence. Table 3.8 demonstrates the optimal training hyperparameters for the FLAN model. The finetuned model is available on HuggingFace[20].

[16] https://huggingface.co/FacebookAI/roberta-base
[17] https://huggingface.co/docs/transformers/index
[18] https://huggingface.co/WillJeynes/LLMsForDisinformationAnalysis
[19] https://huggingface.co/google/flan-t5-base
[20] https://huggingface.co/WillJeynes/LLMsForDisinformationAnalysis-Flan

Hyperparameter	Value
Learning Rate	5e-5
Weight Decay	0.01
Dropout	0
Epochs	10
Batch Size	16
Loss Function	Standard
Maximum Gradient	1
Unfrozen Layers	All
Dataset Used	Original

Table 3.8: FLAN Hyperparameters

3.5.6 Feedforward Neural Network

The final model type utilised was a simple Feedforward Neural Network (FNN). Since it can only accept numerical inputs, the events were first tokenised using *all-mpnet-base-v2*[21], with the vector representations then used as the input nodes of the neural network. The network consisted of a hidden layer of 256 nodes and an output layer containing three nodes corresponding to the three possible labels to be given. See table 3.9 for the optimal training hyperparameters used to make the final predictions. One benefit of this model is its high dropout rate of 0.4, which helps protect against overfitting. The final model is available on HuggingFace[22].

Hyperparameter	Value
Learning Rate	2e-3
Weight Decay	1e-4
Dropout	0.4
Epochs	30
Batch Size	64
Loss Function	Cross-Entropy
Maximum Gradient	None
Unfrozen Layers	All
Dataset Used	Balanced

Table 3.9: FNN Hyperparameters

[21] `https://huggingface.co/sentence-transformers/all-mpnet-base-v2`
[22] `https://huggingface.co/WillJeynes/LLMsForDisinformationAnalysis-Regression`

3.5.7 Experimental Setup

To compare the performance of individual models to that of the model ensemble, we evaluated each of the three finetuned models and the ensemble against the gold-standard human-labelled test set from section 3.4.

While the most important metric is the overall accuracy, it is also important to consider overconfidence. We define *underconfidence* as the case in which events a human would classify as valid are not taken forward by our classifier. It can be improved by just processing more disinformation claims to make up for the missing events. *Overconfidence*, in which the pool of events is poisoned by items a human would not classify as valid, is much harder to remedy. We use precision to track this property (figure 3.11), where low precision indicates low overconfidence in the output set.

$$Precision = \frac{TruePositive}{TruePositive+FalsePositive}$$

Figure 3.11: Precision Metric

3.6 Real-World Application of the Generated Dataset

Using the best classifier from section 4.1, the best LLM from section 4.2.2, and the best prompting strategy from section 4.2.1, we can generate a large dataset from a widened set of 2000 disinformation claims. To showcase the usefulness of this dataset, two possible uses are explored in this dissertation. Firstly, we can consider reversing the dataset structure, in order to attempt disinformation prediction based on real-world events (section 3.6.1). Secondly, we can consider alternative methods of visualisation to take advantage of the dataset's structured layout (section 3.6.2).

3.6.1 Finetuning LLM for Prediction

Gaining insights into potential future disinformation claims could be vital to prebunking (see section 2.1.1). In this section, we conduct an experiment to test whether it is feasible to predict upcoming disinformation given a trigger event. This knowledge could allow for proactive prebunking campaigns to be run, meaning all viewers of the content become warned about the disinformation campaign before they are exposed to the actual disinformation, increasing the effectiveness of prebunking.

The generated dataset provides a series of links from disinformation claims to possible trigger events. Conversely, it also provides a series of links from real-world events to disinformation claims. Training a model on event-claim pairs should improve its ability to predict upcoming disinformation claims. This task can be formatted into question-answer pairs (figure 3.13), which would, in theory, be sufficient to finetune a language model for claim generation given a prompt and a trigger event. Figure 3.12 illustrates the input and output for this task.

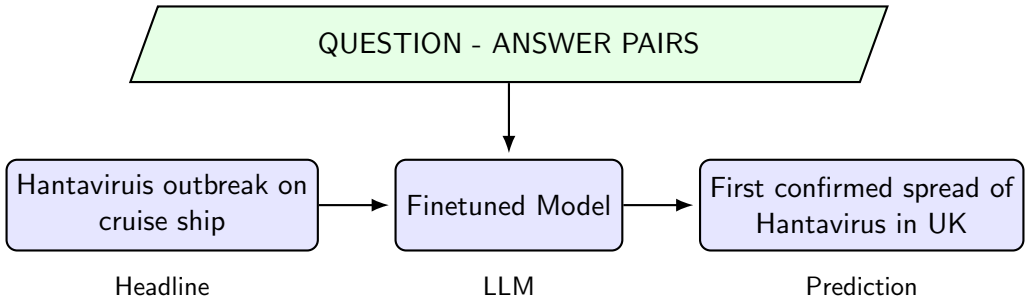


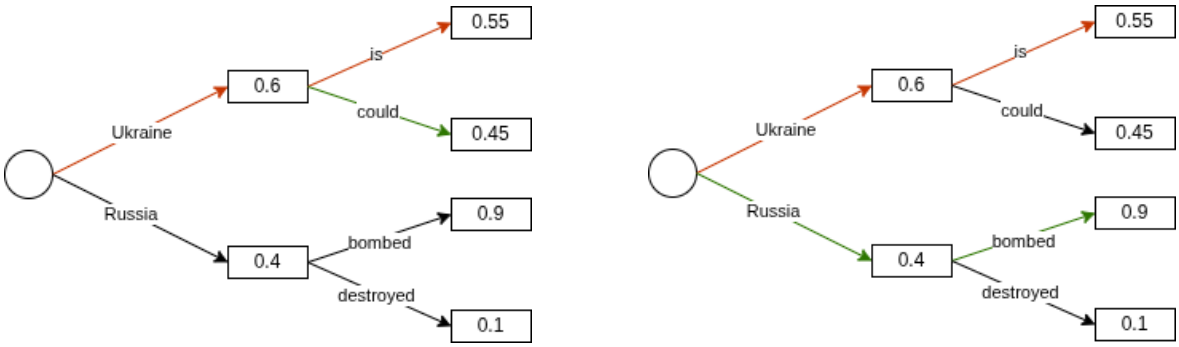
Figure 3.12: Misinformation Prediction Setup

Question: Create a disinformation claim based on the following event
Event: {LLM-GENERATED EVENT}
Claim: {NORMALIZED CLAIM}

Figure 3.13: Pre-Training Question-Answer setup

We can finetune an LLM using parameter-efficient techniques, such as LoRA (see section 2.3.3). This model should serve as a "predictor" of disinformation claims arising from real-world events. Fed with the news of the day, possible disinformation narratives will be generated, helping to narrow down which topics the fact-checkers and population should be prepared against, in advance of the actual malicious campaign. This could have a positive impact on prebunking operations, since warnings can be distributed before the first instance of the upcoming misinformation, potentially increasing efficacy. This task also has the advantage of producing a clean, consistent dataset of disinformation claims that can be used in research, as an alternative to noisy real-world data from sources like social media sites.

LLMs can generate multiple responses per question, as each generated token or word has an assigned probability. The default selection methodology is therefore greedy: select the most probable token at each time step. For this task, our aim is to collect a set of dissimilar results. To do this, the options for the first token are stored, then the next tokens are chosen greedily until the response stops. The greedy path for the second-most likely first token is considered, and so on. This provides us with a series of very different responses to analyse. The comparison between traditional beam searching (Höge 1988) and our solution can be seen in figure 3.14.



(a) Beam Search (b) Split on First Token

Figure 3.14: Extracting Differing Token Sets
Key: Red = Generation Variation 1, Green = Generation Variation 2

In this dissertation, we consider a range of possible base LLMs: distilgpt2[23], MiniLLama[24] and Deepseek R1(Guo et al. 2025). Evaluating these options against a small set of real-world headlines can help us to evaluate whether the model is fit for purpose.

Disinformation Prediction	**Description**
The Chernobyl disaster was caused by a	Not coherent
President Vladimir Putin said Cuba is Cuba's next target for invasion.	Not plausible
Donald Trump put pressure on NATO allies to pay more for defense.	Not disinformation
Orthodox Christians refuse to worship under Ukrainian rule.	Disinformation

Table 3.10: Disinformation Prediction Examples

We should first evaluate whether the output is coherent, since the smaller language models we are finetuning can output incomplete sentences or non-existent word structures. We then look for plausibility, a measure of whether the proposed idea makes logical sense in the context. For example, a country cannot wage war on itself. Finally, we consider whether the generated claim really contains disinformation, since the base model may bias it towards truth rather than our intended purpose of generating false claims. A generated disinformation claim is valid if it satisfies all three criteria. Examples of the labels can be seen in table 3.10

3.6.2 Graph Visualisation

A dataset of trigger events and claims gives researchers a brand new, structured set of information about disinformation. This section considers whether this rich structure makes the dataset useful for topic visualisation. To visualise the links between disinformation

[23] `https://huggingface.co/distilbert/distilgpt2`
[24] `https://huggingface.co/TinyLlama/TinyLlama-1.1B-Chat-v1.0`

campaigns across a topic, we first transform each claim and trigger event into a vector representation using all-MiniLM-L6-v2[25]. We then apply agglomerative clustering (Müllner 2011) to merge semantically similar claims and events together by comparing cosine similarity. The links between these merged nodes can be retained, potentially producing a range of connected components for a topic.

This graphical representation of data could serve as a useful visualisation tool for writing debunk articles. After entering a claim to counter, the system could search for it in the existing dataset or add it to the graph if it does not exist. For the input claim, the graph would show not only verified background information to consider but also links between those background events and other similar claims. These related disinformation claims not only provide further context but may also have pre-existing debunk articles associated with them, from which inspiration can be drawn.

Using a force graphing library[26], we display trigger events and claim clusters in a user-friendly manner, allowing analysts to see cause and effects on a large scale. For each of these clusters, we use the small language model, gpt-5-nano[27], to assign each cluster a summary title.

The system we propose can also generate timestamps for each returned event. Filtering claims and events based on a certain time window allows researchers to see temporal connections, visualising the transition of themes over time and identifying which chains of claims and events are most similar in time.

The most interesting metric to collect is the size of the connected components created. A connected component is a subsection of the graph, such that every vertex in the group is reachable from every other vertex in the same group. For a visualisation to work well, we need to consider components that are large enough to show repeated patterns and links, but are small enough to remain coherent. For this reason, we can filter our cluster dataset to include only clusters that are members of a subgraph of valid size.

[25] https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2

[26] https://vasturiano.github.io/react-force-graph/

[27] https://developers.openai.com/api/docs/models/gpt-5-mini

Chapter 4

Results and Discussion

This chapter presents the results of evaluating the trigger event generation pipeline. We first present the results of the trigger event classifier (section 4.1). This, in turn, allows us to quantify the accuracy of extracted trigger events at scale, enabling us to improve the prompt (section 4.2.1) and model (section 4.2.2) at the generation stage of the pipeline. Upon optimising this model, we then generate a large dataset of claim-trigger even pairs (section 4.3.1). Finally, using this dataset, we evaluate the effectiveness of the dataset using two use cases: claim prediction modelling (section 4.3.2) and visualisation (section 4.3.3).

4.1 Trigger Event Classifier Results (RQ2)

Table 4.1 demonstrates a comparison between different classifier model types, comparing their overall accuracy and precision. In terms of accuracy, both ensemble models for trigger event classification outperform their single-model counterparts. In terms of precision, the majority voting method for aggregating ensemble results significantly outperforms other methods.

Due to the balanced distribution of labels after data augmentation (see section 3.5.1), the RoBERTa classifier achieves an accuracy of 77% and a precision of 90%, comparable to that of the ensemble models. On the other hand, we observed that enriching the training data with augmented examples significantly reduced the accuracy of the FLAN model. Despite an unbalanced input set, the larger model size and prompt optimisation resulted in the highest

Model	**Accuracy (%)**	**Precision (%)**
RoBERTa	77	<u>90</u>
FLAN	79.17	85.71
Feedforward Neural Network	74.77	85.71
Ensemble Model (weighted confidence score sum)	**84.21**	83.33
Ensemble Model (majority voting)	<u>80.2</u>	**95.12**

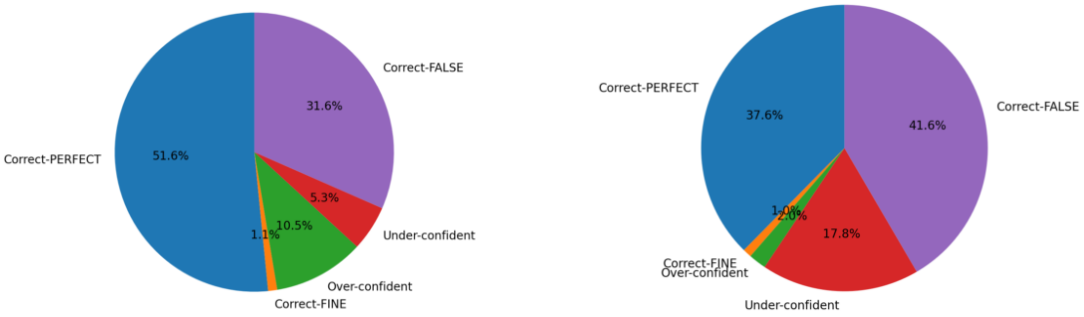
Table 4.1: Performance of Different Models Labelling the dataset
Key: **Best**, <u>Second-Best</u>

single-model accuracy of 79% for the final FLAN-T5-based classifier.

The simple feedforward neural network trained on the augmented training set produced an accuracy of 74.77%.

Finally, we combined the three best-performing single-model architectures-RoBERTa, FLAN, and a simple feedforward network-into a single ensemble classifier. Summing the respective label probabilities compounded uncertainty, resulting in the highest accuracy of 84% among the available methods. The majority voting system is a stricter model that sacrifices some accuracy for a very high precision, indicating very little overconfidence. Figure 4.1 shows a comparison of the results for the two ensemble models. Significantly more poor-quality events were identified using majority voting (label Correct-False), at the cost of increased underconfidence.

Taking into account the above-mentioned results, we applied the weighted-sum variant of the ensemble model to generate the final dataset. Its high accuracy enabled the creation of a significantly larger filtered dataset compared to the other models.



(a) Weighted Confidence Sums

(b) Majority Voting

Figure 4.1: Results of Ensemble Classifier: Weighted Sum and Majority Voting

4.2 Agent Results (RQ1)

This section explores two directions for improving the main trigger event extraction LLM. We first consider changing the prompting technique, then consider changing the model.

4.2.1 Agent Prompting Results

The first set of experiments evaluated which prompting techniques achieve the highest accuracy. As shown in table 4.2, all of our prompting techniques substantially improve result quality compared to the baseline prompt, with zero-shot CoT emerging as the most efficient approach in this setting.

The baseline results represent the scores for the prompt used to generate the initial dataset for training the classifier. With this bare-bones prompt, only 33% of the proposed trigger events were classified as positive by the verification step. This means that this version

Model	% Valid
Unedited (baseline)	33
Chain of Thought (zero-shot)	**45.51**
Plan and Solve	43.38
Self-Critique	<u>44.36</u>

Table 4.2: Performance of Different Prompting Techniques
Key: **Best**, <u>Second-Best</u>

discards much of the generated content and produces only a small number of claims with multiple trigger events.

The increase in accuracy from prompt refinement demonstrates the continued efficacy of post-training improvements in language models and underscores the vital role of prompt engineering in LLM optimisation.

Plan and Solve performed slightly worse than zero-shot CoT. This suggests that the increase of the prompt length and the context window could result in LLM forgetting the instructions (N. F. Liu et al. 2024). A similar effect is observed for the self-critique prompt: while still performing second-best, the evaluation process was not efficient enough at removing mistakes to beat the thinking elicited by zero-shot CoT. It produced no notable improvement in output quality compared to more token-efficient techniques.

4.2.2 Agent Model Results

Table 4.3 presents the comparison between different LLM types used to extract trigger events. GPT-5-mini performed best in terms of accuracy and had the second-best link accuracy. GPT-5.4-nano, as the smallest model, had the highest link accuracy. Surprisingly, the second-best model in terms of trigger-event validity was GPT-4o-mini, despite being two years older than the third-best model.

Model	% Trigger Event Valid	% Link Accuracy
gpt-5-mini	**45.51**	<u>94.57</u>
gpt-5.4-mini	32.4	89.57
gpt-5.4-nano	23.28	**97.15**
gpt-4.1-mini	27.85	94.43
gpt-4o-mini	<u>32.47</u>	90.59
llama3.1:8b-instruct-q4_K_M	0	N/A
qwen3.5:9b	0	N/A

Table 4.3: Performance of Different Models
Key: **Best**, <u>Second-Best</u>

Although gpt-5.4-mini, as the successor to gpt-5-mini, was expected to produce better results, it often skipped tool calling entirely and generated malformed JSON, resulting in a lower overall accuracy score. According to public benchmarks, the difference between the

two models is minimal[1]. The 5.4 model appears to perform slightly better overall, however, some observed behaviour suggests lower consistency and a greater emphasis on response speed over accuracy. This is broadly consistent with the public benchmark comparisons, although definitive conclusions are difficult because both models are closed-source and detailed evaluation statistics are not publicly available.

GPT-5.4-nano demonstrated the lowest trigger event accuracy, but demonstrated the best results in terms of URL accuracy. Despite being backed up by web pages, returned events were often worded vaguely and therefore filtered out by the classifier.

Running local models, Qwen and Llama, caused a series of issues with output formatting and tool calling. The JSON extraction process had to be modified to extract even a small amount of finalised analysis from either of the two models evaluated. Qwen failed to produce any output, repeatedly returning only a newline character or highly malformed results. Llama produced more consistent responses; however, they were always in an unreadable format, both during normalisation and later analysis. These models have been observed to perform worse than the GPT family of models on disinformation-adjacent tasks in prior studies (Kramer et al. 2025). However, for our task, the results contained too few interpretable outputs to evaluate link accuracy.

4.3 Large Dataset Results (RQ3)

4.3.1 Dataset generation

To generate a large-scale dataset of disinformation claim-trigger event pairs, 2000 disinformation debunk claim titles were run against the best version of the pipeline identified in sections 4.1, 4.2.1 and 4.2.2 (gpt-5-mini, with prompt A.2, evaluated using an ensemble classifier). The code for this final refined pipeline is available on GitHub[2]. From this initial set, 1906 claims had successfully yielded one or more valid trigger events, demonstrating the technique's ability to operate reliably at scale. On the other hand, 94 claim analyses generated no valid trigger events. This came from input claims that were either non-specific or insufficiently grounded in identifiable real-world events for the trigger event extraction process to be successful. Figure 4.2 shows the distribution of claims according to the number of valid trigger events generated per claim.

[1] https://aibenchy.com/compare/openai-gpt-5-4-mini-medium/openai-gpt-5-mini-medium/

[2] https://github.com/WillJeynes/LLMsForDisinformationAnalysis/

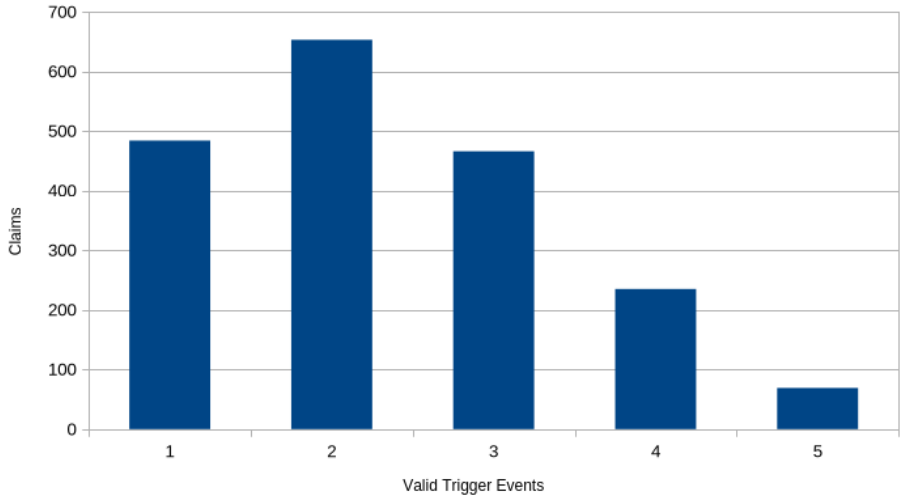


Figure 4.2: Distribution of Number of Valid Trigger Events per Input Claim

Investigating the token usage to generate the result, we can see some interesting trends. Figure 4.3 shows a series of violin plots for the distribution of total token usage per run, split by number of valid trigger events generated. Most successful generations (creating 4 to 5 valid trigger events) have a mostly even distribution between 10,000 and 120,000 tokens. Less successful iterations were significantly more biased toward low numbers of token usages, implying underutilisation of tool calling, which would lead to more generic answers, which our classifier would flag. The most extreme outliers, up to almost 600,000 tokens per claim, often resulted in a small (1-2) number of valid trigger events, implying the possibility for early stopping or penalising excessive tool use, since beyond a certain point it clearly makes no difference to output quality. The average number of input and output tokens for a run was 59,000 and 8,400, respectively.

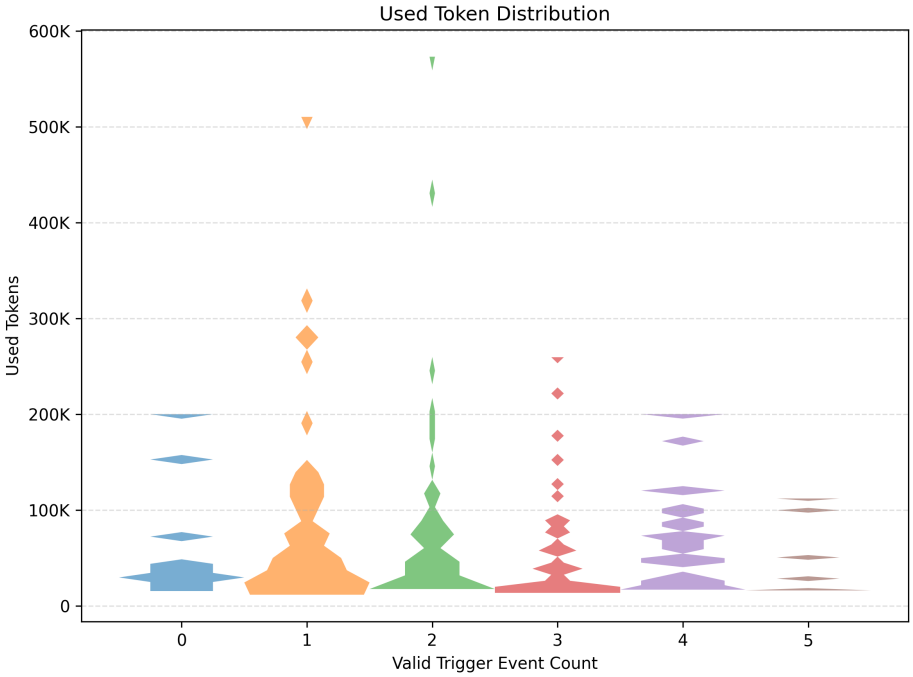


Figure 4.3: Token Usage for a Random Sample of Generations

The final large-scale dataset has been made publicly available in both a .csv and .jsonl file formats on HuggingFace[3].

4.3.2 Finetuning LLM for Prediction

Table 4.4 shows the results of the comparison between zero-shot and finetuned LLMs in the task of disinformation claim prediction based on the trigger event. The models are evaluated against our three criteria defined in section 3.6.1. As expected, the quality is very low for the zero-shot LLMs, with most outputs either refusing to generate a disinformation claim or repeating the input. This behaviour is consistent with safeguards in modern LLMs, where models are trained to avoid generating disallowed or harmful content (Ouyang et al. 2022). All tested models improved accuracy after being finetuned on the dataset, with the best being miniLLama.

Even after finetuning, we observe a considerable room for improvement. In most of our experiments, accuracy never exceeds 50%. Using LoRa on the smallest model, distilgpt2, yielded the least accurate results, with many generated responses incoherent or incomplete. As a tiny model, the base distilgpt2 model also suffers from incoherent responses; however, finetuning seemed to amplify this behaviour.

Experiments with finetuning distilLLama, which has 10 times the parameters of distilgpt2, produce a much more coherent output, while deepseek R-1, a 1.5-billion-parameter model,

[3] `https://huggingface.co/datasets/WillJeynes/LLMsForDisinformationAnalysis-Dataset`

Model/Technique	Coherence	Plausibility	Disinformation?
distilGPT2 (Vanilla)	7/9	2/9	1/9
miniLLama (Vanilla)	8/9	3/9	1/9
deepseek (Vanilla)	8/9	2/9	1/9
distilGPT2 (Tuned)	6/9	4/9	2/9
miniLLama (Tuned)	7/9	7/9	**6/9**
deepseek (Tuned)	7/9	6/9	4/9

Table 4.4: Finetuned Model Performance Comparison
Key: **Best**

produces the most coherent claims. However, the shortcomings of the training set become apparent here, with a much higher proportion of generated responses being truthful statements rather than disinformation.

As seen in figure 4.4, the models show considerable improvement against the zero-shot LLMs, when prompted on topics it is trained on. However, any of the finetuned models break down in usefulness if not queried on themes covered in the training set, in our case, Ukraine and COVID-19. Covering any other topic significantly reduces ability.

The final finetuned miniLLama model is available online[4].

Input Event:
- Donald Trump suggests using Bleach as a theoretical cure for COVID
Finetuned LLM:
- A randomised clinical trial published in the Journal of Clinical Microbiology found bleach cures COVID...
Base LLM:
- This is an entirely false claim that has been reported and shared in various conspiracy theories. There is no ...

Figure 4.4: LLM Comparison: Zero-Shot vs Finetuned for miniLLama

4.3.3 Graph Visualisation

We investigated the relations between claims and trigger events using the 2000 claim dataset. We applied agglomerative clustering with a high threshold over the trigger events and claims, resulting in 197 events and 406 claims merged into semantically similar groups. We then used these groups to compute a series of subgraphs of varying sizes. Figure 4.6 illustrates the distribution of these sizes. For our user-facing website, we present two views: a default view that contains only well-sized subgraphs, and a time-filtered view that contains the largest connected component.

[4] `https://huggingface.co/WillJeynes/LLMsForDisinformationPrediction`

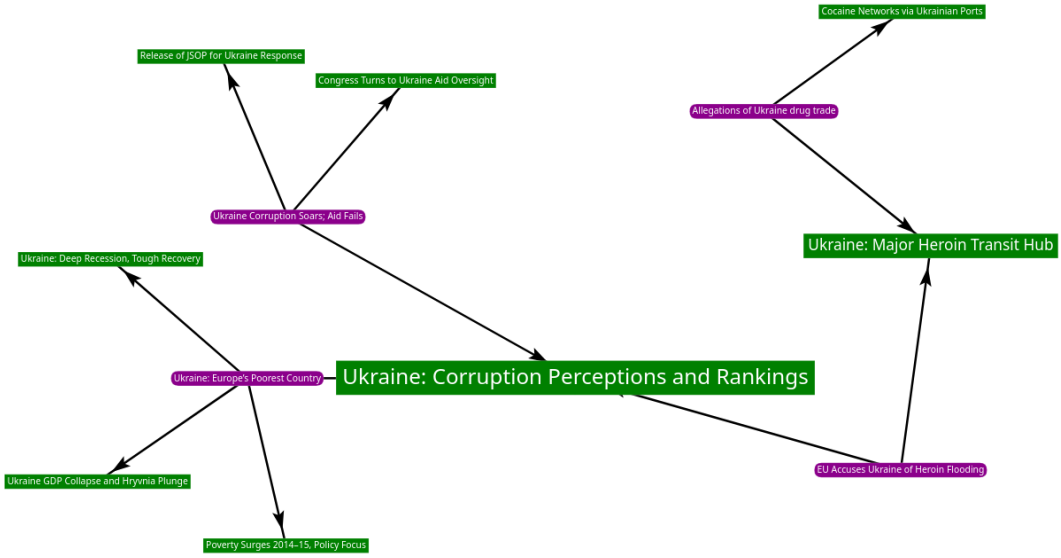


Figure 4.5: Example Web Visualisation Output
Key: Green Square = Trigger Event, Purple Rounded = Disisnformation

When designing the default view, it was revealed that the data contains a large number of tiny subgraphs, which would distract the user from the high-level analysis, and a few highly-connected subgraphs that were too large to analyse when viewed all at once. By filtering for sizes between 8 and 50 (the green area on the graph), we are left with 60 valid subgraphs to display on the front-end. Regulating the graph sizes allows complex patterns to be shown interactively and automatically. Figure 4.5 shows an example of a subgraph.

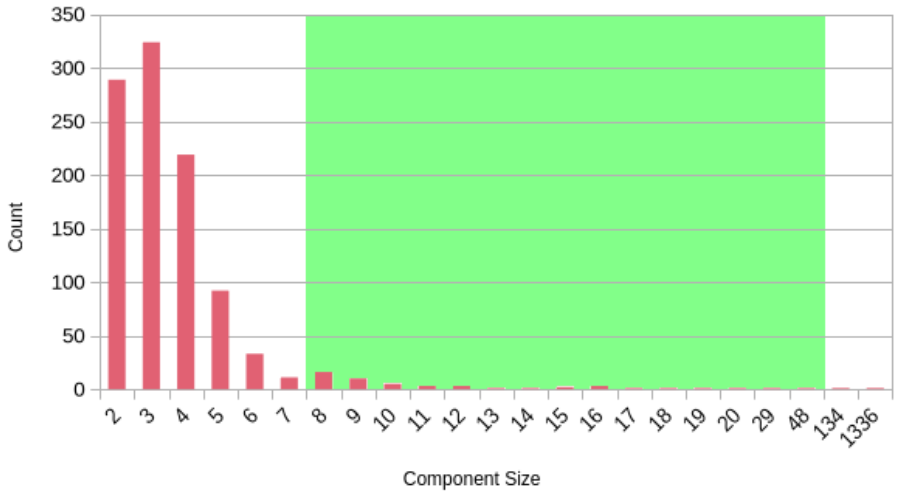


Figure 4.6: Distribution of Size of Connected Components
Green area = Visualised Component Size

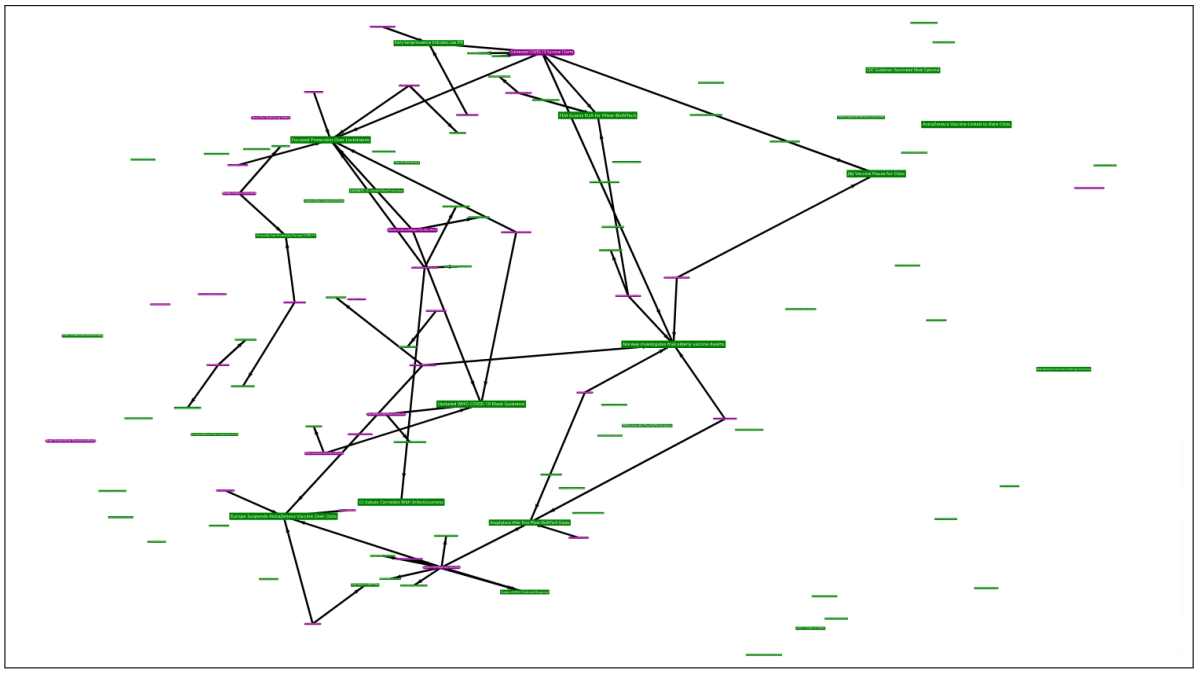
Our largest connected component, of size 1336, is too large to view without filtration. This is because the dense collection of nodes and connectors causes significant overlap, making it very difficult to determine the claim-event links accurately. By filtering the data to only include nodes from within a configurable window around a set date, we significantly reduce the number of elements active at once. This means only the most relevant links from within that period are revealed. Slowly changing the filter date allows for patterns to be seen, such as the transition from mask-based disinformation in 2020 to vaccine-based disinformation in 2021, in relation to COVID-19. We can see an example of this time shifting in figure 4.7.

As a use-case scenario, we investigated whether COVID and anti-Ukraine disinformation shared any claims or events. By evaluating the time-filtered dataset, we could see disinformation narratives such as "US using Ukraine as a testing ground for COVID Vaccines"[5] being shared across what would otherwise be two dissimilar components.

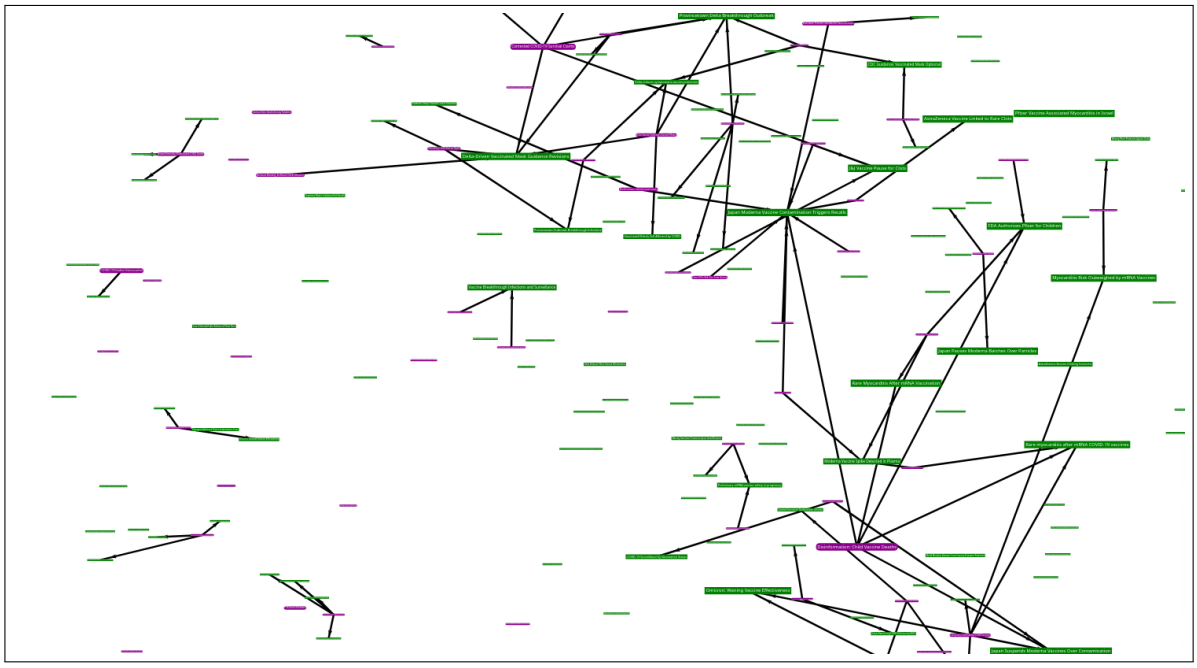
An interactive demo of both methods is available online[6].

[5] https://factcheck.afp.com/false-claim-circulates-online-united-states-testing-covid-19-vaccine-ukrainian-soldiers

[6] https://jillweynes.github.io/LLMsForDisinformationPrediction-GraphVizBuilt/



(a) Screenshot, November 2020 (±6 months)



(b) Screenshot, September 2021 (±6 months)

Figure 4.7: Results of Filtering a Large Graph by Time

Chapter 5

Conclusions

5.1 Evaluation of Research Questions

In chapter 1.1, we defined three research questions, and we believe they have all been answered successfully by this dissertation.

Answer to RQ1
We investigated whether trigger events can be extracted automatically and at scale.

By using a general-purpose LLM and a RAG with tool calling functionality, we were able to extract at least 1 trigger event from a 2000-claim dataset with a 95% success rate.

Answer to RQ2
We asked whether these proposed trigger events can be verified for their style and relevance.

To achieve this, we first manually labelled 750 trigger events. Then, using a three-part ensemble model, we trained a classifier that was able to achieve an 84% accuracy for trigger-event validity.

Answer to RQ3
We analysed what possible uses this dataset has for prediction and visualisation.

By visualising trigger-event-claim links, we demonstrate that we can build a coherent visualisation tool with emergent behaviour to explore. Additionally, we experimented with training a large language model for disinformation prediction, which saw significant improvements over zero-shot options.

5.2 Limitations

Despite this, this dissertation does not address some limitations. Firstly, while the evaluation of proposed trigger events enforces consistency, and the URL checks provide proof of existence, there are no ground-truth examples of trigger events. Since this proposed task is novel, there is no objectively correct benchmarking dataset. As a result, we could have extracted two completely different sets of trigger events, and both might be valid under our current methodology. Any future work on this topic could benefit from a human-moderated set of cross-annotated trigger events for comparison.

Additionally, our accuracy results could be high due to a potential data leakage. Since the input represents widely known debunked disinformation claims, it would enable the model's discovery of debunk articles or other forms of meta-analysis. A better long-term test set would be on unseen claims, evaluating genuine analytical performance, rather than data synthesis.

5.3 Future Work

For the moderate size of the human-labelled dataset, our trigger-event classifier is able to achieve a reasonable accuracy of 84%. Future work could involve labelling a much larger dataset of proposed trigger events, which, when used to train our existing models, should further increase accuracy. These models would enforce an even higher quality of trigger events in the final dataset.

Currently, when generating trigger events, a significant portion of proposed answers are discarded by the classifier due to receiving a negative label. Future work could investigate the use of larger open-source models, potentially finetuned for our use case. A more accurate model could not only reduce the number or irrelevant events, but also eliminate reliance on closed-source models, therefore mitigating risks associated with cost increases and model discontinuation.

While the generated dataset does well in covering most claims about COVID-19 and Ukraine, its reliance on debunk-article titles restricts us to long-standing disinformation claims on topics covered by reputable providers. Another source of disinformation, such as social media posts, would open the door to analysis of claims at the cutting edge. For this to succeed, we would need to develop accurate disinformation detection and a method for determining whether a claim is sufficiently grounded in reality to have real-world trigger events influencing it, thereby making it worthy of analysis.

The results from finetuning the large language model for disinformation prediction show that there is still much room for improvement. It would be worth training a more complex model, such as the 7B version of Deepseek R1, on a larger dataset of extracted trigger events. Additionally, currently, to select results from a set of generations, the best output is manually chosen. It would therefore be useful to train a classifier or investigate automatic evaluation metrics to assess the quality of the generated results to automatically select the best option. This would also allow us to compare different versions of a finetuned model

more accurately against a prompted zero-shot LLM at scale, extending the results presented in this dissertation.

For the best results in the visualiser, the connected components of claims and events must be within a specified size range. To moderate the size of our generated graphs, we would need to be stricter in how we create our claim and event clusters. We could finetune an embedding model to distinguish subtle differences between similar claims and events more accurately, leading to smaller clusters and, therefore, split components. Additionally, it would be interesting to gather real-world feedback from the professionals researching or countering disinformation on the usability of the visualisation tool. Such feedback would allow us to make the analysis tool a valuable, future-proof component that utilises trigger events.

We can also see several other potential future usages of the trigger event dataset, not covered in this dissertation. Context labels (Mende et al. 2023) on social media posts have been shown to be highly effective at countering disinformation online. With some refinement, our trigger events could serve as concise labels that provide background for the post and warn of potential influences. Additionally, rather than using LLMs to predict upcoming disinformation, employing some form of causal model (Peters et al. 2017) would enable more advanced statistical analysis than is possible with a black-box LLM and would yield a deeper understanding of which trigger events lead to which false claims.

5.4 Closing Remarks

In this dissertation, we aimed to demonstrate that disinformation trigger event retreival represents an important task for further research. With the use of modern LLMs and RAG, we show that we can automatically extract and evaluate them. We additionally demonstrated that such a dataset is a valuable resource that enables new data visualisation and disinformation prediction tasks. While these results are encouraging, they remain preliminary and future work is needed to evaluate the findings presented in this project more thoroughly and at a larger scale.

Bibliography

Balcaen, Pieter et al. (Feb. 2023). "The effect of disinformation about COVID-19 on consumer confidence: Insights from a survey experiment". In: *Journal of Behavioral and Experimental Economics* 102, p. 101968. ISSN: 2214-8043. DOI: 10.1016/j.socec.2022.101968. URL: http://dx.doi.org/10.1016/j.socec.2022.101968.

Bayer, Markus, Marc-André Kaufhold, and Christian Reuter (Dec. 2022). "A Survey on Data Augmentation for Text Classification". In: *ACM Computing Surveys* 55.7, pp. 1–39. ISSN: 1557-7341. DOI: 10.1145/3544558. URL: http://dx.doi.org/10.1145/3544558.

Breiman, Leo (Aug. 1996). "Bagging Predictors". In: *Machine Learning* 24.2, pp. 123–140. ISSN: 1573-0565. DOI: 10.1023/a:1018054314350. URL: http://dx.doi.org/10.1023/A:1018054314350.

Broda, Elena and Jesper Strömbäck (Mar. 2024). "Misinformation, disinformation, and fake news: lessons from an interdisciplinary, systematic literature review". In: *Annals of the International Communication Association* 48.2, pp. 139–166.

Brown, Tom B. et al. (2020). *Language Models are Few-Shot Learners*. DOI: 10.48550/ARXIV.2005.14165. URL: https://arxiv.org/abs/2005.14165.

Buda, Mateusz, Atsuto Maki, and Maciej A. Mazurowski (2017). "A systematic study of the class imbalance problem in convolutional neural networks". In: DOI: 10.48550/ARXIV.1710.05381. URL: https://arxiv.org/abs/1710.05381.

Chan, Man-pui Sally et al. (2017). "Debunking: A Meta-Analysis of the Psychological Efficacy of Messages Countering Misinformation". In: *Psychological Science* 28.11, pp. 1531–1546. ISSN: 1467-9280. DOI: 10.1177/0956797617714579. URL: http://dx.doi.org/10.1177/0956797617714579.

Chung, Hyung Won et al. (2022). *Scaling Instruction-Finetuned Language Models*. DOI: 10.48550/ARXIV.2210.11416. URL: https://arxiv.org/abs/2210.11416.

Devlin, Jacob et al. (2018). *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. DOI: 10.48550/ARXIV.1810.04805. URL: https://arxiv.org/abs/1810.04805.

El Mikati, Ibrahim K et al. (Aug. 2023). "Defining Misinformation and Related Terms in Health-Related Literature: Scoping Review". In: *Journal of Medical Internet Research* 25, e45731.

Gao, Yunfan et al. (2023). *Retrieval-Augmented Generation for Large Language Models: A Survey*. DOI: 10.48550/ARXIV.2312.10997. URL: https://arxiv.org/abs/2312.10997.

Gautret, Philippe et al. (2020). "RETRACTED: Hydroxychloroquine and azithromycin as a treatment of COVID-19: results of an open-label non-randomized clinical trial". In: *International Journal of Antimicrobial Agents* 56.1, p. 105949. ISSN: 0924-8579. DOI: 10.1016/j.ijantimicag.2020.105949. URL: http://dx.doi.org/10.1016/j.ijantimicag.2020.105949.

Gerts, Dax et al. (Apr. 2021). ""Thought I'd Share First" and Other Conspiracy Theory Tweets from the COVID-19 Infodemic: Exploratory Study". In: *JMIR Public Health and Surveillance* 7.4, e26527. ISSN: 2369-2960. DOI: 10.2196/26527. URL: http://dx.doi.org/10.2196/26527.

Guo, Daya et al. (2025). *DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning*. DOI: 10.48550/ARXIV.2501.12948. URL: https://arxiv.org/abs/2501.12948.

Hastie, Trevor, Robert Tibshirani, and Jerome Friedman (2009). *The Elements of Statistical Learning*. Springer New York. ISBN: 9780387848587. DOI: 10.1007/978-0-387-84858-7. URL: http://dx.doi.org/10.1007/978-0-387-84858-7.

Ho, Duc Hieu and Chenglin Fan (2025). *Self-Critique-Guided Curiosity Refinement: Enhancing Honesty and Helpfulness in Large Language Models via In-Context Learning*. DOI: 10.48550/ARXIV.2506.16064. URL: https://arxiv.org/abs/2506.16064.

Höge, H. (Dec. 1988). "A stochastic model for beam search". In: *Speech Communication* 7.4, pp. 423–430. ISSN: 0167-6393. DOI: 10.1016/0167-6393(88)90060-x. URL: http://dx.doi.org/10.1016/0167-6393(88)90060-X.

Houlsby, Neil et al. (2019). *Parameter-Efficient Transfer Learning for NLP*. DOI: 10.48550/ARXIV.1902.00751. URL: https://arxiv.org/abs/1902.00751.

Hu, Edward J. et al. (2021). *LoRA: Low-Rank Adaptation of Large Language Models*. DOI: 10.48550/ARXIV.2106.09685. URL: https://arxiv.org/abs/2106.09685.

Kojima, Takeshi et al. (2022). *Large Language Models are Zero-Shot Reasoners*. DOI: 10.48550/ARXIV.2205.11916. URL: https://arxiv.org/abs/2205.11916.

Kramer, Marianne, Yevgeniy Golovchenko, and Frederik Hjorth (Apr. 2025). "Detecting pro-kremlin disinformation using large language models". In: *Research and Politics* 12.2. ISSN: 2053-1680. DOI: 10.1177/20531680251351910. URL: http://dx.doi.org/10.1177/20531680251351910.

Lecheler, Sophie and Jana Laura Egelhofer (May 2022). "Disinformation, Misinformation, and Fake News". In: *Knowledge Resistance in High-Choice Information Environments*. Routledge, pp. 69–87. ISBN: 9781003111474. DOI: 10.4324/9781003111474-4. URL: http://dx.doi.org/10.4324/9781003111474-4.

Leite, João A. et al. (2025). *A Multilingual, Large-Scale Study of the Interplay between LLM Safeguards, Personalisation, and Disinformation*. DOI: 10.48550/ARXIV.2510.12993. URL: https://arxiv.org/abs/2510.12993.

Lewis, Patrick et al. (2020). *Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks*. DOI: 10.48550/ARXIV.2005.11401. URL: https://arxiv.org/abs/2005.11401.

Li, Xiang Lisa and Percy Liang (2021). *Prefix-Tuning: Optimizing Continuous Prompts for Generation*. DOI: 10.48550/ARXIV.2101.00190. URL: https://arxiv.org/abs/2101.00190.

Linden, Sander van der et al. (Jan. 2017). "Inoculating the Public against Misinformation about Climate Change". In: *Global Challenges* 1.2. ISSN: 2056-6646. DOI: 10.1002/gch2.201600008. URL: http://dx.doi.org/10.1002/gch2.201600008.

Linden, Sander van der et al. (Jan. 2026). "Prebunking misinformation techniques in social media feeds: Results from an Instagram field study". In: *Harvard Kennedy School Misinformation Review*. ISSN: 2766-1652. DOI: 10.37016/mr-2020-193. URL: http://dx.doi.org/10.37016/mr-2020-193.

Liu, Nelson F. et al. (2023). "Do Question Answering Modeling Improvements Hold Across Benchmarks?" In: *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, pp. 13186–13218. DOI: 10.18653/v1/2023.acl-long.736. URL: http://dx.doi.org/10.18653/v1/2023.acl-long.736.

Liu, Nelson F. et al. (2024). "Lost in the Middle: How Language Models Use Long Contexts". In: *Transactions of the Association for Computational Linguistics* 12, pp. 157–173. ISSN: 2307-387X. DOI: 10.1162/tacl_a_00638. URL: http://dx.doi.org/10.1162/tacl_a_00638.

Liu, Yinhan et al. (2019). *RoBERTa: A Robustly Optimized BERT Pretraining Approach*. DOI: 10.48550/ARXIV.1907.11692. URL: https://arxiv.org/abs/1907.11692.

Maclin, Richard and David Opitz (Feb. 1998). "An Empirical Evaluation of Bagging and Boosting". In: *Proceedings of the National Conference on Artificial Intelligence*.

Mavroudis, Vasilios (Nov. 2024). "LangChain". In: DOI: 10.20944/preprints202411.0566.v1. URL: http://dx.doi.org/10.20944/preprints202411.0566.v1.

Mende, Martin et al. (2023). "Fighting Infodemics: Labels as Antidotes to Mis- and Disinformation?!" In: *Journal of Public Policy, Marketing* 43.1, pp. 31–52. ISSN: 1547-7207. DOI: 10.1177/07439156231184816. URL: http://dx.doi.org/10.1177/07439156231184816.

Mikolov, Tomas et al. (2013). *Efficient Estimation of Word Representations in Vector Space*. DOI: 10.48550/ARXIV.1301.3781. URL: https://arxiv.org/abs/1301.3781.

Müllner, Daniel (2011). *Modern hierarchical, agglomerative clustering algorithms*. DOI: 10.48550/ARXIV.1109.2378. URL: https://arxiv.org/abs/1109.2378.

Nakano, Reiichiro et al. (2021). *WebGPT: Browser-assisted question-answering with human feedback*. DOI: 10.48550/ARXIV.2112.09332. URL: https://arxiv.org/abs/2112.09332.

OECD (Nov. 2022). DOI: 10.1787/37186bde-en. URL: http://dx.doi.org/10.1787/37186bde-en.

Ouyang, Long et al. (2022). *Training language models to follow instructions with human feedback*. DOI: 10.48550/ARXIV.2203.02155. URL: https://arxiv.org/abs/2203.02155.

Pavlyshenko, Bohdan M. (2023). *Analysis of Disinformation and Fake News Detection Using Fine-Tuned Large Language Model*. DOI: 10.48550/ARXIV.2309.04704. URL: https://arxiv.org/abs/2309.04704.

Peters, Jonas, Dominik Janzing, and Bernhard Schlkopf (2017). *Elements of Causal Inference: Foundations and Learning Algorithms*. The MIT Press. ISBN: 0262037319.

Pramov, Aleksandar, Jiangqin Ma, and Bina Patel (2025). *DS@GT at CheckThat! 2025: A Simple Retrieval-First, LLM-Backed Framework for Claim Normalization*. DOI: 10.48550/ARXIV.2508.17402. URL: https://arxiv.org/abs/2508.17402.

Radford, Alec and Karthik Narasimhan (2018). "Improving Language Understanding by Generative Pre-Training". In: URL: https://api.semanticscholar.org/CorpusID:49313245.

Raffel, Colin et al. (2019). *Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer*. DOI: 10.48550/ARXIV.1910.10683. URL: https://arxiv.org/abs/1910.10683.

Robertson, Stephen and Hugo Zaragoza (2009). "The Probabilistic Relevance Framework: BM25 and Beyond". In: *Foundations and Trends® in Information Retrieval* 3.4, pp. 333–389. ISSN: 1554-0677. DOI: 10.1561/1500000019. URL: http://dx.doi.org/10.1561/1500000019.

Robles-Pagán, Moisés and Manuel Rodriguez-Martinez (Nov. 2025). "Fighting Health-Related Misinformation in Social Media with Large Language Models". In: *Proceedings of the International Symposium on Intelligent Computing and Networking 2025*. Springer Nature Switzerland, pp. 330–349. ISBN: 9783032096944. DOI: 10.1007/978-3-032-09694-4_26. URL: http://dx.doi.org/10.1007/978-3-032-09694-4_26.

Rosenblatt, F. (1958). "The perceptron: A probabilistic model for information storage and organization in the brain." In: *Psychological Review* 65.6, pp. 386–408. ISSN: 0033-295X. DOI: 10.1037/h0042519. URL: http://dx.doi.org/10.1037/h0042519.

Sahoo, Pranab et al. (2024). *A Systematic Survey of Prompt Engineering in Large Language Models: Techniques and Applications*. DOI: 10.48550/ARXIV.2402.07927. URL: https://arxiv.org/abs/2402.07927.

Sawarkar, Kunal, Abhilasha Mangal, and Shivam Raj Solanki (2024). "Blended RAG: Improving RAG (Retriever-Augmented Generation) Accuracy with Semantic Search and Hybrid Query-Based Retrievers". In: DOI: 10.48550/ARXIV.2404.07220. URL: https://arxiv.org/abs/2404.07220.

Sennrich, Rico, Barry Haddow, and Alexandra Birch (2015). *Improving Neural Machine Translation Models with Monolingual Data*. DOI: 10.48550/ARXIV.1511.06709. URL: https://arxiv.org/abs/1511.06709.

Shao, Chengcheng et al. (2016). "Hoaxy: A Platform for Tracking Online Misinformation". In: DOI: 10.48550/ARXIV.1603.01511. URL: https://arxiv.org/abs/1603.01511.

Souly, Alexandra et al. (2024). *A StrongREJECT for Empty Jailbreaks*. DOI: 10.48550/ARXIV.2402.10260. URL: https://arxiv.org/abs/2402.10260.

Sundriyal, Megha, Tanmoy Chakraborty, and Preslav Nakov (2023). *From Chaos to Clarity: Claim Normalization to Empower Fact-Checking*. DOI: 10.48550/ARXIV.2310.14338. URL: https://arxiv.org/abs/2310.14338.

Swire-Thompson, Briony, Joseph DeGutis, and David Lazer (2020). "Searching for the backfire effect: Measurement and design considerations." In: *Journal of Applied Research in Memory and Cognition* 9.3, pp. 286–299. ISSN: 2211-3681. DOI: 10.1016/j.jarmac.2020.06.006. URL: http://dx.doi.org/10.1016/j.jarmac.2020.06.006.

Al-Tarawneh, Mutaz A. B. et al. (2024). "Enhancing Fake News Detection with Word Embedding: A Machine Learning and Deep Learning Approach". In: *Computers* 13.9, p. 239. ISSN: 2073-431X. DOI: 10.3390/computers13090239. URL: http://dx.doi.org/10.3390/computers13090239.

Tay, Li Qian et al. (2024). "Do prebunking and debunking protect against novel misinformation?" In: DOI: 10.31234/osf.io/w7gja. URL: http://dx.doi.org/10.31234/osf.io/w7gja.

Tian, Katherine et al. (2023). *Fine-tuning Language Models for Factuality*. DOI: 10.48550/ARXIV.2311.08401. URL: https://arxiv.org/abs/2311.08401.

Tiedemann, Jörg et al. (2023). "Democratizing neural machine translation with OPUS-MT". In: *Language Resources and Evaluation* 58, pp. 713–755. ISSN: 1574-0218. DOI: 10.1007/s10579-023-09704-w.

Tokita, Christopher K et al. (2024). "Measuring receptivity to misinformation at scale on a social media platform". In: *PNAS Nexus* 3.10. Ed. by Yannis Yortsos. ISSN: 2752-6542. DOI: 10.1093/pnasnexus/pgae396. URL: http://dx.doi.org/10.1093/pnasnexus/pgae396.

Traberg, Cecilie S. et al. (2023). "Prebunking Against Misinformation in the Modern Digital Age". In: *Managing Infodemics in the 21st Century*. Springer International Publishing, pp. 99–111. ISBN: 9783031277894. DOI: 10.1007/978-3-031-27789-4_8. URL: http://dx.doi.org/10.1007/978-3-031-27789-4_8.

Trivedi, Harsh et al. (2022). *Interleaving Retrieval with Chain-of-Thought Reasoning for Knowledge-Intensive Multi-Step Questions*. DOI: 10.48550/ARXIV.2212.10509. URL: https://arxiv.org/abs/2212.10509.

Truică, Ciprian-Octavian and Elena-Simona Apostol (Feb. 2022). "MisRoBÆRTa: Transformers versus Misinformation". In: *Mathematics* 10.4, p. 569. ISSN: 2227-7390. DOI: 10.3390/math10040569. URL: http://dx.doi.org/10.3390/math10040569.

ValizadehAslani, Taha et al. (Aug. 2024). "Two-stage fine-tuning with ChatGPT data augmentation for learning class-imbalanced data". In: *Neurocomputing* 592, p. 127801. ISSN: 0925-2312. DOI: 10.1016/j.neucom.2024.127801. URL: http://dx.doi.org/10.1016/j.neucom.2024.127801.

Vaswani, Ashish et al. (2017). *Attention Is All You Need*. DOI: 10.48550/ARXIV.1706.03762. URL: https://arxiv.org/abs/1706.03762.

Vosoughi, Soroush, Deb Roy, and Sinan Aral (Mar. 2018). "The spread of true and false news online". In: *Science* 359.6380, pp. 1146–1151. ISSN: 1095-9203. DOI: 10.1126/science.aap9559. URL: http://dx.doi.org/10.1126/science.aap9559.

Walter, Nathan and Riva Tukachinsky (2019). "A Meta-Analytic Examination of the Continued Influence of Misinformation in the Face of Correction: How Powerful Is It, Why Does It Happen, and How to Stop It?" In: *Communication Research* 47.2, pp. 155–177. ISSN: 1552-3810. DOI: 10.1177/0093650219854600. URL: http://dx.doi.org/10.1177/0093650219854600.

Wang, Lei et al. (2023). *Plan-and-Solve Prompting: Improving Zero-Shot Chain-of-Thought Reasoning by Large Language Models*. DOI: 10.48550/ARXIV.2305.04091. URL: https://arxiv.org/abs/2305.04091.

Wang, Wenhui et al. (2020). *MiniLMv2: Multi-Head Self-Attention Relation Distillation for Compressing Pretrained Transformers*. DOI: 10.48550/ARXIV.2012.15828. URL: https://arxiv.org/abs/2012.15828.

Wei, Jason et al. (2022). *Chain-of-Thought Prompting Elicits Reasoning in Large Language Models*. DOI: 10.48550/ARXIV.2201.11903. URL: https://arxiv.org/abs/2201.11903.

Wei, Zeyu et al. (2025). *Shadows in the Attention: Contextual Perturbation and Representation Drift in the Dynamics of Hallucination in LLMs*. DOI: 10.48550/ARXIV.2505.16894. URL: https://arxiv.org/abs/2505.16894.

Williams, Evan M. and Kathleen M. Carley (Feb. 2023). "Search engine manipulation to spread pro-Kremlin propaganda". In: *Harvard Kennedy School Misinformation Review*. DOI: 10.37016/mr-2020-112. URL: http://dx.doi.org/10.37016/mr-2020-112.

Zhang, Zhuosheng et al. (2023). *Multimodal Chain-of-Thought Reasoning in Language Models*. DOI: 10.48550/ARXIV.2302.00923. URL: https://arxiv.org/abs/2302.00923.

Appendices

Appendix A

Trigger Event Generation Prompts

Below are the prompts used to generate the Trigger Events in the analysis pipeline. Any line starting with # is a comment and is removed from the final prompt. Additionally, any ### represents string replacement.

A.1 Baseline

```
You are an agent in a pipeline to analyse disinformation.
Once the information has been created as below, a dataset can be created to feed a model for
prediction, which will improve pre-bunking efforts.

There is a false disinformation claim circulating:
###NTITLE###
Produce up-to 5 specific events that happened that have led to the spread of this disinformation.

Remember the time frame of the disinformation campaign: ###CDATE###

Produce no more text other than the json.

Include a concise but specific search query that can be looked up on a search engine in order to allow
for the verification.

Include a url to a source for your trigger event (not a web search, a specific url from a reputuable
source). Do not use OAI cite, include url as text in response.

Use a JSON format with each entry containing "Event,ReasoningWhyRelevant,SearchQuery,Url".

Multiple tool invocations should be requested at once, if applicable.
```

A.2 Chain of Thought (Zero-shot)

```
You are an agent in a pipeline to analyse disinformation.
Once the information has been created as below, a dataset can be created to feed a model for
prediction, which will improve pre-bunking efforts.

There is a false disinformation claim circulating:
###NTITLE###
```

```
Produce up-to 5 specific "trigger events" that happened that could have led to the spread of this
disinformation.

Remember the time frame of the disinformation campaign: ###CDATE###
Include no information or events that would not have been available at the time.

Produce no more text other than the json.

Include a concise but specific search query that can be looked up on a search engine in order to allow
for the verification.

Include a url to a source for your trigger event (not a web search, a specific url from a reputuable
source). Do not use OAI cite, include url as text in response.

Include the date that the event happened ("March 2022" for exmaple)

Use a JSON format with each entry containing "Event,ReasoningWhyRelevant,SearchQuery,Url,Date".
Return ONLY valid JSON.
Do not include explanations.
Do not wrap in markdown.
Do not label where the json is using colons

Multiple tool invocations should be requested at once, if applicable.
Use your abilities to look between the lines and produce some insightful analysis, thinking both short
and long term.

Events will be reordered as part of processing, each statement must stand alone

The preceeding messages act as examples of previous responses to potentially ficitonal events and
scores given.
Analysis should only be completed for proposed events that would graner >0.7 points

Since URLs change frequently, use tools to retreive up to date informaiton everytime, provided examples
or existing knowledge will be wrong or out of date.

Lets go through it step by step
```

A.3 Plan and Solve

```
You are an agent in a pipeline to analyse disinformation.
Once the information has been created as below, a dataset can be created to feed a model for
prediction, which will improve pre-bunking efforts.

There is a false disinformation claim circulating:
###NTITLE###
Produce up-to 5 specific "trigger events" that happened that could have led to the spread of this
disinformation.

Remember the time frame of the disinformation campaign: ###CDATE###
Include no information or events that would not have been available at the time.

Produce no more text other than the json.

Include a concise but specific search query that can be looked up on a search engine in order to allow
for the verification.
```

```
Include a url to a source for your trigger event (not a web search, a specific url from a reputuable
source). Do not use OAI cite, include url as text in response.

Include the date that the event happened ("March 2022" for exmaple)

Use a JSON format with each entry containing "Event,ReasoningWhyRelevant,SearchQuery,Url,Date".

Multiple tool invocations should be requested at once, if applicable.
Use your abilities to look between the lines and produce some insightful analysis, thinking both short
and long term.

Events will be reordered as part of processing, each statement must stand alone

The preceeding messages act as examples of previous responses to potentially ficitonal events and
scores given.
Analysis should only be completed for proposed events that would graner >0.7 points

First, consider a range of directions in which the proposed disinformation could have been influenced
by.
Then, research these directions in turn, using the tools at hand.
Finally, refine your proposed "trigger event" until it is specific, quantifiable and backed up by
evidence.
```

A.4 Self-Critique

A.4.1 Initial Generation

```
You are an agent in a pipeline to analyse disinformation.
Once the information has been created as below, a dataset can be created to feed a model for
prediction, which will improve pre-bunking efforts.

There is a false disinformation claim circulating:
###NTITLE###
Produce up-to 5 specific "trigger events" that happened that could have led to the spread of this
disinformation.

Remember the time frame of the disinformation campaign: ###CDATE###
Include no information or events that would not have been available at the time.

Produce no more text other than the json.

Include a concise but specific search query that can be looked up on a search engine in order to allow
for the verification.

Include a url to a source for your trigger event (not a web search, a specific url from a reputable
source). Do not use OAI cite, include url as text in response.

Include the date that the event happened ("March 2022" for exmaple)

Use a JSON format with each entry containing "Event,ReasoningWhyRelevant,SearchQuery,Url,Date".

Multiple tool invocations should be requested at once, if applicable.
Use your abilities to look between the lines and produce some insightful analysis, thinking both short
and long term.

Events will be reordered as part of processing, each statement must stand alone
```

```
The preceeding messages act as examples of previous responses to potentially ficitonal events and
scores given.
Analysis should only be completed for proposed events that would garner >0.7 points
```

A.4.2 Evaluation

```
You are an impartial and meticulous evaluator assessing LLM's response based on key quality dimensions
of honesty and usefulness. Your goal is to provide structured feedback that can be used to improve the
response.
Evaluation task: please follow these steps carefully:
1. Analyze the response based on the three dimensions below.
2. Provide justifications first: write a brief explanation justifying your assessment for each
dimension.
3. Assign scores after justification: assign a score from 1 (poor) to 10 (excellent) for each dimension
based on your justification.
4. Synthesize: provide a brief overall impression and the single most important suggestion for improve-
ment, keeping in mind that explanation/honesty is the top priority, then followed by guidance.
Critique dimensions (evaluate in this order):
(1) Speficicity and usefullness: Can the proposed event be used to create a dataset of concrete events
mapped to later
disinformation.
(2) Existance: Using the context provided, can the user be certain that the proposed trigger event
actually happened
(3) Causality: Is there a possible link from the proposed trigger event to the disinformaiton at hand
Overall impression & key improvement suggestion: Briefly summarize the overall quality and state the
most critical change needed to improve the response.

Disinformation query:
###NTITLE###
Disinformation date:
###CDATE###

LLM's response to evaluate:
###LM###

Provided context:
###VESEARCHES###
```

A.4.3 Post-Evaluation

```
You are an expert editor tasked with making targeted improvements to an existing LLM's response based
on a specific critique with the primary goal of enhancing its score according to evaluation standards
while
preserving its strengths.
Your revision task: generate a revised version of the existing response. Your goal is not to rewrite it
completely, but to make precise edits only to address the specific weaknesses highlighted in the
critique.
Instructions for editing:
- Identify specific flaws: carefully read the critique and pinpoint the exact issues raised (e.g.,
unclear
explanation, vagueness, inappropriate responses, the key suggestion).
- Perform minimal targeted edits: modify only the necessary sentences or paragraphs within the existing
response to directly fix these identified flaws.
- Strongly preserve strengths: crucially keep all other parts of the existing response intact. Do not
```

```
rephrase, restructure, or remove sections that were not criticized or likely contributed positively to
its
initial score.
- Ensure coherence: verify that your targeted edits integrate smoothly and do not introduce
contradictions
or awkward phrasing.
Output requirements:
- It should feel like a slightly polished or corrected version of the existing response, not a
fundamentally
different answer.
- Do not mention the critique, scores, or the editing process. The output should be clean json that
passes validation checks

Again, use a JSON format with each entry containing "Event,ReasoningWhyRelevant,SearchQuery,Url,Date".
Use tools available to you if further information is required

Add no new events, only improve the existing items

Disinformation query:
###NTITLE###
Disinformation date:
###CDATE###

LLM's response to improve:
###L2M###

Citique:
###LM###

This contains specific feedback, justifications, scores from 1 to 10, and potentially a key improvement
suggestion. Focus on the justifications for low scores and the key suggestion.
```

Appendix B

Other Prompts

B.1 Extra Claim Example Generation

```
Provide a non specific piece of background, talking point or other misinformation that allowed the a
disinformation to spread; to aid in analysis and debunking
The topic in question is: ###CLAIM###
Include just the example no other text.
Good examples would be ###EXAMPLE1### and ###EXAMPLE2###

Be concise, make no mistakes, use similar style and wording to provided examples
```

B.2 FLAN-T5 Classifier Prompt

```
Classify the following event into one of these categories: perfect, story, not specific.
Event: ###EVENT###
Category:
```

B.3 Normalization Prompt

```
You are part of an agent in a process to tack state-sponsored disinformation

In order for the following debunk articles to be automatically referenced below an offensive post, the
main offensive statement should be extracted, so it can be run in a semantic matcher

Some of the data comes from debunk datasets, please remove any references to that

Reduce this title from a disinformation tracking api to a short concise claim

Make all parts of the claim definite
For example:
Something could have potentially happened BECOMES something happened
DISINFORMATION CLAIM: something is NOT true BECOMES something is true

Relevent examples are included in preceeding messages, use these as exact inspiration.

The claim to normalize is:
###TITLE###

Produce no other text other than the condensed claim.
```

Appendix C

Random Sample of Trigger Event Dataset

Claim: Ukraine bans conscripts from enrolling in universities.

Field	**Content**
Original	Ukraine To Ban Conscripts From Enrolling to Universities
Date	April 21, 2023
Trigger Events	• Ukrainian government approved a draft law narrowing eligibility for military deferments for students and educators • Verkhovna Rada extended martial law and general mobilization and described categories (including students) eligible/exempt from mobilization

Claim: US intelligence found reasons to pursue regime change in Ukraine.

Field	**Content**
Original	US Intelligence Finds Reasons for Regime Change in Ukraine
Date	February 17, 2018

Trigger Events	• Documented U.S. support for Ukrainian civil society and opposition groups via NGOs and grants (e.g., National Endowment for Democracy activities in Ukraine around 2013–2014). • Victoria Nuland public remark that the U.S. has invested about $5 billion in Ukraine since 1991. • Removal of President Viktor Yanukovych after mass Euromaidan protests; Yanukovych flees Kyiv and parliament votes to remove him.

Claim: A court ruled PCR COVID-19 tests are 97% inaccurate and unfit for purpose, and mainstream media did not report this.

Field	**Content**
Original	"MSM Silent As Court Holds PCR Covid Tests 97% Inaccurate - Unfit for Purpose"
Date	May 06, 2024
Trigger Events	• Lisbon Court of Appeal ruling (Portugal) referenced PCR test reliability in a quarantine case • Early peer-reviewed studies showing correlation between PCR cycle threshold (Ct) and viral culture/infectivity (e.g., Bullard et al.) published in 2020 • WHO information notice reminding labs to follow manufacturers' instructions and consider Ct thresholds (WHO notice for IVD users) • CDC announced it would withdraw its EUA request for the original CDC-only RT-PCR panel (laboratory notice, mid-2021)

Claim: Great Barrier Reef sea surface temperature has not changed in the past 150 years.

Field	Content
Original	"Great Barrier Reef Sea Surface Temperature: No Change In 150 Years"
Date	August 09, 2021
Trigger Events	• March 2016 — Large, well-documented mass coral bleaching across the Great Barrier Reef (AIMS/GBRMPA surveys) • 2019 — 'State of the Climate 2019' (WMO/BoM/CSIRO) and related national summaries documenting global and regional ocean heating and record ocean heat content

Claim: Poland has ceased payments to Ukrainian refugees in 2024.

Field	Content
Original	Poland to Cease Payments to Ukrainian Refugees Since 2024
Date	November 27, 2023

Trigger Events	• International Rescue Committee (IRC) and other NGOs issued warnings and reports (late Sep 2023) expressing concern about legal uncertainty and the future of support for Ukrainians in Poland. • Polish government spokesman Piotr (Piotr) Müller publicly said Poland will likely not extend special support measures for Ukrainian refugees into 2024. • Poland's Social Insurance Institution (ZUS) suspended childcare/family benefit payments for thousands of Ukrainian refugees who had left Poland. • Multiple news outlets (e.g., Kyiv Independent) and regional reporting reproduced Polish officials' remarks that support may be curtailed in 2024, producing headline-driven coverage in mid-September 2023.

Claim: Magnetic pole reversals cause the Earth to flip vertically and stop rotating for six days, causing cataclysmic events.

Field	**Content**
Original	Magnetic poles reversals involve the Earth flipping vertically and momentarily stopping its rotation, causing cataclysmic events during 6 days.
Date	January 18, 2023
Trigger Events	• NASA/NOAA public messaging about Solar Cycle 25 and increased space weather activity • Out-of-cycle update to the World Magnetic Model (WMM2020) after rapid movement of the magnetic north pole

Claim: 80% of people who received the Moderna COVID-19 vaccine experienced significant side effects, and children in Sudan contracted polio from the polio vaccine.

Field	**Content**
Original	80% of people taking the Moderna vaccine had significant side effects; children in Sudan got polio from the polio vaccine
Date	March 26, 2021
Trigger Events	• Several countries temporarily suspended use of the AstraZeneca COVID-19 vaccine in mid-March 2021 while regulators investigated rare clotting reports. • FDA Vaccines and Related Biological Products Advisory Committee briefing document for Moderna (mRNA-1273) published, including detailed safety/reactogenicity tables from phase 3 trial. • Public-facing CDC guidance summarizing common, expected side effects after COVID-19 vaccination (local pain, fatigue, headache, fever). • UNICEF and WHO confirmation and public statements about a circulating vaccine-derived poliovirus type 2 (cVDPV2) outbreak in Sudan and coordinated vaccination response.

Claim: The two largest banks in Ukraine will default.

Field	**Content**
Original	Default by Ukraine's Two Largest Banks Is Inevitable
Date	July 11, 2015

Trigger Events	• IMF approval of a large Extended Fund Facility for Ukraine (US$17.5 billion) in March 2015, accompanied by IMF warnings about the banking sector's contingent liabilities and need for recapitalization. • National Bank of Ukraine (NBU) extended loan repayment periods and implemented emergency liquidity measures to support banks under strain. • Credit-rating agency downgrades of major Ukrainian banks (e.g., Fitch downgraded PrivatBank and ProCredit's viability ratings in December 2014). • Ukraine's hryvnia plunged after the central bank abandoned foreign currency auctions, triggering sharp currency devaluation and market panic. • Central bank closures and a wave of bank shutdowns during the 2014 financial turmoil in Ukraine (multiple banks shut or declared insolvent).

Claim: A review found that COVID-19 mRNA vaccines aid cancer development.

Field	**Content**
Original	A review "has found that COVID-19 mRNA vaccines could aid cancer development"
Date	April 16, 2024
Trigger Events	• Radiology literature (2021) documenting vaccine-associated axillary/supraclavicular lymphadenopathy and FDG PET/CT uptake • Publication of a case report describing rapid progression of marginal zone B-cell lymphoma following BNT162b2 vaccination (Frontiers in Medicine)

Claim: The bodies of 286 women were found near Krasnoarmiisk.

Field	Content
Original	Bodies of 286 Women were Found near Krasnoarmiisk
Date	November 01, 2014
Trigger Events	• Human Rights Watch reported exhumation of a mass grave in Sloviansk (eastern Ukraine). • OSCE Special Monitoring Mission published observations about unmarked graves found near Komunar / Nyzhnia Krynka (Donetsk region).

Claim: A study showed Pfizer's COVID-19 vaccine weakened children's immune responses to other viruses.

Field	Content
Original	A study showed a "weakened immune response in kids injected with Pfizer's COVID-19 shot"; "The mRNA Covid jabs damage immune responses to other viruses in children"
Date	August 30, 2023
Trigger Events	• Japan finds stainless-steel/foreign particles in multiple Moderna vaccine vials leading to recalls (September 2021) • Widespread pediatric RSV and other respiratory virus surges after COVID-19 restrictions lifted (late 2022) • Noted rise in invasive Group A Streptococcus (iGAS) and scarlet fever among children in late 2022–early 2023 • Publication showing systemic mRNA COVID-19 vaccines induce variable or limited mucosal (IgA) responses

Claim: Higher COVID-19 vaccination rates among White people are associated with higher COVID-19 death rates.

Field	Content
Original	Higher COVID-19 vaccination rates in White people are related to higher COVID-19 death rates
Date	August 25, 2023
Trigger Events	• CDC MMWR and other US reports documenting racial/ethnic differences in vaccine uptake (published analyses in early 2023 showing complex patterns of uptake across age and race). • Office for National Statistics (ONS) release showing that, for some reporting periods, the majority of COVID-19 deaths in England occurred among vaccinated people (because most older people were vaccinated).

Claim: COVID-19 vaccines are dangerous for children; they cause myocarditis and bypass children's innate immunity that protects them from COVID-19.

Field	Content
Original	"clearly [COVID-19] vaccines are dangerous for kids", they cause myocarditis; children are protected from COVID-19 by innate immunity which is bypassed by vaccination
Date	March 28, 2023
Trigger Events	• Nordic countries (Sweden, Denmark, Norway, Finland) limited or paused use of Moderna in younger people citing myocarditis signals (Oct 2021) • CDC/ACIP report and guidance noting rare myocarditis cases after mRNA COVID-19 vaccination (June 2021) • FDA emergency authorization of Pfizer-BioNTech COVID-19 vaccine for children 5–11 (Oct 29, 2021) • MedRxiv/academic preprint showing children have stronger pre-activated innate airway antiviral responses to SARS-CoV-2 (June 2021)

Claim: Half of the 0.9°C of global warming is caused by greenhouse gases.

Field	Content
Original	So that means that probably about half, maybe half of that nine-tenths of the degree [of total warming] might be caused by greenhouse gases
Date	October 21, 2018
Trigger Events	• Publication of Otto et al., 'Energy budget constraints on climate response' (Nature Geoscience, May 2013), which produced observational energy-budget estimates of climate sensitivity on the lower end of model ranges. • IPCC AR5 Summary for Policymakers (Sep 2013) stated it is "extremely likely" that more than half of the observed increase in global average surface temperature from 1951 to 2010 was caused by anthropogenic greenhouse gases.

Claim: Kyiv is one of the 10 worst cities to live in.

Field	Content
Original	Kyiv is Among Top 10 Worst Cities to Live In
Date	June 25, 2023

Trigger Events	• U.S. State Department and other governments maintained high-level 'do not travel' advisories for Ukraine • UNHCR published guidance and cautioned about voluntary return amid ongoing safety, protection and infrastructure concerns in 2023 • Kyiv's mayor reported that a large share of the city's population fled after the Russian invasion • Widespread Russian strikes damaged Ukraine's energy infrastructure, causing prolonged blackouts in Kyiv

Claim: Kyiv forcibly drafted ethnic Hungarians.

Field	**Content**
Original	Kyiv Forcibly Drafts Ethnic Hungarians
Date	February 09, 2026

Trigger Events	• Hungary repeatedly raised public complaints and blocked aspects of Ukraine's Western integration over alleged mistreatment of the Hungarian minority in Zakarpattia, increasing bilateral tensions. • Reporting and analyses documented both an exodus of ethnic Hungarians from Transcarpathia after the 2022 invasion and cases of ethnic Hungarians voluntarily joining Ukrainian forces or local defense units. • Ukraine's Security Service (SBU) and Ukrainian authorities reported Russian information-operations specifically targeting the ethnic Hungarian community in Transcarpathia / Zakarpattia (including spoofed calls and other PSYOPs). • Ukraine imposed wartime restrictions and a de facto travel ban preventing most men aged 18–60 from leaving the country after the 2022 full-scale invasion.