EX-FEVER: A Dataset for Multi-hop Explainable Fact Verification

Anonymous ACL submission

Abstract

Fact verification aims to automatically probe the veracity of a claim based on several pieces of evidence. Existing works are always engaging in accuracy improvement, let alone explainability, a critical capability of fact verification systems. Constructing an explainable fact verification system in a complex multi-hop scenario is consistently impeded by the absence of a relevant, high-quality dataset. Previous datasets either suffer from excessive simplification or fail to incorporate essential considerations for explainability. To address this, we present EX-FEVER, a pioneering dataset for multi-hop explainable fact verification. With over 60,000 claims involving 2-hop and 3-hop reasoning, each is created by summarizing and modifying information from hyperlinked Wikipedia documents. Each instance is accompanied by a veracity label and an explanation that outlines the reasoning path supporting the veracity classification. Additionally, we demonstrate a novel baseline system on our EX-FEVER dataset, showcasing document retrieval, explanation generation, and claim verification, and validate the significance of our dataset. Furthermore, we highlight the potential of utilizing Large Language Models in the fact verification task. We hope our dataset could make a significant contribution by providing ample opportunities to explore the integration of natural language explanations in the domain of fact verification.

1 Introduction

Fact verification, also known as fact checking, is a task to predict the veracity of a claim based on retrieved evidence, i.e., evidence that supports the claim, refutes the claim, or has insufficient information to judge the claim. Since the misinformation is widely spread with the proliferation of social platforms, recent years have witnessed the rapid development of automatic fact checking over various domains, such as politics (Ostrowski et al., 2021; Wang, 2017; Sofia University "St. Kliment Ohridski", Bulgaria et al., 2017), public health (Kotonya and Toni, 2020; Shahi and Nandini, 2020; Nakov et al., 2022), and science (Lazer et al., 2018; Wadden et al., 2020).

A typical fact-checking system consists of two main stages: evidence retrieval and veracity prediction. The evidence retrieval stage aims to improve the recall of golden evidence. The veracity prediction is made based on the interaction between the given claim and the retrieved evidence.

The first large-scale fact-checking dataset, FEVER (Thorne et al., 2018), has significantly contributed to the promotion of existing works in the field where claims are annotated by crowd workers and rely on information sourced from Wikipedia articles. The fact-checking systems inspired by FEVER aim to enhance both the performance (e.g., precision and recall) of evidence retrieval and the accuracy of verdict prediction (Zhou et al., 2019; Liu et al., 2020). However, it is important to acknowledge that over 87% of the claims in FEVER rely on information sourced from a single Wikipedia article. In contrast, real-world claims often involve information from multiple sources, making it challenging for single-hop models to reason effectively without resorting to word-matching shortcuts (Jiang and Bansal, 2019). To address this limitation, the HOVER dataset is introduced by Jiang et al. (2020). The HOVER dataset is derived from the QA dataset (Yang et al., 2018), where claims require evidence from up to four English Wikipedia articles. The multi-hop design presents numerous challenges for both retrieval and verification models, but its oversimplification of the verification task as a binary classification is a major reason for its limited adoption in the fact verification task.

Though the research on multi-hop complex reasoning, the explainability is still under-explored. Explainability plays a significant role in fact-

checking models for two main reasons. Firstly, displaying the veracity prediction along with the corresponding textual explanations can make the fact-checking system more credible to human users (Atanasova et al., 2020; Kotonya and Toni, 2020). Moreover, the performance (e.g., F1 score, Accuracy) of models can be improved when the explanation is fed (Stammbach and Ash, 2020). Hence, shedding light on the research of explainability in fact checking holds great value.

Datasets	Hops	Explainable	Class
HOVER	2-3-4	✗	2
FEVER	1-2	✗	3
e-FEVER	1-2	✓	3
EX-FEVER	2-3	✓	3

Table 1: Related Datasets Comparison

Currently, only a few works in fact checking take textual explanations into consideration. The datasets LIAR-PLUS (Alhindi et al., 2018) and PUBHEALTH (Kotonya and Toni, 2020) involve about 10,000 claims with journalists' comments as explanations in politics and public health respectively, which are quite limited regarding the scale and the domain. Besides, though e-FEVER (Stammbach and Ash, 2020) complements textual explanations for the dataset FEVER using GPT-3, the quality of automatically generated explanations cannot be guaranteed. Furthermore, verifying the claims in all mentioned datasets always requires only one piece of information, i.e., one-hop fact-checking, where lots of methods have been proposed and achieved remarkable success (Zhou et al., 2019; Liu et al., 2020). However, generating precise explanations in complex, multi-hop fact-checking scenarios remains an open and challenging question.

To facilitate the development of research around the aforementioned question, we propose the first dataset for multi-hop explainable fact verification, namely EX-FEVER. In general, the dataset involves 60,000 claims requiring 2-hop or 3-hop reasoning. Each claim is accompanied by two or three golden documents containing the necessary information for veracity reasoning. Different from existing datasets focusing on evaluating performance solely, a claim is assigned with not only a veracity label (SUPPORTS, REFUTES, and NOT ENOUGH INFO) but also a piece of golden explanation. The textual explanation describes the minimally sufficient information in each hop to verify a claim. Take Figure 2 as an example, the veracity of the claim requires 3-hop reasoning, with each color representing the information from a different hop. Note that the explanation is possibly a segment of a complete sentence in the golden document and requires necessary rewriting since we want to keep the minimally sufficient information and other unrelated information will not be included, which is different from annotating a whole sentence as golden evidence in previous datasets. We employed crowd workers and more details can be seen in Section 3.

Then we develop a baseline system and perform a comprehensive benchmark evaluation in Section 4 which encompasses document retrieval, explanation generation, and verdict prediction. Through this evaluation, we achieve the following insights: The effectiveness of the multi-hop design retrieval model (Xiong et al., 2021), and retrieval is a critical bottleneck in the fact-checking system. The weakness of the graph-based verdict prediction model (Zhou et al., 2019). The limitations of existing fact-checking models trained on previous datasets such as HOVER (Jiang et al., 2020). In Section 5, we do some preliminary investigations using Large Language Models (LLMs) in two distinct ways: using LLMs as an actor and using LLMs as a planner finding that LLMs excel as a planner and generating explanations, rather than directly making predictions.

In a nutshell, our contributions can be listed as follows,

- We propose the first dataset for multi-hop explainable fact verification, which can support and promote the development of such a challenging domain.
- We develop a baseline system to demonstrate the practical application of the dataset in various areas, including document retrieval, explanation generation, and verdict prediction.
- We explore the potential of leveraging LLMs for fact verification in two distinct ways providing valuable insights into their effectiveness and applicability in this context.

2 Related Work

In this section, we briefly review representative works in the field of fact verification and model

Claim	John Mayer is an American singer-songwriter whose debut EP was later re-released by an American record label owned by Sony Music Entertainment.
Golden Explanation	John Mayer is an American singer-songwriter who released his first extended play, Inside Wants Out. Inside Wants Out is the debut EP by John Mayer that was later re-released by Columbia Records. Columbia Records is an American record label owned by Sony Music Entertainment.
Golden Document	John Mayer, Inside Wants Out, Columbia Records
Label	**SUPPORT**

Figure 1: A sample in the proposed dataset EX-FEVER. The textual explanation in different colors refers to the information in different documents.

explainability.

2.1 Fact Verification

Fact verification is a task similar to natural language inference, where the target is to predict whether evidence entails a claim. There are several models carefully designed for veracity reasoning based on retrieved evidence, including transformer-based methods (Jiang et al., 2021; Kruengkrai et al., 2021) and graph-based methods (Zhou et al., 2019; Liu et al., 2020; Zhong et al., 2020). Existing works can be roughly grouped into two categories aiming to enhance performance and robustness, respectively.

Though precision-targeted models achieve satisfactory performance, some researchers discovered the precision improvement is derived from dataset biases, where these models would suffer from significant performance decline under different data distributions, i.e., the poor robustness (Schuster et al., 2019). To this end, several unbiased datasets Symmetric (Schuster et al., 2019), FEVER2.0 (Thorne et al., 2019), and FM2 (Eisenschlos et al., 2021) are proposed to evaluate the model robustness.

While the research on precision and robustness achieved remarkable results, they were not of explainability. Different from the former two problems, this paper focuses on the explainability of the fact-checking system, which still lacks a fundamental benchmark, including both a dataset and an evaluation system.

2.2 Model Explainability

The generation of natural language explanations is treated as one of the ways to reveal the mechanism inside the 'black-box' deep learning model(Ribeiro et al., 2016; Camburu et al., 2018; Jolly et al., 2022). Generally, there are two pipelines, i.e., extractive approaches (Yang et al., 2018; Atanasova et al., 2020; Inoue et al., 2021) and abstractive approaches (Kotonya and Toni, 2020).

In real-life applications, the extracted approaches may contain redundant information and pronouns lacking context, which is not ideal. Therefore, the abstractive approach, which aims to provide concise and contextually understandable explanations by filling in the pronouns with an understanding of the context, has become a more popular and reasonable method for achieving explainability.

Despite its importance and growing research interests, there are only a few datasets in fact verification considerate non-extracted approaches. Fact-checking datasets LIAR-PLUS (Alhindi et al., 2018) and PUBHEALTH (Kotonya and Toni, 2020) involve about 10,000 claims with journalists' comments as the explanation in politics and public health respectively, which is quite limited regarding the scale and the domain. Although Stammbach and Ash (2020) made efforts to generate textual explanations for the FEVER dataset using GPT-3, automatic machine-generated interpretation text cannot guarantee quality. In contrast, our EX-FEVER offers a comprehensive platform on a large scale, offering high-quality human-annotated explanations in the intricate multi-hop scenario.

3 Data Collection

In this section, we provide a detailed description of the specific steps involved in annotating claims, including their corresponding labels and explanations.

3.1 Claim Annotation

Given a set of evidence documents, we initially require annotators to write the SUPPORTS claim first, and then create the corresponding REFUTES claim and NOT ENOUGH INFO claim by modifying the SUPPORTS claim. By doing so, we obtain three distinct claims, each labeled differently. These claims are based on a set of 2 or 3 golden documents, where the number of documents corresponds to the information hops involved in the claims. For brevity, we refer to these claims as **SUPPORTS**, **REFUTES**, and **NOT ENOUGH INFO** claims.

SUPPORTS Claim Creation. In this class, claims must incorporate information from related golden documents, which are interconnected via hyperlinks. Annotators replace phrases in the initial document with their corresponding descriptions from hyperlinked documents, creating a 2-hop claim. Similarly, if there are three documents, this process is iterated to form a 3-hop claim. For instance, consider the following example: the Wikipedia article for "Bohemian Rhapsody" states that *Bohemian Rhapsody" is a song by the British rock band Oucen. It was written by Freddie Mercury for the band's 1975 album Night at the Opera*. Correspondingly, the Wikipedia page for "A Night at the Opera" further elucidates that *A Night at the Opera is the fourth studio album by the British rock band Queen, released on 21 November 1975 by . . . .* In this context, the generated claim would read as follows: *"Bohemian Rhapsody" is a song by the British rock band Queen for their1975 music album that was released on 21 November 1975.*

REFUTES Claim Creation. To generate contradictory REFUTES claims, we follow a process of modifying SUPPORTS claims. Similar to previous fact-checking datasets (Jiang et al., 2020), we utilize several mutation methods to ensure dataset diversity. These methods include:

- *Entity Replacement:* Entities in a claim play a crucial role in determining its veracity. By substituting the entity in a SUPPORTS claim with an unrelated entity, we can transform it into a REFUTES claim. This change creates a semantic conflict with the evidence documents. Common entities that are frequently replaced include names, places, and organizations. For instance, consider the original claim: *Love & Mercy was about the leader of an Australian rock band consisting of an American musician, singer, and songwriter.* By changing the country from American to Australian, the claim now conflicts with the evidence documents.

- *Logical Word Replacement:* This method tests the fact-checking model's ability to reason logically. It involves replacing logical words, such as comparative adjectives or temporal phrases, to alter the semantics in the opposite direction. For example, the word 'smaller' can be replaced with 'larger' to reverse the meaning. Similarly, temporal phrases like "during the 1960s" can be modified to "before the 1960s" or "after the 1960s." Additionally, specific time periods can be replaced with other relevant time frames.

- *Negation Word Insertion/Removal:* This method aims to assess the model's comprehension of negations within a sentence. Annotators are instructed to remove negation words, such as 'not' and 'never,' from SUPPORTS claims if they exist. Alternatively, negation words can be inserted into the SUPPORTS claim, or adjectives can be substituted with their antonyms. It's important to note that while it is necessary to include some REFUTES claims using this method, caution must be exercised. Some researchers have observed that models tend to establish spurious relationships between negations and the 'REFUTES' label, leading to biased and unreliable predictions. Therefore, annotators are encouraged to primarily utilize the first two methods, resorting to this method only if necessary. This approach helps maintain dataset balance and mitigate biases.

NOT ENOUGH INFO Claim Creation. There are two methods to create the claim in this class from both the evidence side and the claim side. Firstly, we randomly remove one of the shown documents to create a lack of evidence information. Then, the NOT ENOUGH INFO claim is the same as the SUPPORTS claim. The second method is to write a claim unrelated to the evidence information and keep the documents unchanged.

For 2-hop claims, we all utilize the second method since there are only two documents, and removing one of them will leave only one piece of information, which is easy for models to detect. For 3-hop claims, we randomly employ two methods to generate NOT ENOUGH INFO claims with the same probability.

3.2 Explanation Annotation

After writing claims, we ask annotators to write textual explanations that demonstrate how to reach the veracity label based on the evidence.

The Explanation for SUPPORTS Claim contains minimally sufficient information to draw a supported conclusion. Specifically, there are two (three) sentences for a 2-hop (3-hop) claim in explanation, where each sentence is the summarization of claim-related information in each golden document. Such an explanation describes the reasoning path among the given documents.

The Explanation for REFUTES Claim. Different from the explanation in the class SUPPORT, it not only involves the summarization of each golden document, but also points out which part of the claim is inconsistent with the documents. For instance, the claim is that *"Bohemian Rhapsody" is a song by the British rock band Queen for their 1975 music album that was released on 21 December 1975*, the corresponding explanation would read as follows: *"Bohemian Rhapsody" is a song by the British rock band Queen from their 1975 album A Night at the Opera. A Night at the Opera was released on 21 November 1975, not 21 December 1975*. The explanation not only describes the reasoning process but also pinpoints the specific parts of the claim that are inconsistent with the relevant golden documents.

The Explanation for NOT ENOUGH INFO Claim. Since there are two types of NOT ENOUGH INFO claims generated via different methods, the corresponding explanation is also distinct. In detail, for claims whose evidence documents are randomly removed, we ask annotators to indicate which piece of the necessary information is dropped and involve information in the preserved documents in the explanation. For claims unrelated to the evidence document, the explanation should be *there is no information to verify the claim.* For example, the claim is *Louise Simonson won an award for Outstanding Achievement in Comic Arts*, the corresponding explanation will be *There is no information showing that Louise Simonson won an award for Outstanding Achievement in Comic Arts*.

3.3 Overall Annotating Process

We employed the annotators from Appen[1] and supplied them with our detailed annotation guidelines. We first identify a seed article from the top 50,000 popular Wikipedia pages to initiate the process. Then we spend two weeks training those annotators from our feedback after the quality is consistent and high enough.

In order to harvest a high-quality dataset, we employed additional annotators as quality inspectors to scrutinize the annotations submitted by the primary annotators. Further details can be found in A.1.

3.4 Data characteristic

In summary, we collected over 60,000 claims featuring three distinct labels while maintaining label balance, with each label constituting approximately 33% of the dataset. The label and hop count distribution is described in table 2. We divide the dataset into training, validation, and test subsets based on a 70%-20%-10% split. The claim mean length and the explanation length at the word-level are shown in table 2. We also randomly extract a mini-test data subset from the full test dataset, comprising 1,000 claims, and utilize this to test the capabilities of large language models (e.g., ChatGPT).

Table 2: Data Statistics with different number of hops and different label classes. The average claim length and explanation length in word level are reported

Hops	SUP	REF	NEI	Claim	EXP
2 Hops	11053	11059	11412	21.63	28.39
3 Hops	9337	9463	8941	30.69	43.45
Total	20390	20522	20353	25.73	35.21

4 Baseline System Description

Our baseline system comprises three stages. Firstly, given a claim, the system aims to provide the most relevant documents as evidence. Secondly, the system summarizes the information from multiple documents into a concise summary, which then serves as the system's output explanation. Finally, based on the interaction between the claim and the summary, the system generates a verdict. In the subsequent sections, we will delve into the specific models employed in each stage.

Document Retrieval At this stage, given a claim c, the objective is to retrieve relevant documents $D = d_1, \cdots, d_N$. Firstly, we employ a rule-based document retrieval based on the DrQA system (Chen et al., 2017), using cosine similarity between binned unigram and bigram Term FrequencyInverse Document Frequency (TF-IDF) vec-

[1] https://www.appen.com

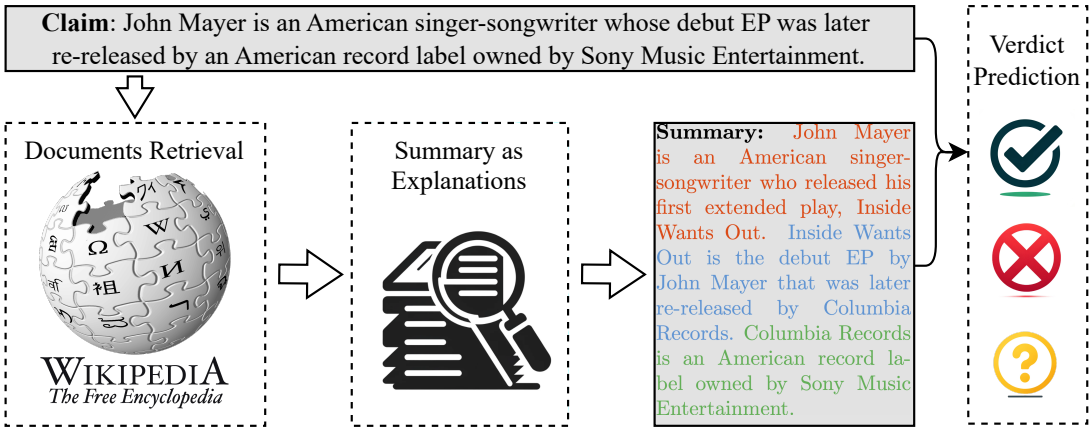

Figure 2: The baseline system comprises three stages: document retrieval, summary generation as explanations, and verdict prediction. The system produces two main outputs: a veracity label indicating whether the claim is 'SUPPORT'ed, 'REFUTE'd, or there is 'NOT ENOUGH INFO', and a summary that serves as an explanation for the prediction.

tors. Then we use two neural-based document retrieval models, the BERT-based model (Kenton and Toutanova, 2019) and the MDR model (Xiong et al., 2021). The BERT-based model calculates the similarity score between each document d and the given claim c. The MDR model incorporates a multi-hop retrieval design, which iteratively selects a document based on the probability modeled through a query reformulation process conditioned on pre-retrieval results. Finally, we select the top-5 documents feeding into the next stage.

Explanatory Stage This stage aims to use the model to understand the relevant documents and produce a piece of minimally sufficient summarization as the system explanation. We fine-tune a BART (Lewis et al., 2020) model, which is the state-of-the-art model in text generation and summarization tasks, with the given supervised golden explanation on our dataset's training split.

Verdict Predict In this stage, the system needs to make a verdict based on the interaction between the claim and the generated summary explanation. We adapt a transformer-based method BERT(Kenton and Toutanova, 2019). As a comparative alternative, we implement a graph-based text reasoning model, the state-of-art fact-checking model GEAR (Zhou et al., 2019).

Aug-HOVER To further evaluate the importance of our dataset, we conduct training on the previous dataset HOVER (Jiang et al., 2020), but the system is tested on our own dataset. In this approach, the EX-FEVER data is not used for training the verification model. Instead, it is exclusively utilized for validation and testing purposes. The verification model is trained solely on the Hover dataset. Since the Hover dataset consists of binary class labels, we merge the "NOT ENOUGH INFO" class and the "REFUTE" class from EX-FEVER, creating a unified "NOT_SUPPORTED" class for evaluation. The remaining data is aligned with the Hover dataset.

4.1 Experimental results

Table 3: Retrieve Model Performance Comparison

Model	EM	Hit@6	Hit@12	Hit@30
MDR	43.3	55.00	60.90	68.60
BERT-based	32.4	66.12	70.28	73.98

Document Retrieval Our evaluation metrics of choice are the exact match (EM) score and the hit score. The experimental results are presented in Table 3, demonstrating that the MDR model achieves a better EM score, while the BERT-based model obtains a higher hit score. **It is possible that in scenarios involving multiple hops, iterative information retrieval (MDR) is effective in achieving a better EM score. Additionally, the hit score improves marginally with an increasing number of hits.**

Explanatory Stage We choose the Rouge score as an evaluation metric to assess the quality of the generated explanation which measures the similarity between the golden explanation and the generated explanation. The results are summarized in Table 4. When fed into the documents retrieved

Table 4: Generated Summary Metrics Comparison

Model	Length	rouge1	rouge2	rougeL	rougeLsum
MDR	54.79	54.88	41.34	49.42	53.02
BERT-based	46.05	46.88	32.80	35.52	44.41
Explanation from ChatGPT					
GPT-0example	58.05	52.28	33.74	48.13	49.89
GPT-3example	48.56	59.98	42.85	57.66	55.61

Table 5: Verify Model Comparison. The accuracy (%) of each model is reported

Model	Val	Test	Test On Golden	Train With Golden
Gear@BERT-based	54.96	54.71	53.08	61.05
Gear@MDR	59.68	58.89	53.98	-
BERT@BERT-based	68.07	67.65	76.69	99.29
BERT@MDR	73.86	73.34	76.89	-
HOVER@MDR	46.58	45.41	33.79	-

from the MDR model, BART achieves a superior rouge score, revealing that **the retrieval model is the bottleneck of the system.** Moreover, this observation further validates the importance of the EM score, as we cannot simply increase the number of hits due to the input length constraint of the text-generating model.

Verdict Predict Table 5 displays the final verdict outcomes. The BERT model outperforms the graph-based method GEAR in terms of accuracy, whether using the MDR retrieval model or the BERT-based retrieval model, by a significant margin. To further evaluate the models, we conducted tests using golden explanations, which represent an ideal scenario assuming a perfect explanation-generating model. Surprisingly, the performance of the GEAR model did not improve. **This finding suggests that the graph method relies more on pattern recognition rather than genuine reasoning capabilities.** When the input data shifted from generated explanations to golden explanations, the GEAR model failed to show any improvement.

Furthermore, we devised an approach for training the verdict prediction model using golden explanations. Although this setting is not practical for real-world usage, it allows us to assess the quality of our golden explanations. In this scenario, the verdict prediction model achieved optimum accuracy, indicating the high quality of our golden explanations.

Aug-HOVER The experimental results of Aug-HOVER are also showcased in Table 5. The obtained test results demonstrate a performance of only approximately 45%, indicating a subpar outcome. **This suggests that the verification model trained on the previous dataset may not be capable of addressing the task we proposed which further validates the importance of our dataset.**

5 Prompt based approach

With the PLMs (pre-trained language models) and GPT model series development (Brown et al., 2020; Radford et al., 2018, 2019; Ouyang et al., 2022), large language models (LLMs) exhibit immense potential in many general tasks, especially with suitable prompts (Wei et al., 2022, 2023). In this section, we do some preliminary investigations exploring using LLMs in the fact checking task in two directions: directly using LLMs as an actor, and using LLMs as a planner also we both evaluate the verdict accuracy and the ability of LLMs to generate explanations.

5.1 LLMs as an actor

In this approach, we directly prompt an LLM as an actor instructing ChatGPT to directly make a verdict with a given claim. We evaluate both its ability in verdict prediction and explanation generation. We use a variety of prompt templates which include only giving the claim, giving the claim with the golden documents, adding the few shot examples, giving the instruction in json format, and instructing GhatGPT with or without requiring explanation, and details are shown in A. We also extract the explanation from LLMs's responses to evaluate the explanation quality.

5.2 LLMs as a planner

In this approach, we adopt the methodology outlined by (Pan et al., 2023). Following their setting, we instruct ChatGPT to generate program guides for each individual claim. Subsequently, we employ a verification model to execute these program guides, thereby obtaining the verdict prediction for each claim.

5.3 Results dicussion

Table 6: Use LLM as an actor or a planner. The accuracy (%) of each model is reported.

Type	Model	Close	Open	Gold
Actor	ClaimOnly	45.78	-	-
	w/o exp	-	-	47.91
	w/ exp	-	-	47.92
	1 shot	-	-	47.91
	3 shots	-	-	58.69
Planner	ProgramFc	47.30	51.70	64.90

The verdict accuracy of the two paradigms is reported in table 6, and to evaluate the quality of the explanation generated from ChatGPT, we use Rouge score, and the test result is appended in table 4.

In the context of employing the "LLMs as an actor" paradigm, when we provide only the claim without relevant evidence documents, necessitating LLMs to rely on their internal knowledge for predictions, ChatGPT yields the lowest favorable result with an accuracy of 45.78%. **This outcome may suggest that despite the vast training data LLMs have been exposed to, they still need extra knowledge to perform this task.** When we introduce a few shot examples to assist LLMs, a noticeable enhancement in performance becomes apparent. Furthermore, as the quantity of provided examples increases, there is a corresponding improvement in performance, from 47.91% to 58.68%. **This suggests that the incorporation of few-shot, in-context learning proves to be an effective approach for addressing the task at hand in our study.**

Nonetheless, within the "LLMs as a planner" paradigm, we do not directly obtain verdict results from ChatGPT itself. Instead, we solely rely on the large model to generate program guides, while the verification model employed is a non-finetuned 6-billion-parameter FLan-T5 model(Chung et al., 2022). Surprisingly, this approach yields substantial improvements in performance.

In contrast to the "LLMs as an actor" approach, when provided with golden evidence, the accuracy increases from 58% to 64%. **This intriguing phenomenon appears to suggest that the large model excels not in making predictions but rather in serving as a planner, generating guides to facilitate judgments by other models.**

Then, we conducted a test using ChatGPT to generate explanations for after making the veracity predictions. The results of this test are appended in Table 4. We observe that ChatGPT's performance in explanation generation improves as the number of input examples increases. When we provide three examples, ChatGPT outperforms the fine-tuned Bart model. **That suggests that ChatGPT performs better in the explanation generation task compared to making predictions for claims.**

6 Conclusion

We introduce a publicly accessible, extensive fact-checking dataset, named EX-FEVER, encompassing over 60,000 intricate multi-hop claims. For each claim's veracity, we provide an elucidating annotation to facilitate human comprehension of the adjudication process. We devise a comprehensive system using EX-FEVER, encompassing a retrieval phase, a summarization component serving as the explanatory stage, and a subsequent verification stage. The experiment results validate the challenge and the importance of EX-FEVER. Furthermore, we do some preliminary investigations exploring using LLMs in the fact checking task. We elect the GPT-3.5-turbo model to represent large language models. Utilizing two distinct settings using LLMs as an actor and using LLMs as a planner, we evaluated the LLM on our dataset mini-test set. We find that the LLM exhibits better performance when utilized as planners rather than directly employed as actors and that LLM excels in generating explanations rather than making predictions. The findings reveal that despite the prowess of LLMs, there remains significant potential for improvement. In summary, EX-FEVER could serve as a valuable benchmark in studying the explainable multi-hop fact-checking task by improving reliability, trustworthiness, and facilitating better decision-making across different domains.

Limitations

Although the intention behind proposing this dataset is to bridge the gap between human experts and automated fact verification systems in terms of explainability, evaluating the effectiveness of the system remains challenging. In this paper, we utilize the Rouge score to assess the system's output explanation quality. However, to evaluate the output explanation quality the Rouge score is insufficient. There are instances where different lexical choices in the explanation could yield the same result, but the Rouge score, which is based on similarity, may penalize such cases, leading to inaccurate evaluation results. **Hence, further investigation is warranted to explore an evaluation scoring mechanism that goes beyond mere similarity and effectively assesses the quality of explanations.**

Another limitation is that in the retrieval stage, a situation arises where information can be sourced from more than one Wikipedia document. While we have deliberately simplified and overlooked this scenario to enhance the manageability of the annotation process, it's important to acknowledge that models might receive undue penalties when retrieving information from these sources, particularly when the retrieved documents aren't in the golden documents.

Ethics Statement

Biases. Our data is collected based on the top 50,000 popular Wikipedia pages. Our process does not introduce any additional biases, although there may be inherent biases present in Wikipedia that are beyond our control.

Intended Use and Misuse Potential. Our dataset can be valuable for enhancing the development of auto-fact-checking systems in areas such as document retrieval, document summaries, and verdict prediction. To the best of our knowledge, there is no specific potential for misuse associated with this dataset, or at least no more potential for misuse than other fact verification datasets.

References

Tariq Alhindi, Savvas Petridis, and Smaranda Muresan. 2018. Where is your evidence: Improving fact-checking by justification modeling. In *Proceedings of the First Workshop on Fact Extraction and VERification (FEVER)*, pages 85–90.

Pepa Atanasova, Jakob Grue Simonsen, Christina Lioma, and Isabelle Augenstein. 2020. Generating Fact Checking Explanations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7352–7364, Online. Association for Computational Linguistics.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners.

Oana-Maria Camburu, Tim Rocktäschel, Thomas Lukasiewicz, and Phil Blunsom. 2018. e-snli: Natural language inference with natural language explanations. *Advances in Neural Information Processing Systems*, 31.

Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. 2017. Reading Wikipedia to answer open-domain questions. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1870–1879, Vancouver, Canada. Association for Computational Linguistics.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Y. Zhao, Yanping Huang, Andrew M. Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. 2022. Scaling instruction-finetuned language models. *CoRR*, abs/2210.11416.

Julian Eisenschlos, Bhuwan Dhingra, Jannis Bulian, Benjamin Börschinger, and Jordan Boyd-Graber. 2021. Fool me twice: Entailment from Wikipedia gamification. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 352–365.

Naoya Inoue, Harsh Trivedi, Steven Sinha, Niranjan Balasubramanian, and Kentaro Inui. 2021. Summarize-then-answer: Generating concise explanations for multi-hop reading comprehension. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6064–6080.

Kelvin Jiang, Ronak Pradeep, and Jimmy Lin. 2021. Exploring listwise evidence reasoning with t5 for fact verification. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics*

and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers), pages 402–410.

Yichen Jiang and Mohit Bansal. 2019. Avoiding reasoning shortcuts: Adversarial evaluation, training, and model development for multi-hop qa. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2726–2736.

Yichen Jiang, Shikha Bordia, Zheng Zhong, Charles Dognin, Maneesh Singh, and Mohit Bansal. 2020. Hover: A dataset for many-hop fact extraction and claim verification. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3441–3460.

Shailza Jolly, Pepa Atanasova, and Isabelle Augenstein. 2022. Generating fluent fact checking explanations with unsupervised post-editing. *Inf.*, 13(10):500.

Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, pages 4171–4186.

Neema Kotonya and Francesca Toni. 2020. Explainable automated fact-checking for public health claims. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7740–7754.

Canasai Kruengkrai, Junichi Yamagishi, and Xin Wang. 2021. A multi-level attention model for evidence-based fact checking. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 2447–2460.

David MJ Lazer, Matthew A Baum, Yochai Benkler, Adam J Berinsky, Kelly M Greenhill, Filippo Menczer, Miriam J Metzger, Brendan Nyhan, Gordon Pennycook, David Rothschild, et al. 2018. The science of fake news. *Science*, 359(6380):1094–1096.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Zhenghao Liu, Chenyan Xiong, Maosong Sun, and Zhiyuan Liu. 2020. Fine-grained fact verification with kernel graph attention network. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7342–7351, Online. Association for Computational Linguistics.

Preslav Nakov, Firoj Alam, Yifan Zhang, Animesh Prakash, and Fahim Dalvi. 2022. QCRI's COVID-19 Disinformation Detector: A System to Fight the COVID-19 Infodemic in Social Media. *arXiv:2204.03506 [cs]*.

Wojciech Ostrowski, Arnav Arora, Pepa Atanasova, and Isabelle Augenstein. 2021. Multi-hop fact checking of political claims. In *IJCAI*, pages 3892–3898. ijcai.org.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback.

Liangming Pan, Xiaobao Wu, Xinyuan Lu, Anh Tuan Luu, William Yang Wang, Min-Yen Kan, and Preslav Nakov. 2023. Fact-checking complex claims with program-guided reasoning.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Marco Túlio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. "why should I trust you?": Explaining the predictions of any classifier. In *KDD*, pages 1135–1144. ACM.

Tal Schuster, Darsh J. Shah, Yun Jie Serene Yeo, Daniel Filizzola, Enrico Santus, and Regina Barzilay. 2019. Towards debiasing fact verification models. In *EMNLP*.

Gautam Kishore Shahi and Durgesh Nandini. 2020. Fakecovid- A multilingual cross-domain fact check news dataset for COVID-19. In *ICWSM Workshops*.

Sofia University "St. Kliment Ohridski", Bulgaria, Pepa Gencheva, Preslav Nakov, Qatar Computing Research Institute, HBKU, Qatar, Lluís Màrquez, Qatar Computing Research Institute, HBKU, Qatar, Alberto Barrón-Cedeño, Qatar Computing Research Institute, HBKU, Qatar, Ivan Koychev, and Sofia University "St. Kliment Ohridski", Bulgaria. 2017. A Context-Aware Approach for Detecting Worth-Checking Claims in Political Debates. In *RANLP 2017 - Recent Advances in Natural Language Processing Meet Deep Learning*, pages 267–276. Incoma Ltd. Shoumen, Bulgaria.

Dominik Stammbach and Elliott Ash. 2020. e-fever: Explanations and summaries for automated fact checking. *Proceedings of the 2020 Truth and Trust Online (TTO 2020)*, pages 32–43.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018. FEVER: a large-scale dataset for fact extraction and verification. In *NAACL-HLT*, pages 809–819. Association for Computational Linguistics.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2019. Evaluating adversarial attacks against multiple fact verification systems. In *EMNLP*.

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

William Yang Wang. 2017. "Liar, Liar Pants on Fire": A New Benchmark Dataset for Fake News Detection. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 422–426, Vancouver, Canada. Association for Computational Linguistics.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, pages 24824–24837. Curran Associates, Inc.

Yifan Wei, Yisong Su, Huanhuan Ma, Xiaoyan Yu, Fangyu Lei, Yuanzhe Zhang, Jun Zhao, and Kang Liu. 2023. Menatqa: A new dataset for testing the temporal comprehension and reasoning abilities of large language models.

Wenhan Xiong, Xiang Lorraine Li, Srini Iyer, Jingfei Du, Patrick S. H. Lewis, William Yang Wang, Yashar Mehdad, Scott Yih, Sebastian Riedel, Douwe Kiela, and Barlas Oguz. 2021. Answering complex open-domain questions with multi-hop dense retrieval. In *ICLR*. OpenReview.net.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2369–2380.

Wanjun Zhong, Jingjing Xu, Duyu Tang, Zenan Xu, Nan Duan, Ming Zhou, Jiahai Wang, and Jian Yin. 2020. Reasoning over semantic-level graph for fact checking. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6170–6180.

Jie Zhou, Xu Han, Cheng Yang, Zhiyuan Liu, Lifeng Wang, Changcheng Li, and Maosong Sun. 2019. Gear: Graph-based evidence aggregating and reasoning for fact verification. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 892–901.

A Appendix

A.1 Details of data collection

We employ annotators from Appen and supply them with our detailed annotation guidelines. To initiate the process, we first identify a seed article from the top 50,000 popular Wikipedia pages and then select subsequent articles based on the hyperlink entities on the first page. We iterate this process one more time to get 25 instance pairs, which consist of three Wikipedia entities as the multi-hop reasoning path. Then we randomly sample some pairs from the 25 pairs to avoid position bias. Since the bridge entity might not be reasonable enough, the annotator is able to discard this instance to keep more reasonable pairs, and the skipped data is discarded.

Following this, annotators proceed with the annotation process as previously described. We first spend two weeks training those annotators from our feedback, after the quality is consistent and high enough.

We first ask annotators to read two or three Wikipedia documents that are highlighted to feature a linked entity. From there, they generate a claim that encompasses all the golden documents labeled as SUPPORT and construct REFUTE examples. For the NOT ENOUGH INFO label, we employ two approaches: either randomly removing an article or composing a claim that is unrelated to the current article information. Upon completion of these steps, the submitted data is handed over to quality inspectors for further examination.

In order to harvest a high-quality dataset, we employ additional annotators as quality inspectors to scrutinize the annotations submitted by the primary annotators. If the data meets the inspection criteria, it is accepted; however, if the data is found to be inadequate, it is returned to the annotators along with revision suggestions for re-annotation until it successfully passes inspection. We ask annotators to annotate every 15,000 instances as a milestone. Then we roll a data validation period.

We collect Wikipedia article pairs and then engage annotators from Appen to annotate a total of 60,000 claims. Each claim is assigned a verdict label and an accompanying explanation. In total, we compensate the crowdworkers with a sum of 130,000 CNY. Each instance costs about 2.2 CNY.

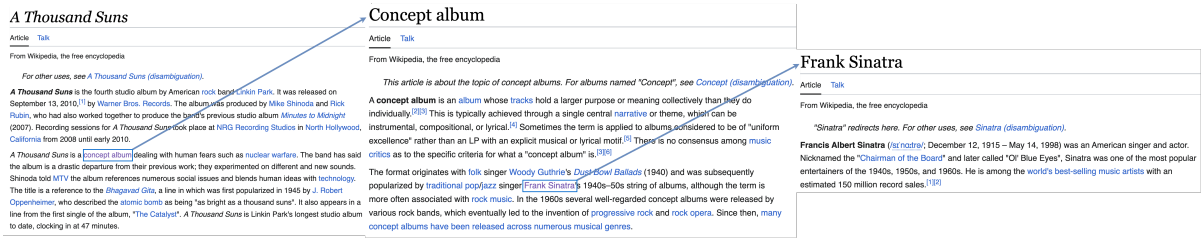

Figure 3: A sample in the proposed dataset EX-FEVER. The corresponding claim is "A Thousand Suns is an album dealing with human fears such as nuclear warfare, where the theme of the album was subsequently popularized by a traditional pop/jazz American singer and actor"

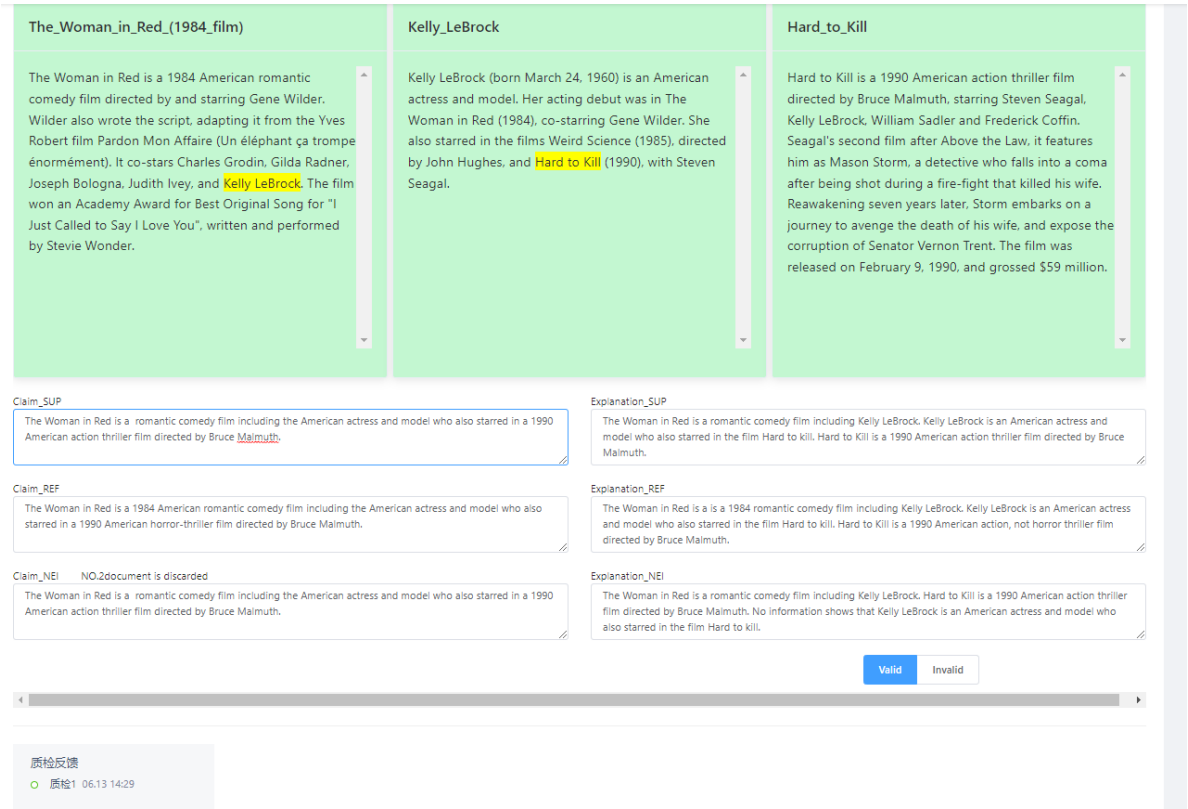

Figure 4: Annotation platform

These crowdworkers are hired from Appen who are fluent in English.

A.2 Details of the baseline system implementation

Document Retrieval Given a specific claim c, we initially utilize the TF-IDF model to generate the top 200 documents, denoted as $D = d_1, \cdots, d_N$, where $N = 200$. Subsequently, both neural-based retrieval models access the resulting document corpus. The two neural-based models are fine-tuned on our dataset's training split.

For the BERT-based mode, during the training phase, we select five documents for each claim. This training set comprises two (three) golden documents and three (two) non-golden documents that possess the highest cosine similarity scores from the TF-IDF model. The BERT-based model is fine-tuned on positive-negative pairs data from the dataset. This model takes a single document $d \in D$ and the claim c as inputs, and outputs a score that reflects the relatedness between d and c. During the testing phase, the model evaluates the relevance score for each claim-document pair and then sorts the documents D according to their relevance score.

The MDR model uses a shared RoBERTa-base (Liu et al., 2019) encoder for both document and query encoders and implements maximum inner product search over dense representations of

the pre-retrieval documents set.

The process of MDR is slightly different. For each claim, we select either the first or second document with the highest ranking from the TF-IDF model as the hard negative sample. Since the retrieval process of MDR operates in an iterative manner, with each query relying on the results from the previous retrieval, we conduct experiments differentiating between 2-hop and 3-hop data. In a single training epoch, we randomly assign MDR to initially train on either the 2-hop or 3-hop data. During the testing phase, we retrieve three documents for each claim. However, when calculating the evaluation metrics, we disregard the third document for the 2-hop portion of the data.

Explanatory Stage We fine-tune the BART model via the Hugging Face Transformers library[2] on our dataset. To enable the model to capture all relevant information during training, all golden documents are included in the model's input. Similar to the Document Retrieval experimental setting and given that the BART model has limited input tokens, the training data comprise five documents for each record, two (three) golden documents and three (two) non-golden documents that possess the high-rank order from the retrieval model. In contrast, during the test phase, we select the top five documents yielded by the retrieval models without additional consideration for including all golden documents and we select the rouge score as the evaluation metric.

Verdict Predict We designate the claim to be verified as the MNLI task hypothesis and the relevant explanation of the verdict as the MNLI task premise. The BERT model is fine-tuned by using the Hugging Face Transformers library[2]. We implement the GEAR model (Zhou et al., 2019), a graph-based text reasoning model, as a comparative alternative. The GEAR model conceptualizes sentences functioning either as evidence or claims within a graph theoretic framework, regarding such sentences as nodes on a graph. An evidential reasoning network and evidential aggregator transmit evidential information and make predictions. By combining these elements, GEAR leverages both evidence and claim in BERT to obtain an evidence representation e_i. The claim is fed into BERT alone to obtain representation c. Translating evidential and claim sentences into graph nodes expressing their relations, GEAR provides an integrative approach to adjudicating claim veracity via reasoning across the evidence.

[2] https://github.com/huggingface/transformers/tree/main/examples/pytorch/text-classification

A.3 Details of GPT-3.5-turbo Prompt

Claim only

Check the claim: [claim]
Choices:['SUPPORT', 'REFUTE','NOT ENOUGH INFO']
Answer:

W/o explanation required

Claim: [claim]
Evidence: [golden documents]
Evaluate the claim based on the provided evidence and
choose one of the following labels: 'SUPPORT', 'REFUTE', or 'NOT ENOUGH INFO'.

W/ explanation required

Claim: [claim]
Evidence: [evidence]
Evaluate the claim based on the provided evidence and
choose one of the following labels: 'SUPPORT', 'REFUTE', or 'NOT ENOUGH INFO'.
Provide a brief explanation for your choice.

Few-shot prompt

I will provide you with evidence and a claim. Your task is to determine if the claim is supported, refuted, or if there is not enough information based on the given evidence. You need to choose one of the following labels: 'SUPPORT', 'REFUTE', or 'NOT ENOUGH INFO'. After choosing a label, please provide a brief explanation for your choice.
Example 1:
Evidence: [evidence]
Claim: [claim]
Choices: ['SUPPORT', 'REFUTE', 'NOT ENOUGH INFO']
Answer: [answer]
Explanation: [explanation]
...
Example N:
Evidence: [evidence]
Claim: [claim]
Choices: ['SUPPORT', 'REFUTE', 'NOT ENOUGH INFO']
Answer: [answer]
Explanation: [explanation]
Now, please evaluate the following claim based on the provided evidence:

Evidence: [evidence]
Check the claim: + [claim] + from the above evidence
Choices: ['SUPPORT', 'REFUTE', 'NOT ENOUGH INFO']
Answer:

where N = 1, or 3.