PolitiReceipts: Structuring Fact-Checks for Automated Fact-Checking of Real-World Claims

Anonymous ACL submission

Abstract

Fact-checking is crucial for combating misinformation by investigating claims, reviewing evidence, and determining veracity. Although fact-check articles typically include standardized components like the claim, context, and final verdict (as outlined in the ClaimReview schema), they often lack detailed documentation of supporting evidence and justification due to the extra workload required for comprehensive annotation. This limitation not only reduces the reusability of fact-checks but also highlights the need for more scalable methods. To address this, we introduce PolitiReceipts, a novel resource constructed from over 17,000 PolitiFact fact-checking articles published between 2007 and January 2025. By leveraging a Large Language Model (LLM) with few-shot inference, our approach jointly extracts evidence spans, their decontextualized variations, and evidence-grounded justifications. A small human study confirms overwhelming agreement on all evaluated aspects of the extractions. Furthermore, our benchmarks for Automated Fact-Checking (AFC) with LLMs demonstrate that decontextualization significantly improves downstream task performance and that larger models consistently yield better results in scenarios with optimal evidence as well as in end-to-end settings. Our findings highlight the potential of PolitiReceipts to serve as a robust foundation for future research in explainable and scalable automated fact-checking.

1 Introduction

In a connected and an increasingly polarized world, fact-checking remains a necessary tool against the continuous spread of misinformation. Fact-checking is a form of journalism that investigates check-worthy claims (Panchendrarajan and Zubiaga, 2024), examines evidence, and draws logical conclusions to arrive at a final verdict ruling towards the veracity of the claim (Graves, 2016; Jiang et al., 2020). The products of this resource-intensive process are fact-checking articles, also referred to as fact-checks. Fact checks generally specify common components such as claim, context information, and the final verdict, which have been made standardized properties within the ClaimReview[1] schema. While many fact-checking organizations have integrated the annotation of these properties, more detailed documentation of supporting evidence and concluding justification is omitted due to the additional workload that comprehensive annotation would impose on experts (Jiang et al., 2020). This limitation not only reduces the reusability of high-quality fact checks, but also highlights the need for more scalable methods.

Figure 1: The left panel shows the claim and the reduced article, while the right panels present one of the extracted evidences and the decontextualized variation (upper) and generated justification (lower). The original article does not have an explicit conclusion paragraph.

Automatic fact-checking (AFC) approaches provide a computational perspective towards verifying real-world claims. However, these approaches often struggle with handling inconsistent evidence and benchmarking the performance and quality of

[1] https://schema.org/ClaimReview

generated justifications due to a lack of references (Guo et al., 2022; Sahitaj et al., 2025). By automatically extracting structured information from fact-checking articles, we aim to bridge the gap between traditional, manually curated resources and automated systems. This idea of leveraging existing human-written fact-checking efforts has also been explored in claim matching, where a list of associated fact-checking articles is retrieved from a knowledge base (KB) for a presented novel claim (Panchendrarajan and Zubiaga, 2024). Extracting detailed components from fact-checking articles can enable better matching and more fine-grained investigations.

Existing methods extract information from fact-checks using fixed-length chunks or punctuation-based cues, which breaks up continuous sequences that rely on contextual usage of references, resulting in lost information (Khan et al., 2022; Zeng and Gao, 2024a). More sophisticated approaches from adjacent areas of application implement decontextualization to formulate extracted atomic facts into self-contained statements (Gunjal and Durrett, 2024; Deng et al., 2024). We transfer this idea of extracting decontextualized facts for the verification of generative content to the extraction of precise evidence spans and their decontextualized variations from fact-checks. Similarly, heuristic cues used for extracting justifications do not generalize well across candidate articles and introduce a selection bias by excluding articles that lack predefined markers (Zeng and Gao, 2024a). We extract justifications, even when a conclusion paragraph does not exist, to document the reasoning in terms of the extracted evidence. Figure 1 illustrates an example of the joint extraction of one of the exact evidence spans, its decontextualized variation, and the extracted justification.

Our contributions are threefold. First, we assemble a comprehensive dataset of over 17,000 fact-checking articles published between 2007 and January 2025, organized within a detailed ontology and enriched with robust metadata. Second, we propose an automated extraction approach that leverages few-shot inference with LLMs to jointly extract exact and decontextualized evidence, justifications, and final verdicts. We name the enriched dataset—comprising our extracted, decontextualized evidence pieces and the justifications—'PolitiReceipts.'. Third, we benchmark performance on PolitiReceipts with four state-of-the-art LLMs for AFC task, empirically evaluating their ability to generate useful justifications and accurate verdicts. To capture realistic performance, we evaluate models under optimistic scenarios, assuming optimal evidence with no false positives or negatives, in an end-to-end retrieval setting, and under more challenging conditions with no evidence at all to indirectly measure the impact of models' parametric knowledge.

2 Related Work

To facilitate AFC, a number of fact-checking datasets were published; however, many of them contain false claims that are synthetically generated by modifying real claims (by meaning-altering), with the main source of evidence being Wikipedia (Thorne et al., 2018; Schuster et al., 2021; Aly et al., 2021; Ma et al., 2024). These datasets address challenges that require natural entailment (comparing if a premise entails a hypothesis), in which the sentences are usually short. However, real-world claims are usually more complex than claims from computational fact-checking datasets. Verifying these claims requires having strong context and evidence that are not obvious or easily retrieved from the web. To address this, some works have focused on collecting claims from real-world fact-checking articles, such as LIAR-PLUS (Alhindi et al., 2018), MultiFC (Augenstein et al., 2019), PubHealth (Kotonya and Toni, 2020), FactEx (Althabiti et al., 2023). In these datasets, the entire fact-checking article were treated as evidence, and some also extracted justifications that appears in the end of the articles. For example, Alhindi et al. (2018) extract justifications based on textual cues (e.g.: "Our ruling") and used them directly as evidence for AFC. Similarly, Zeng and Gao (2024b) extracted justifications from the fact-checking articles based on textual cues as paragraph-level extraction, but used them as explanations for the extracted reference evidences.

When evidence pieces are directly taken from fact-checking articles, they usually are at risk of leaking the final verdict label assigned to the claim (Glockner et al., 2022). A few works have explored the utilization of evidences from references presented in the fact-checks (Khan et al., 2022; Zeng and Gao, 2024b). However, by extracting chunked evidences from external resources and excluding the framing argument which is presented in the fact-checking article, we necessarily lose contextual information between external refer-

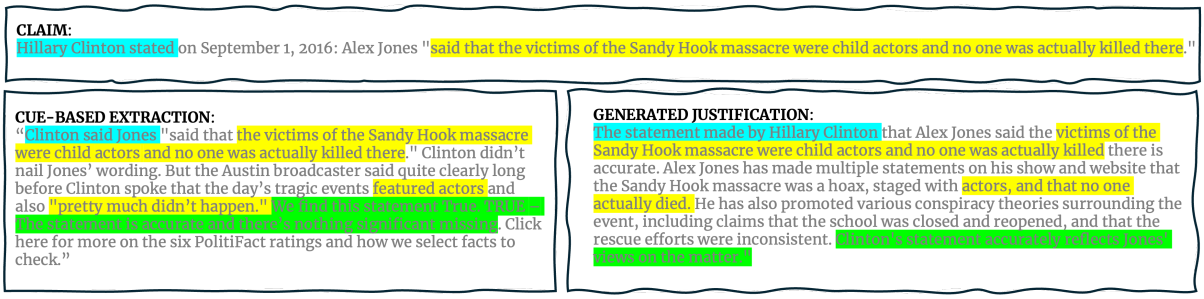

Figure 2: The rule-based extraction based on 'Our ruling' and 'Our rating' usually yield unrefined paragraphs with auxiliary information, whereas our approach generates a concise, targeted explanation that is grounded in the extracted evidence.

ences and presented verdict that may be necessary to differentiate different aspects of the evidence.

One promising direction is to extract structured information from the fact-checking articles. Existing approaches—such as BERT-based token-level sequence classification (Jiang et al., 2020)—often rely on paragraph-level chunking to manage articles that exceed BERT's 512-token limit. While effective for extracting isolated elements like claims or verdicts, these methods struggle when evidence spans multiple sentences or even paragraphs, as chunking fails to capture broader contextual dependencies. In contrast, Dagdelen et al. (2024) explore structured information extraction in scientific texts by fine-tuning a quantized Llama2 70B model using parameter-efficient methods to jointly extract named entities and their relations from segments of 512 to 1024 tokens. Since our focus is on document-wide joint extraction of evidences and dependent justifications, fine-tuning with such restricted context sizes is not a viable option.

The most closely related work is PolitiHop (Ostrowski et al., 2021), in which the evidence were manually annotated by selecting sentences from PolitiFact fact-checking articles. However, as human annotation takes time, the dataset only contains 500 claims. In stead of human annotation, our work uses an LLM for automatic extraction, which largely scales up dataset size, with our dataset containing 17,276 articles from year 2007 to 2025.

3 PolitiReceipts Corpus Construction

3.1 Data Collection

For the purpose of creating a useful resource for the area of AFC, we construct a structured dataset of fact-checks from PolitiFact, a non-profit fact-checking organization that rigorously verifies the accuracy of political statements. PolitiFact adheres to a strict review process and emphasizes editorial independence, with journalists selecting claims to investigate based on relevance and verifiability. PolitiFact uses a systematic methodology that incorporates reliable sources and expert interviews to ensure the credibility of its fact-checks. Thus, PolitiFact fact-checks are generally considered a preferred source for check-worthy claims and accompanying fact-checking articles.

We systematically collect 17,276 fact-checks from politifact.com between 2007 and January 2025. The collected fact-checks are structured into claim reviews, with additional metadata such as speaker information and sources, following a transparent data collection pipeline. Further details on data acquisition, ontology, and distribution are attached in the Appendix A.1. The data will be made available for the non-profit research community on request.

3.2 Extraction Methodology

Our extraction methodology leverages few-shot inference with LLMs to jointly extract decontextualized evidences, justifications, and final verdicts from unstructured fact-checking articles. This approach addresses the inherent challenges of processing long-form, context-dependent texts and ensures that the core reasoning components of the fact-checking process are effectively captured. To guide the LLM in navigating the diverse landscape of fact-checking articles written by 661 authors in the presented PolitiReceipts corpus, we curate a set of nine hand-annotated examples. The few-shot examples are carefully selected to cover a wide range of extraction patterns across all six original labels. Specifically, we focus on extracting exact spans of evidences from single-sentence to connected multi-sentence examples to remain flexible and overcome the limitations of atomic fact extraction. Addition-

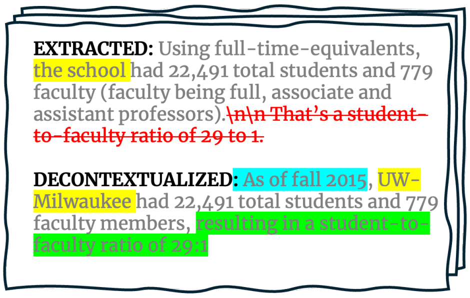

Figure 3: Extracted and decontextualized evidence.

ally, we decontextualize the extracted exact evidence spans to ensure self-contained statements that can be used independently of the surrounding context (Gunjal and Durrett, 2024; Zeng and Gao, 2024a). Figure 3 illustrates the advantages to the joint extraction and decontextualization of evidences. Not only, are references to the date and institution correctly resolved, but also the relevant information from a subsequent paragraph and evidence is correctly integrated.

Zeng and Gao (2024c) apply cue-based extraction of paragraphs justification based on "Our Ruling" and "Our Rating". This is specific to PolitiFact and not available in every case. For the collected previously 17,276 articles, we only identify 10,648 with this cue. Thus, this approach is not viable. Figure 2 illustrates the difference in quality between cue-based extracted and our generated justification from PolitiReceipts. We formulate the justification extraction based on (1) the information that can be often found towards the end of an article, if available, and (2) explicitly resolve references towards the utilized evidences to enable the generation of evidence-grounded justification of the final verdict ruling.

The extraction task is formulated as a structured joint information extraction problem, where the model is required to simultaneously generate:

- **Exact Evidence Spans:** Continuous text segments from the article that collectively represent unified, contextually anchored arguments pertinent to the claim.
- **Decontextualized Evidences:** Refined versions of the extracted evidences that are paraphrased to be self-contained, resolving contexual references from the original text.
- **Justification:** The reasoning linking the evidence to the final verdict, capturing the conclusions drawn by the fact-checking expert.
- **Verdict:** The final fact-checking label, included both as a component of the extraction output and as a consistency check against the annotated metadata.

The prompt with the detailed instruction is available in Appendix 8. By performing joint extraction, the approach aims to minimize error propagation and prevent information leakage, issues that have been noted in earlier systems relying on cue-based or isolated extraction methods (Jiang et al., 2020; Dagdelen et al., 2024).

A key aspect of our methodology is the transformation of raw evidences into self-contained factual statements. Adhering to the principles of decontextualization and minimality (Gunjal and Durrett, 2024), we ensure that the extracted evidence pieces are both concise and comprehensive, containing only the minimal necessary context to stand alone. Preliminary experiments indicated that aggregating non-connected sequences did not improve downstream performance, thereby reinforcing our focus on continuous, decontextualized text spans.

We implement the above extraction approach as a structured generation task using an LLM configured with a necessary 32k token context length, using the `vLLM`[2] and `outlines`[3] framework. We use 3.3 70B Llama as it achieves comparable performance to the 3.1 405B model, making it one of the state-of-the-art open source LLMs at the size of 70B. The prompt design integrates the hand-annotated few-shot examples with the full article text, allowing the model to draw upon extended context while adhering to the extraction schema. This integration not only facilitates the extraction of coherent evidences and justifications but also enables the simultaneous prediction of the verdict—thus providing a built-in mechanism for consistency verification.

We assessed our extraction pipeline by first comparing the generated verdicts with the collected metadata. For 81 articles, the extracted labels did not match the collected labels. A closer look revealed that these articles were dated back to the founding year and were relatively short, often lacking a clear verdict. In these cases, neither our expert annotators nor the model could accurately determine the correct verdict given the article, so we excluded them. For 36 articles with claims labeled as “panths-on-fire”, the model refused to

[2] https://github.com/vllm-project/vllm
[3] https://github.com/dottxt-ai/outlines

extract evidences with given the instructions, but correctly extracted the justification. We exclude these edge cases from the corpus. In the remaining cases with a volume of less than 1%, parsing issues with correctly extracted content were resolved by re-running the extraction. The extraction process left us with 17,135 articles from the original 17,276 that were collected. The final corpus has a total count of 106,036 evidences, with a median count of 6, and a standard deviation of 3.07.

4 Corpus Evaluation

In this section, we evaluate the quality and utility of our extracted evidences and justifications. To verify that our evidences are correctly extracted in the source articles, we compute the Levenshtein distance between the extracted text spans and candidate sequences from the fact-checking article using fuzzy sub-sequence search. Using a fixed limit of five edit operations, we find a 94,85% success rate in identifying the sequences in the article. We attribute the remaining cases largely due to formatting issues. As a majority of sequences can be matched without issues, we are confident in continuing with the evaluation of the quality of the extracted components.

4.1 Human Evaluation

While automated metrics offer useful insights, human evaluation is useful towards assessing the utility of our extracted components. Our human evaluation focuses primarily on precision rather than recall, we want to ensure that the extracted evidence spans and justifications are correct and grounded in the original articles, rather than exhaustively capturing all possible evidence that may be found in the article. For this study, a stratified random sample of 50 cases (selected by year and label) was chosen from the dataset, and each case was independently evaluated by three expert annotators to ensure reliable measurements.

- **Evidence Label Leakage Q_0:** Do the extracted text spans or their decontextualized versions leak the label of the article?
- **Evidence Meaning Preservation Q_1:** Is the decontextualized evidence faithful towards the extracted evidence (meaning is retained)?
- **Justification Consistency Q_2:** Does the extracted justification follow the reasoning presented in the article?
- **Justification Coverage Q_3:** Is the justification fully grounded in the extracted evidences? (1-5 Likert scale)
- **Justification Utility Q_4:** Is the justification sufficient to justify the assigned verdict?

Q_0 is simply answered as a binary question, while aspects Q_1 to Q_4 are rated on a $1-5$ Likert scale.

We evaluate each of the above aspects Q_1 to Q_4 using Themis (Hu et al., 2024), a reference-free metric that enables the assessment of flexible evaluation criteria. We compare the automated scores from Themis against our human evaluation results for each aspect, except for label leakage, to validate our extraction quality and the overall utility of the extracted components.

4.2 Results

No label leakage was detected in our stratified sample. However, inter-annotator agreement for the remaining aspects, as summarized in Table 1, yielded low reliability scores. Krippendorff's alpha is computed based on the annotations of the three experts. Cohen's Kappa is cpmputed based on the majority vote of the expert annotators against automated Themis evaluations. Although the calculated Krippendorff's alpha and Cohen's Kappa values appear low, this may be misleading due to the highly skewed distribution of ratings. Because Krippendorff's alpha measures the observed disagreement relative to the expected disagreement in a skewed distribution, it becomes difficult to interpret the actual reliability of the annotations.

Metric	Q_1	Q_2	Q_3	Q_4
Krippendorff's Alpha	0.215	0.059	-0.050	0.029
Cohen's Kappa	0.106	0.030	0.024	0.004

Table 1: Inter-Annotator Agreement Metrics.

To gain a deeper understanding of the quality ratings, we further examined consensus levels. Table 2 presents the counts for which all three annotators and at least two annotators assigned a rating of 5. These results indicate that a substantial proportion of evaluations received the highest rating.

Each annotator scored the aspects with a mean of above 4.5, a mode of 5 and a standard deviation below 1.0 below. The overall quality ratings are on average very high and clustered at the top

Question	3/3 Concesus 5	2/3 Concesus 5
Q_1	241/322	299/322
Q_2	42/50	49/50
Q_3	35/50	49/50
Q_4	28/50	47/50

Table 2: Consensus ratings of 5 for each question: the second column shows the counts where all three annotators agreed (3 Consensus 5), and the third column shows the counts where at least two annotators agreed (2 Consensus 5).

of the scale, underscoring the effectiveness of our extraction process.

5 Benchmarking AFC on PolitiReceipts

5.1 Task Description

Using PolitiReceipts, we are interested in studying the utility of state-of-the-art LLMs on AFC for realistic claims. The central objective is twofold: given a check-worthy real-world claim, our goal is to (i) correctly predict its veracity and (ii) generate a human-aligned justification that explains the reasoning behind the verdict. Both verdict prediction and justification generation require a set of relevant evidences that, together, provide a sufficient basis for verifying the claim.

5.2 Evidence Retrieval

To simulate a full end-to-end scenario, we incorporate an evidence retrieval step. Each model is evaluated on both the raw extracted evidences and their decontextualized counterparts to investigate the impact of the decontextualization. Our experimental design aims to establish upper and lower performance bounds by examining:

- **Optimal Evidence:** The ideal scenario in which the model receives all the relevant evidence as directly extracted from the article.
- **No Evidence:** A lower-bound scenario where the model must rely exclusively on its parametric knowledge.
- **Retrieved Evidence:** A realistic end-to-end condition in which evidence is automatically retrieved from the KB of extracted evidences

For the retrieval, we utilize the claim as query. Both, claims and evidences are embedded using `bge-small-en-v1.5`[4] All embeddings are normalized prior to indexing. We use `Qdrant` as the vector database, with cosine similarity as the retrieval metric. Our search algorithm is HNSW, and we limit the maximum number of retrieved evidences to 10.

5.3 Verdict Prediction

To mirror a realistic fact-checking scenario, we reformulate the original six verdict labels into two distinct schemes: a five-class setup and a binary (two-class) setup. In the five-class scheme, the "pants-on-fire" label is merged into the broader "false" category, while preserving the remaining classes. In the binary scheme, labels are aggregated into two broad categories (mostly-false and mostly-true), thus simplifying the task.

5.4 Justification Generation

For each case, the model is required to produce a justification that connects the provided evidence to a final verdict prediction. To ensure uniformity in output, we require that the generated justification meet the following criteria:

- **Clarity:** The explanation should be concise, coherent, and complete.
- **Relevance:** The explanation must directly relate to the claim and the evidences provided.
- **Consistency:** The justification should be consistent with the presented information.
- **Utility:** The explanation should help users evaluate the claim's veracity.

6 Experiments

6.1 Setup

We implement the AFC task as a structured generation approach using the `vLLM` and `outlines` where justification and verdict are required properties of an output structure. We evaluate several LLM architectures ranging between 7B to 72B parameters. For each model, we establish the performance for the ideal case, the no evidence scenario, and the end-to-end case with retrieved evidence. The performance is evaluated on both the original extracted text spans (E) and their decontextualized versions (D).

[4]https://huggingface.co/BAAI/bge-base-en-v1.5

Method	nDCG@3	nDCG@5	nDCG@10	F1
E	0.71	0.68	0.67	0.57
E (w. Context)	0.89	0.86	0.85	0.76
D	0.75	0.72	0.71	0.61
D (w. Context)	0.90	0.88	0.87	0.78

Table 3: Retrieval Evaluation

6.2 Evaluation Metrics

Our evaluation framework integrates the following reference-based and reference-free measures:

BLEURT BLEURT (BT) evaluates as a regression task, based on BERT and fine-tuned to predict human judgment ratings from source to reference. Assigns values from 0 to 1 (Sellam et al., 2020).

BARTScore BARTScore (BS) evaluates as a text generation task, based on BART (Lewis et al., 2019) and computes log-likelihood of source being generated, given reference (Yuan et al., 2021).

TIGERScore TIGERScore (TS) is a reference-free metric, based on Llama2 and fine-tuned on human evaluation reports to generate error penalty scores for the generated source in the range of $[-5, -0.5]$ per error and report a cumulative score (Jiang et al., 2024).

7 Benchmarking AFC Results

7.1 Evidence Retrieval

We evaluated our evidence retrieval performance using two key metrics: normalized Discounted Cumulative Gain (nDCG) at ranks 3, 5, and 10, as well as the F1 score. In our analysis, we compared retrieval performance when using the claim without any additional contextual information versus using the claim with full context. This comparison helps us assess the impact of contextual cues on retrieval quality.

Table 3 demonstrates that decontextualized evidence (D) consistently yields better retrieval performance compared to the extracted evidence (E). Moreover, we observe that retrieval based on the claim and its context, also ensures higher retrieval performance. As k increases, more evidences with no or lower relevance are included, which reduces the overall nDCG scores in all settings.

7.2 Verdict Prediction

The results of our benchmarking experiments for the Qwen2.5 7B and 72B, and Llama3 8B and 70B models are documented in Tables 4 and Table 5 for the five- and two-class label schemes, respectively. Additional results for the DeepSeek-R1-Distill-Llama 8B and 70B models are provided in Tables 8 and 7 in the Appendix. Larger models generally achieve better scores than smaller models across all metrics, highlighting the impact of model capacity. Overall, models perform best when provided with optimal evidence, with generally higher scores across metrics compared to the no-evidence condition. Decontextualized (D) evidence tends to yield slightly better performance to the extracted (E) evidence across models and metrics. The gap between optimal evidence and retrieved evidence is noticeable, indicating that during retrieval, errors were introduced as expected. Thus reducing the verdict prediction performance as well as the justification quality.

7.3 Justification Generation

Across all models, larger models consistently achieved better scores. Similarly, justifications generated with decontextualized evidence (D) outperformed those based on the raw extracted evidence (E) in all metrics. BLEURT showed higher scores for larger models and, specifically, when using optimal decontextualized evidences, indicating better alignment with human evaluations. BARTScore measurements suggest that the justifications were more likely to be generated from optimal decontextualized evidences. TIGERScore also reflected these trends, as it penalized fewer errors in the justifications produced by larger models with decontextualized evidence. The retrieval component reduced performance in each setting, indicating that providing evidence that is not relevant to the problem, reduces the quality of the justification. These findings emphasize that both model capacity and the type of evidence, significantly impact the quality of generated justifications in end-to-end AFC.

8 Discussion

We demonstrate the utility of extracted evidence and justifications for evaluating the utility of AFC with LLMs. We collected over 17,000 PolitiFact fact-checks as PolitiReceipts, providing a scalable resource with extracted evidence, decontextualized variations, and justifications. This resource addresses the challenge of the manual, resource-intensive documentation of fact-checks. Our hu-

Model	Five Class BT	BS	F1	TS
Qwen2.5-7B				
Optimal Evidence (D)	0.30	-2.36	42.70	-1.52
Retrieved Evidence (D)	0.26	-2.48	40.42	-1.63
Optimal Evidence (E)	0.27	-2.44	41.84	-1.54
Retrieved Evidence (E)	0.23	-2.54	39.05	-1.74
No Evidence	0.09	-2.52	32.72	-3.20
Qwen2.5-72B				
Optimal Evidence (D)	0.33	-2.20	51.38	-0.91
Retrieved Evidence (D)	0.30	-2.28	48.53	-0.96
Optimal Evidence (E)	0.31	-2.24	51.08	-0.95
Retrieved Evidence (E)	0.27	-2.32	48.59	-1.02
No Evidence	0.13	-2.34	38.08	-2.75
Llama-3.1-8B				
Optimal Evidence (D)	0.25	-2.35	41.52	-1.91
Retrieved Evidence (D)	0.22	-2.47	38.41	-2.16
Optimal Evidence (E)	0.22	-2.41	40.41	-2.01
Retrieved Evidence (E)	0.20	-2.52	37.87	-2.33
No Evidence	0.02	-2.82	28.78	-3.43
Llama-3.3-70B				
Optimal Evidence (D)	0.33	-2.12	52.66	-1.03
Retrieved Evidence (D)	0.29	-2.20	49.23	-1.13
Optimal Evidence (E)	0.31	-2.16	52.73	-1.06
Retrieved Evidence (E)	0.27	-2.24	49.34	-1.16
No Evidence	0.12	-2.42	39.93	-2.36

Table 4: Results of the AFC task with 5 labels, reporting BLEURT (BT), BARTScore (BS), F_1 score, and TIGERScore (TS) (§6.2).

Model	Two Class BT	BS	F1	TS
Qwen2.5-7B				
Optimal Evidence (D)	0.30	-2.36	76.21	-1.32
Retrieved Evidence (D)	0.26	-2.48	74.42	-1.46
Optimal Evidence (E)	0.27	-2.44	75.27	-1.38
Retrieved Evidence (E)	0.23	-2.54	72.84	-1.60
No Evidence	0.10	-2.52	66.77	-2.75
Qwen2.5-72B				
Optimal Evidence (D)	0.33	-2.19	82.48	-0.93
Retrieved Evidence (D)	0.33	-2.32	80.94	-1.00
Optimal Evidence (E)	0.31	-2.24	82.70	-0.95
Retrieved Evidence (E)	0.28	-2.32	80.42	-1.02
No Evidence	0.14	-2.35	71.74	-2.29
Llama-3.1-8B				
Optimal Evidence (D)	0.25	-2.35	73.09	-1.93
Retrieved Evidence (D)	0.22	-2.46	71.73	-2.16
Optimal Evidence (E)	0.22	-2.41	72.43	-2.04
Retrieved Evidence (E)	0.20	-2.51	70.70	-2.25
No Evidence	0.02	-2.83	66.14	-3.51
Llama-3.3-70B				
Optimal Evidence (D)	0.33	-2.12	82.04	-1.02
Retrieved Evidence (D)	0.30	-2.19	80.68	-1.12
Optimal Evidence (E)	0.31	-2.16	81.82	-1.06
Retrieved Evidence (E)	0.27	-2.23	80.13	-1.16
No Evidence	0.12	-2.42	72.87	-2.13

Table 5: Results of the AFC task with 2 labels.

man evaluation study further shows high agreement on the quality of these extractions, reinforcing the utility of our approach. In our AFC benchmarks, decontextualized evidence, consistently outperformed raw evidence extractions. Larger models yielded better performance in verdict prediction and justification generation, across all metrics. We observed a drop in performance, when using automatically retrieved evidence, which highlights challenges in the discarding information that is not useful. Overall, our findings support the value of the implemented extraction approach and suggest that refining evidence into self-contained, context-independent units improves end-to-end AFC systems.

9 Future Work

Future work should explore several avenues to further enhance our AFC system. First, expanding the dataset to include fact-checks from multiple sources could help validate the generality of our approach. Additionally, future studies should incorporate larger, more diverse human evaluation samples to provide a more comprehensive assessment of both precision and recall in evidence extraction. Investigating alternative task formulations—such as predict-then-explain or analyze-predict-explain could also yield insights into the optimal structuring of the AFC task. Moreover, improving the retrieval component to better handle noise and different types of evidence is an interesting direction. Finally, addressing challenges related to the extraction from fact-checks with complex layouts (e.g., tables, verbatim interviews) and integrating an initial entity and abbreviation resolution step could further improve the robustness of evidence extraction and subsequent justification generation.

Limitations

Despite promising results, our study has several limitations. First, our human evaluation is based on a relatively small, stratified sample, which may not fully capture the variability in extraction quality across the entire dataset. Although automated metrics and consensus analyses suggest high-quality ratings, the low reliability scores (as detailed in Table 1 and Table 2) indicate that we have not selected the best metric for evaluating agreement in our setting. Second, our experiments rely solely on the PolitiReceipts dataset, which, while extensive, represents only one source of fact-checking content. This slightly limits the generalizability of our findings. Third, the extraction process can be sensitive to document formatting; for instance, articles containing table-style layouts or conversational formats can challenge the extraction pipeline, leading to unexpected extractions. Additionally, issues with structured generation (such as JSON formatting errors when handling quotation marks) and difficulties in verifying complex claims—where underlying messages are not directly reflected in the text further emphasize areas for improvement. Addressing these limitations will be useful for future iterations of the proposed methodology.

Ethical Considerations

Our research primarily relies on data from PolitiFact.com, which may inherently carry biases in its data and fact-checking annotations, as well as in the factual judgements. These biases are out of our control and might influence the predictions of our models. Such biases could potentially intensify if the models are applied on a large scale. We strongly advise caution when considering the implementation of our methods in real-world applications. In addition, while our data could benefit the general public and using LLMs for rationale generation could significantly expedite the automated fact-checking process, there is a risk of misuse by harmful entities. We encourage researchers to exercise caution. Our study utilizes datasets solely in the English language, and it is unclear whether our approach would be equally effective with datasets in other languages.

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.

Saud Althabiti, Mohammad Ammar Alsalka, and Eric Atwell. 2023. Generative ai for explainable automated fact checking on the factex: A new benchmark dataset. In *Multidisciplinary International Symposium on Disinformation in Open Online Media*, pages 1–13. Springer.

Rami Aly, Zhijiang Guo, Michael Sejr Schlichtkrull, James Thorne, Andreas Vlachos, Christos Christodoulopoulos, Oana Cocarascu, and Arpit Mittal. 2021. The fact extraction and VERification over unstructured and structured information (FEVEROUS) shared task. In *Proceedings of the Fourth Workshop on Fact Extraction and VERification (FEVER)*, pages 1–13, Dominican Republic. Association for Computational Linguistics.

Isabelle Augenstein, Christina Lioma, Dongsheng Wang, Lucas Chaves Lima, Casper Hansen, Christian Hansen, and Jakob Grue Simonsen. 2019. MultiFC: A Real-World Multi-Domain Dataset for Evidence-Based Fact Checking of Claims. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4685–4697.

John Dagdelen, Alexander Dunn, Sanghoon Lee, Nicholas Walker, Andrew S. Rosen, Gerbrand Ceder, Kristin A. Persson, and Anubhav Jain. 2024. Structured information extraction from scientific text with large language models. *Nature Communications*, 15(1):1418.

Zhenyun Deng, Michael Schlichtkrull, and Andreas Vlachos. 2024. Document-level Claim Extraction and Decontextualisation for Fact-Checking. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11943–11954, Bangkok, Thailand. Association for Computational Linguistics.

Max Glockner, Yufang Hou, and Iryna Gurevych. 2022. Missing counter-evidence renders NLP fact-checking unrealistic for misinformation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5916–5936, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Lucas Graves. 2016. *Deciding What's True: The Rise of Political Fact-Checking in American Journalism*. Columbia University Press, New York.

Anisha Gunjal and Greg Durrett. 2024. Molecular Facts: Desiderata for Decontextualization in LLM Fact Verification. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 3751–3768, Miami, Florida, USA. Association for Computational Linguistics.

Zhijiang Guo, Michael Schlichtkrull, and Andreas Vlachos. 2022. A Survey on Automated Fact-Checking. *Transactions of the Association for Computational Linguistics*, 10:178–206.

Xinyu Hu, Li Lin, Mingqi Gao, Xunjian Yin, and Xiaojun Wan. 2024. Themis: A Reference-free NLG Evaluation Language Model with Flexibility and Interpretability. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 15924–15951, Miami, Florida, USA. Association for Computational Linguistics.

Dongfu Jiang, Yishan Li, Ge Zhang, Wenhao Huang, Bill Yuchen Lin, and Wenhu Chen. 2024. TIGERScore: Towards Building Explainable Metric for All Text Generation Tasks. *Preprint*, arXiv:2310.00752.

Shan Jiang, Simon Baumgartner, Abe Ittycheriah, and Cong Yu. 2020. Factoring Fact-Checks: Structured Information Extraction from Fact-Checking Articles. In *Proceedings of The Web Conference 2020*, WWW '20, pages 1592–1603, New York, NY, USA. Association for Computing Machinery.

Kashif Khan, Ruizhe Wang, and Pascal Poupart. 2022. WatClaimCheck: A new Dataset for Claim Entailment and Inference. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1293–1304, Dublin, Ireland. Association for Computational Linguistics.

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, Online. Association for Computational Linguistics.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. *Preprint*, arXiv:1910.13461.

Huanhuan Ma, Weizhi Xu, Yifan Wei, Liuji Chen, Liang Wang, Qiang Liu, Shu Wu, and Liang Wang. 2024. EX-FEVER: A dataset for multi-hop explainable fact verification. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 9340–9353, Bangkok, Thailand. Association for Computational Linguistics.

Wojciech Ostrowski, Arnav Arora, Pepa Atanasova, and Isabelle Augenstein. 2021. Multi-hop fact checking of political claims. In *International Joint Conferences on Artificial Intelligence (IJCAI)*.

Rrubaa Panchendrarajan and Arkaitz Zubiaga. 2024. Claim Detection for Automated Fact-checking: A Survey on Monolingual, Multilingual and Cross-Lingual Research. *Natural Language Processing Journal*, 7:100066.

Premtim Sahitaj, Iffat Maab, Junichi Yamagishi, Jawan Kolanowski, Sebastian Möller, and Vera Schmitt. 2025. Towards Automated Fact-Checking of Real-World Claims: Exploring Task Formulation and Assessment with LLMs. *Preprint*, arXiv:2502.08909.

Tal Schuster, Adam Fisch, and Regina Barzilay. 2021. Get your vitamin C! robust fact verification with contrastive evidence. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 624–643, Online. Association for Computational Linguistics.

Thibault Sellam, Dipanjan Das, and Ankur P. Parikh. 2020. BLEURT: Learning Robust Metrics for Text Generation. *Preprint*, arXiv:2004.04696.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018. FEVER: a large-scale dataset for fact extraction and VERification. In *NAACL-HLT*.

Albert Weichselbraun. Inscriptis - a python-based HTML to text conversion library optimized for knowledge extraction from the web. 6(66):3557.

Weizhe Yuan, Graham Neubig, and Pengfei Liu. 2021. BARTScore: Evaluating Generated Text as Text Generation. *Preprint*, arXiv:2106.11520.

Fengzhu Zeng and Wei Gao. 2024a. JustiLM: Few-shot Justification Generation for Explainable Fact-Checking of Real-world Claims. *Transactions of the Association for Computational Linguistics*, 12:334–354.

Fengzhu Zeng and Wei Gao. 2024b. JustiLM: Few-shot justification generation for explainable fact-checking of real-world claims. *Transactions of the Association for Computational Linguistics*, 12:334–354.

Fengzhu Zeng and Wei Gao. 2024c. JustiLM: Few-shot Justification Generation for Explainable Fact-Checking of Real-world Claims. *Transactions of the Association for Computational Linguistics*, 12:334–354. Place: Cambridge, MA Publisher: MIT Press.

A Appendix

A.1 Data Collection

All fact-checks within the period of up to January 2025 were collected in an automated process. All fact-checks are written in English. These fact-checks then served as the basis for retrieving additional information, such as descriptions of the speakers involved, drawn from dedicated pages. This included biographical information and relevant affiliations. Responses were enriched with additional metadata and versioned. Throughout this process, raw data was never altered. Instead, new artifacts were created at each stage to ensure that the pipeline could be reapplied to the original data if necessary. This principle allowed an iterative development of the pipeline, with structured data being extracted progressively, while avoiding unnecessary complexity that could introduce errors.

Internal links found within articles and their sources were processed separately, as they followed different path structures for historical reasons. Since, redirects were online for these links, it was possible to normalize their URLs. This normalization may help to simplify the application of graph-based methods for future research. A particular challenge was posed by the articles themselves.

As with the links, the markup has evolved over time, while changes were not applied to existing articles. Initially, there was an effort to break down the articles into finer components. However, distinguishing between document elements types (e.g., paragraphs, headlines) proved impossible without more complex methods. The multimodal distribution of element counts relative to their length, as shown in Figure 4, illustrates this well. Therefore, a more detailed decomposition of layout elements was avoided in favor of keeping their actual distribution.

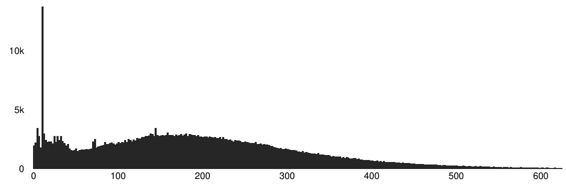

Figure 4: Distribution of document element lengths.

To preserve the spatial arrangement and thus the visual layout that readers see—while converting the articles into plain text and avoiding significant distortion during this transformation—inscriptis from Weichselbraun was used and customized accordingly. The articles, once processed, were then filtered. fact-checks that dealt with multimodal claims or claims by non-professional speakers (e.g., social media claims) were excluded, as they were not the focus of this work. This step is a substantial restriction of the corpus, limiting it to certain research questions.

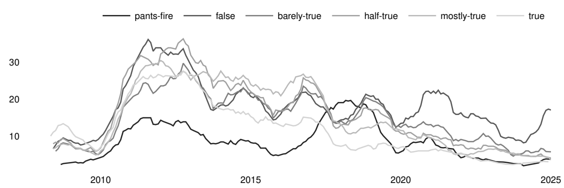

Figure 5: Annual rolling average of verdicts per month on professional claims.

Next, the dataset was cleaned. This involved removing those articles that lacked essential information or had formatting issues leaving them unreadable. The cleaning step was statistically tested for significance on the marginal distributions of verdict label and year of publication. The chi-square goodness-of-fit test was performed to compare the observed label percentages across different years with the expected percentages. The result with $\chi^2 = 0.05, p = 1.0$ suggests that the distributions do not differ significantly.

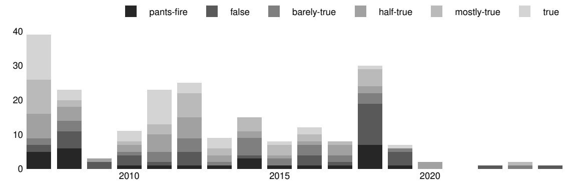

Figure 6: Removed reviews by publication year and verdict.

After the cleaning, the dataset was serialized in RDF 1.1 Turtle[5] format, using schema.org[6] as the foundation for the ontology. This structure enables enhanced query capabilities and ensures interoperability with other linked data sources, making the dataset more accessible.

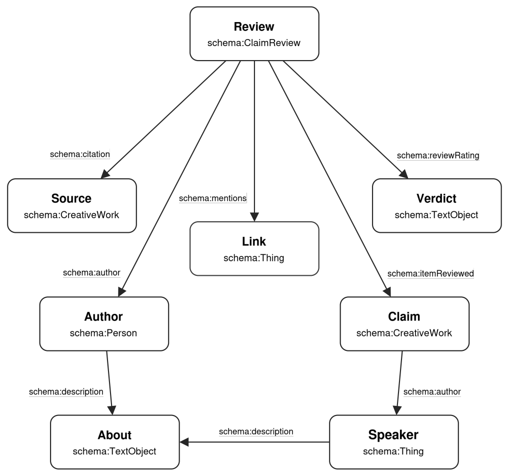

Figure 7: Ontology of the dataset.

The total counts of the key entities are summarized in Table 6, providing a breakdown of the dataset. The final dataset created has been made publicly available on Hugging Face[7].

Entity	Count
Author	661
Link	113, 571
Review	17, 276
Speaker	4796
Source	188, 918

Table 6: Breakdown of entities in the dataset.

For the extraction, we employ the largest LLM we can host on the available hardware with a 32k context length in a few-shot inference scenario. Specifically, we select Llama3.3 70B for inference

[5] https://www.w3.org/TR/turtle
[6] https://schema.org/ClaimReview
[7] https://doi.org/10.82392/hf/123456789

on 8 H100 GPUs using structured generation with the vLLM framework. For the few-shot setting, we annotate 9 articles, aiming to incorporate as many diverse examples as possible within the context length so that the model is exposed to varied extraction patterns when tasked to extract information from a presented article for a given real-world claim.

Guidelines for Human Annotation

Preparation: Before beginning the annotation, carefully read the provided fact-checking article along with its corresponding extracted evidence, decontextualized evidence, and justification. Ensure that you are familiar with all the annotation criteria and understand the overall objective of verifying the quality and utility of the extractions. Annotate each case independently.

Annotation Criteria:

- **Evidence Label Leakage (Q0):**
  - Answer **Yes** or **No** to whether the extracted content reveals the final verdict.
- **Evidence Meaning Preservation (Q1):**
  - Rate (1-5) whether the decontextualized evidence retains the original meaning.
- **Justification Consistency (Q2):**
  - Rate (1-5) whether the justification follows the reasoning in the article.
- **Justification Coverage (Q3):**
  - Rate (1-5) whether the justification is fully grounded in the provided evidences.
- **Justification Utility (Q4):**
  - Rate (1-5) whether the justification sufficiently supports the final verdict.

A.2 Prompts

A.3 Additional Results

B

	Five Class			
Model	**BT**	**BS**	**F1**	**TS**
DeepSeek-R1-Distill-Llama-8B				
Optimal Evidence (D)	0.16	-2.52	71.60	-2.27
Retrieved Evidence (D)	0.15	-2.62	70.58	-2.46
Optimal Evidence (E)	0.15	-2.56	70.79	-2.38
Retrieved Evidence (E)	0.14	-2.64	68.99	-2.48
No Evidence	0.06	-2.70	64.59	-4.10
DeepSeek-R1-Distill-Llama-70B				
Optimal Evidence (D)	0.21	-2.33	79.20	-1.21
Retrieved Evidence (D)	0.21	-2.41	77.71	-1.23
Optimal Evidence (E)	0.21	-2.36	78.89	-1.24
Retrieved Evidence (E)	0.21	-2.43	76.71	-1.31
No Evidence	0.07	-2.51	69.22	-2.92

Table 7: Results of the AFC task with 2 labels for DeepSeek-R1-Distill-Llama models.

	Two Class			
Model	**BT**	**BS**	**F1**	**TS**
DeepSeek-R1-Distill-Llama-8B				
Optimal Evidence (D)	0.16	-2.52	71.60	-2.03
Retrieved Evidence (D)	0.15	-2.62	70.58	-2.17
Optimal Evidence (E)	0.15	-2.56	70.79	-2.12
Retrieved Evidence (E)	0.14	-2.64	68.99	-2.26
No Evidence	0.06	-2.70	64.59	-3.68
DeepSeek-R1-Distill-Llama-70B				
Optimal Evidence (D)	0.21	-2.33	79.20	-1.15
Retrieved Evidence (D)	0.21	-2.41	77.71	-1.23
Optimal Evidence (E)	0.21	-2.36	78.89	-1.17
Retrieved Evidence (E)	0.21	-2.43	76.71	-1.22
No Evidence	0.07	-2.51	69.22	-2.47

Table 8: Results of the AFC task with 2 labels for DeepSeek-R1-Distill-Llama models.

Extraction Prompt

```
SYSTEM:
You are an advanced annotation support system specializing in automated fact-checking.
Your task is to analyze a fact-checking article that adresses a specific claim and produce a structured JSON extraction.

# Input:
1. Context: [SOURCE] stated on [DATE] in [MEDIUM] the claim [CLAIM].
2. Article: The full text of the fact-checking article that addresses [CLAIM].

# Instructions:
1. **Extract Evidence:**
- Objective: Identify and extract all evidence snippet from the article that are presented by the author to verify the claim.
- **Do not** include the claim itself, the final verdict ruling, or any opinion-based statements as evidence.
- Each evidence snippet must be captured exactly as it appears in the article.

2. **Decontextualize Evidence:**
- Objective: For each extracted evidence snippet, create a version that is fully understandable on its own,
  without requiring additional context from the article.
- Clearly identify all entities mentioned in the snippet (e.g., people, institutions, organizations, locations, dates)
  by providing their full names and explicit references.
- Replace any abbreviations, acronyms, or pronouns with their complete forms to eliminate ambiguity.
- Rewrite the evidence so that it is self-contained and understandable independently of the original article's context.

3. **Extract Justification:**
- Locate the section of the article, typically found towards the end, where the author explains the reasoning behind
  their conclusion.
- Extract the text that provides a detailed explanation of the claim's accuracy based on the evidence.

4. **Extract Ruling:**
- Identify the final verdict ruling provided by the article's author regarding the claim.
- Ensure the verdict matches one of the predefined categories.

# Output:
Respond in **valid JSON** with the structure:
{
    "$defs": {
        "Evidence": {
            "properties": {
                "extracted_text_span": {...},
                "decontextualized_text": {...}
            },
        },
        "Justification": {
            "properties": {
                "text": {...}
            },
        },
        "Verdict": {
            "enum": ${LABELS},
        }
    },
    "properties": {
        "evidences": List["Evidence"],
        "justification": {"text": str},
        "verdict": Enum: str
    },
    "required": [
        "evidences",
        "justification",
        "verdict"
    ],
}

USER:
${CONTEXT}
${ARTICLE}

ASSISTANT:
```

Figure 8: Prompt for the extraction of the evidences, justification, and the verdict from PolitiFact fact-checking articles.

Fact-Checking Prompt

```
SYSTEM:
You are a intelligent decision support system for the task of automated fact-checking.
Your task is to analyze a claim made by a public figure based on the presented evidence and produce a structured JSON.

# Input:
1. Context: [SOURCE] stated on [DATE] in [MEDIUM] the claim [CLAIM].
2. Evidence: The evidence retrieved for the verification of [CLAIM].


# Instructions:

1. **Analyze** the [CLAIM] step-by-step. Highlight key arguments, inconsistencies, or gaps in the available evidence.

2. **Classify** the veracity of [CLAIM] based on the on the analysis. Assign a verdict from below labeling scheme:

${LABELS}

3. **Justify the veracity of [CLAIM] briefly with a natural language explanation.
Focus on analyzing the claim based on the presented evidence if available. If evidence is not available, analyze the
claim based on your available knowledge. Adhere to the following criteria:

**Clarity**: The explanation should be concise, coherent and complete.
**Consistency**: The explanation should be consistent with the presented information.
**Relevance**: The explanation must be relevant to the claim and the context provided.
**Utility**: The explanation should be useful in evaluating the prediction of the claim's veracity."

# Output Format:
Respond in **valid JSON** with the structure:
{
    "$defs": {
        "Justification": {
            "properties": {
                "text": {...}
            },
        },
        "Reasoning": {
            "properties": {
                "text": {...}
            },
        },
        "Verdict": {
            "enum": ${LABELS},
        }
    },
    "properties": {
        "reasoning": {"text": str},
        "verdict": Enum: str,
        "justification": {"text": str},
    },
    "required": [
        "reasoning",
        "verdict",
        "justification"
    ],
}

USER:
${CONTEXT}
${EVIDENCE}

ASSISTANT:
```

Figure 9: Prompt for the fact-checking task.

THEMIS Evaluation Prompt

```
SYSTEM:
###Instruction###
Please act as an impartial and helpful evaluator for natural language generation (NLG), and the audience is an expert in the field.
Your task is to evaluate the quality of Automated Fact-Checking strictly based on the given evaluation criterion.
Begin the evaluation by providing your analysis concisely and accurately, and then on the next line, start with
"Rating:" followed by your rating on a Likert scale from 1 to 5 (higher means better).
You MUST keep to the strict boundaries of the evaluation criterion and focus solely on the issues and errors involved;
otherwise, you will be penalized.
Make sure you read and understand these instructions, as well as the following evaluation criterion and example content, carefully.

###Evaluation Criterion###
{Aspect}

### Output Format ###:
Respond in **valid JSON** with the structure:
{
    "$defs": {
        "Analysis": {
            "properties": {
                "text": {...}
            },
        },
        "Rating": int
    },
    "properties": {
        "Analysis": {"text": str},
        "Rating": int,
    },
    "required": [
        "Analysis",
        "Rating",
    ],
}

USER:
###Input###
{Aspect-Input:}

###Outpu###
{Aspect-Output:}

###Your Evaluation###
ASSISTANT:
```

Figure 10: Prompt for the reference-free evaluation with THEMIS targeting as specific aspect of the corpus evaluation.

TIGERSCORE Evaluation Prompt

```
SYSTEM:
###Instruction###
Please act as an impartial and helpful evaluator for natural language generation (NLG), and the audience is an expert in the field.
Your task is to evaluate the quality of Automated Fact-Checking strictly based on the given evaluation criterion.
Begin the evaluation by providing your analysis concisely and accurately, and then on the next line, start with
"Rating:" followed by your rating on a Likert scale from 1 to 5 (higher means better).
You MUST keep to the strict boundaries of the evaluation criterion and focus solely on the issues and errors involved;
otherwise, you will be penalized.
Make sure you read and understand these instructions, as well as the following evaluation criterion and example content, carefully.

###Evaluation Criterion###
Evaluate the claim's veracity, the coherence of the reasoning, and the adequacy of the explanation provided.
The analysis should consider whether the reasoning is logical and supported by the evidence.

### Output Format ###:
Respond in **valid JSON** with the structure:
{
    "$defs": {
        "Analysis": {
            "properties": {
                "text": {...}
            },
        },
        "Rating": int
    },
    "properties": {
        "Analysis": {"text": str},
        "Rating": int,
    },
    "required": [
        "Analysis",
        "Rating",
    ],
}

USER:
###Example###
Claim & Evidence:
Claim: ${CONTEXT}
Evidence: ${EVIDENCE}
True Label: ${ACTUAL_VERDICT}

Model Output:
Reasoning: ${REASONING}
Predicted Label: ${PREDICTED_VERDICT}
Explanation: ${JUSTIFICATION}

###Your Evaluation###
ASSISTANT:
```

Figure 11: Prompt for the reference-free evaluation with TIGERSCORE as a general quality metric for natural language generation.