Explaining Sources of Uncertainty in Automated Fact-Checking

Anonymous ACL submission

Abstract

Human–AI collaboration in knowledge-intensive tasks such as fact-checking requires understanding model uncertainty in multi-document reasoning amid conflicting/agreeing evidence. Yet, existing methods only express uncertainty as numbers or hedges without revealing which evidence conflicts cause the uncertainty, leaving users unable to resolve disagreements. We present CLUE (**C**onflict-&Agreement-aware **L**anguage-model **U**ncertainty **E**xplanations), a plug-and-play white-box framework that, to our knowledge, is the first to generate natural-language explanations of model uncertainty grounded in conflicting/agreeing evidence. CLUE (i) identifies span-level claim–evidence and inter-evidence relations that signal conflict or agreement without supervision, and (ii) uses these relations to steer explanation generation, articulating how they drive the model's uncertainty. Across three language models and two fact-checking datasets, CLUE produces explanations that more faithfully track model uncertainty and better align with the model's fact-checking decisions than span-agnostic explanation prompting; human raters also judge them more helpful, more informative, less redundant, and more logically consistent with the input. By explicitly tying uncertainty to evidence conflicts and agreements, CLUE supports practical fact-checking and other tasks that require reasoning over complex, conflicting information.

1 Introduction

Large Language Models (LLMs) are increasingly prevalent in supporting high-stakes tasks that involve reasoning about information reliability, such as fact-checking (Wang et al., 2024; Fontana et al., 2025). To foster effective use of such models in fact-checking tasks, these models must explain the rationale for their predictions (Atanasova et al., 2020; Kotonya and Toni, 2020).

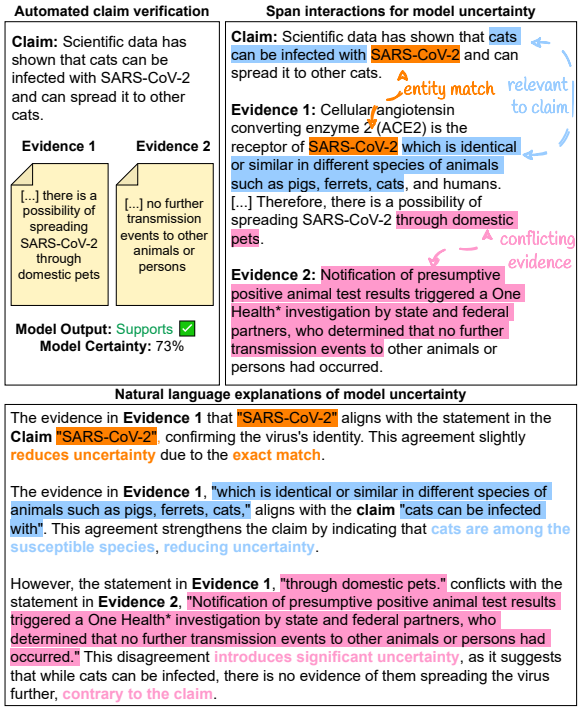

Figure 1: An example of CLUE output for a claim with two evidence passages: model verdict and certainty, and an uncertainty explanation grounded in conflict/agreement span interactions.

However, current methods in automated fact-checking have been criticised for misaligning with the practical needs of fact-checkers, who must assess source reliability and reconcile conflicting evidence, rather than merely predicting a verdict label (Warren et al., 2025; Schlichtkrull et al., 2023). Current approaches often produce explanations to justify the verdict prediction (Atanasova et al., 2020; Stammbach and Ash, 2020; Zeng and Gao, 2024), however, they provide no support for communicating model uncertainty or surfacing agreements and conflicts within the evidence (See Fig. 2). Methods that do express uncertainty do so via numeric scores (e.g., "I am 73% confident") that are hard to contextualise (Zimmer, 1983; Wallsten

et al., 1993; van der Waa et al., 2020; Liu et al., 2020) or via natural language expressions (e.g., "I'm not sure") that often fail to reflect model uncertainty faithfully and can inflate perceived confidence (Steyvers et al., 2025; Yona et al., 2024; Kim et al., 2024). Therefore, existing explainable fact-checking systems usually exhibit two limitations: (1) neglecting to communicate model uncertainty with explanations and (2) failing to surface evidentiary conflicts and agreements that contribute to model uncertainty. This constitutes a fundamental methodological gap, as effective fact-checking requires precisely identifying the sources of uncertainty, for example, from conflicting evidence, to guide targeted verification (Graves, 2017; Micallef et al., 2022).

To bridge this gap, we propose CLUE, a pipeline that generates natural language explanations (NLEs) of model uncertainty by explicitly capturing conflicts and agreements in the input (e.g., a claim and its supporting or refuting evidence), see Fig. 1. The pipeline first identifies the salient span-level interactions that are essential to the prediction of the model through an unsupervised approach, providing an input-feature explanation that highlights key relationships between separate input segments (e.g., claim and evidence) (Ray Choudhury et al., 2023). Previous work has shown that these interactions are faithful to the model and plausible to humans (Sun et al., 2025). CLUE then converts these signals into uncertainty-aware explanations by explicitly referring to the interactions, the conflict/agreement relations they express, and how they contribute to uncertainty regarding the verdict. CLUE does not require gold-label explanations or fine-tuning, and operates entirely at inference time.

Across three language models (§3.2) and two fact-checking datasets (§3.1), we evaluate two variants of CLUE, which leverage distinct methods to guide NLE generation based on the extracted key conflicts/agreements (§2.4). Automatic evaluation shows that both variants generate explanations that are more faithful to each model's uncertainty and agree more closely with the gold fact-checking labels than a prompting baseline that lacks conflict-/agreement-span guidance (§4.5). Human study participants likewise judge CLUE explanations as more helpful, more informative, less redundant, and more logically consistent with the input. We also observe a trade-off between two variants of our CLUE framework, one attains higher faithfulness, the other higher plausibility, highlighting a promising avenue for future work to achieve both simultaneously (§4.5).

2 Method

2.1 Preliminaries and Overall Framework

Our objective is to *explain why* a LLM is uncertain about a multi-evidence fact-checking instance by grounding that uncertainty in specific agreements or conflicts within the input.

Problem setup. Each input instance is a triple $X = (C, E_1, E_2)$ consisting of a claim C and two evidence pieces E_1, E_2. In this work, we explore the scenarios in which one claim and two or three evidence pieces are presented.[1] For clarity, we denote their concatenation as $X = [x_1, \ldots, x_{|C|+|E_1|+|E_2|}]$. The task label comes from the set $\mathcal{Y} = \{\text{SUPPORTS}, \text{REFUTES}, \text{NEUTRAL}\}$.

Pipeline overview. Our framework comprises three stages:

1. **Uncertainty scoring.** We compute *predictive entropy* from the model's answer logits to obtain a scalar uncertainty score $u(X)$ (§ 2.2). This logit-based measure is model-agnostic.
2. **Conflicts/Agreement extraction.** We capture the agreements and conflicts most relevant to the model's reasoning by identifying the text-span interactions between C, E_1, and E_2 that embody these relations (§ 2.3).
3. **Explanation generation.** The model receives the extracted spans as soft constraints and produces a natural language explanation $Y_R = [y'_1, \ldots, y'_r]$ along with its predicted label $\hat{y}$ to the identified interactions (§ 2.4).

Outputs. For each instance X, the framework returns the predicted task label $\hat{y} \in \mathcal{Y}$; the numeric uncertainty score $u(X)$; and the textual explanation $Y_R = [y'_1, \ldots, y'_r]$ that grounds the source of uncertainty in the specific agreements or conflicts between C, E_1, E_2.

2.2 Predictive Uncertainty Score Generation

To quantify model uncertainty for generating an answer label on a specific input sequence, we follow previous work (Kadavath et al., 2022; Yang et al.,

[1] As our framework is based on the pairwise input part interaction, it is readily applicable to scenarios where more claims and evidence pieces are involved.

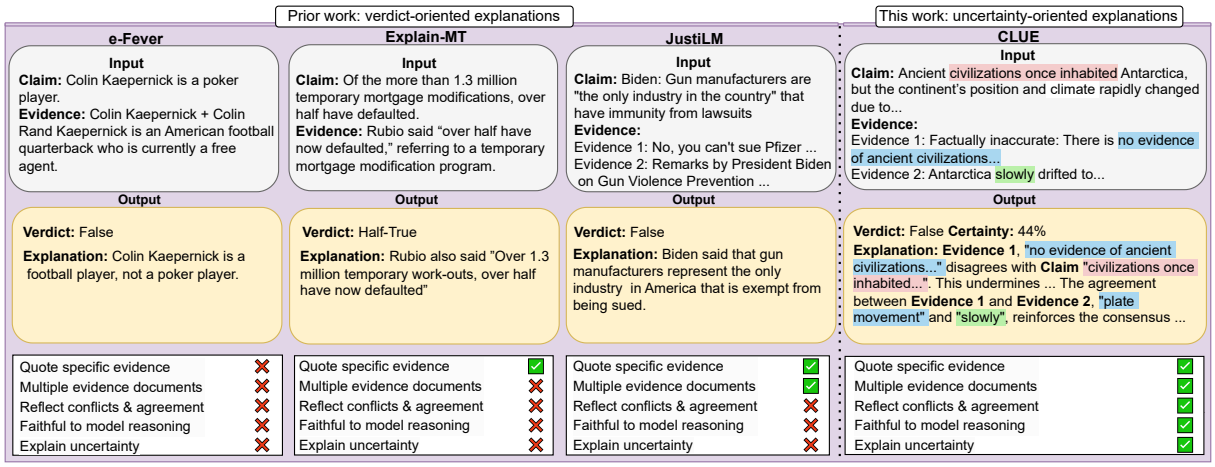

Figure 2: Illustrative comparison of verdict-oriented fact-checking explanations (e-FEVER (Stammbach and Ash, 2020), Explain-MT (Atanasova et al., 2020), JustiLM (Zeng and Gao, 2024)) and CLUE's uncertainty-oriented explanations. In this comparison, CLUE is the only approach that explicitly traces predictive uncertainty to claim-evidence and inter-evidence conflict/agreement across multiple evidence passages.

2025) and calculate predictive uncertainty with entropy theory, which does not require multiple runs and is widely used in open-source models.

Specifically, we define the numeric uncertainty score u as the entropy of the softmax distribution over the model's output logits for a set of candidate answers $\mathcal{Y} = \{\text{SUPPORTS}, \text{REFUTES}, \text{NEUTRAL}\}$. For each candidate label $y_i \in \mathcal{Y}$:

$$P(y_i \mid X) = \frac{\exp(\text{logit}(y_i))}{\sum_{j=1}^{|\mathcal{Y}|} \exp(\text{logit}(y_j))} \quad (1)$$

where $\text{logit}(y_i)$ is the model's output logit towards candidate answer y_i given input X. $P(y_i \mid X)$ is the confidence score of model for selecting y_i as the final answer across all candidate answers within $\mathcal{Y}$. Finally, the model's uncertainty towards the input sequence X is:

$$u(X) = -\sum_{y_i \in \mathcal{Y}} P(y_i \mid X) \log P(y_i \mid X) \quad (2)$$

2.3 Conflict and Agreement Span Interaction Identification for Answer Uncertainty

To surface the conflicts and agreements that drive a model's uncertainty, we extract and then label salient span interactions among the claim C and two evidence passages, E_1 and E_2.

Span interaction extraction. For each ordered input part pair $(F, T) \in \{(C, E_1), (C, E_2), (E_1, E_2)\}$, we follow previous work (Ray Choudhury et al., 2023; Sun et al., 2025) to extract the important span interactions and their importance score to model's answer by (i) identifying the most important attention head to the model's answer prediction from its final layer, (ii) obtaining its attention matrix $\mathbf{A} \in \mathbb{R}^{(|F|+|T|)\times(|F|+|T|)}$, and (iii) symmetrizing the cross-part score $a'_{p,q}$ for each possible token interaction (x_p, x_q) within (F, T):

$$a'_{p,q} = \tfrac{1}{2}\left(\mathbf{A}_{p,q} + \mathbf{A}_{q,p}\right), \quad x_p \in F,\ x_q \in T.$$

Based on these token interactions, we treat each $a'_{p,q}$ as an edge weight in a bipartite token graph between the two input parts (F, T). Then the Louvain algorithm (Blondel et al., 2008) is applied to search for communities of tokens with dense intra-cluster and sparse inter-cluster relationships. From each detected token community, one span interaction, $(\text{span}_w, \text{span}_v)$, can be extracted by treating the neighboring tokens of the same input part, F and T, respectively, as a span. By averaging the token importance score $a'_{p,q}$ of each token interaction within $(\text{span}_w, \text{span}_v)$, its importance score a_{wv} can be obtained by

$$a_{wv} = \frac{1}{|\text{span}_w|\,|\text{span}_v|} \sum_{x_p \in \text{span}_w} \sum_{x_q \in \text{span}_v} a'_{p,q}. \quad (3)$$

The scored interactions for (F, T) form $S^{(F,T)} = \{((\text{span}_w, \text{span}_v),\ a_{wv})\}$.

Relation labeling. To tag each span interaction as *agreement*, *disagreement*, or *unrelated*, we use a relation labeler L_{rel} that can be instantiated with either an open-weight or closed LLM. L_{rel} takes the full (C, E_1, E_2) context together with

a candidate span pair and outputs one label in $\{agree, disagree, unrelated\}$. We implement this using an instruction prompt (see App. C.1) and report the labeling model in Sec. 3.2.

After labelling all three pairs, the complete interaction set for instance X is

$$S_R = S_R^{(C,E_1)} \cup S_R^{(C,E_2)} \cup S_R^{(E_1,E_2)}, \quad (4)$$

where, for example, $S_R^{(C,E_1)} = \{((\text{span}_w, \text{span}_v), a_{wv}, r_{wv})\}$. Each element links two spans with an importance score and a relation label, thereby supplying the conflict- or agreement-span interactions used in later stages.

2.4 Uncertainty Natural Language Explanation Generation

To convert the extracted conflict- and agreement-span interactions into explanations for model uncertainty, we rely on two complementary mechanisms. (i) **Instruction-driven prompting** embeds the spans directly in the input so the model is instructed which segments to reference. (ii) **Intrinsic attention steering** guides the model's own attention toward those same segments while it is generating the NLE. Both mechanisms use *self-rationalization*: the model first states its verdict $\hat{y}$ and then explains Y_R, a sequencing shown to improve faithfulness over pipeline approaches (Wiegreffe et al., 2021; Marasovic et al., 2022; Siegel et al., 2025).

Instruction-driven NLE. For each instance X, we rank all labelled interactions by descending importance scores and keep the top $k = 3$, denoted $S_R(k)$, to avoid overly long explanations. These three span pairs are slotted into a three-shot prompt (See App. G.1), which instructs the model to explain how the highlighted agreements or conflicts influence its confidence. Finally, the standard transformer decoding process outputs both the predicted label $\hat{y}$ and the accompanying explanation Y_R.

Attention steering. Instead of explicit instructions, we can guide NLE generation by modifying attention on the fly following the attention steering method in PASTA (Zhang et al., 2024b). Starting from the same $S_R(k)$, we collect all token indices that fall inside any selected span,

$$\mathcal{I} = \{p \ : \ (\text{span}_w, \text{span}_v) \in S_R(k), \ p \in \text{span}_w \cup \text{span}_v\}. \quad (5)$$

For each attention head (ℓ, h) deemed relevant to model uncertainty, let $\mathbf{A}$ be its attention matrix. We down-weight non-target tokens by β:

$$\tilde{A}_{ij} = \frac{A_{ij}}{Z_i} \begin{cases} 1 & \text{if } j \in \mathcal{I}, \\ \beta & \text{otherwise}, \end{cases} \quad (6)$$

$$Z_i = \sum_{j \in \mathcal{I}} A_{ij} + \beta \sum_{j \notin \mathcal{I}} A_{ij}. \quad (7)$$

All other heads remain unchanged. Following Zhang et al. (2024b), we steer $|H| = 100$ heads and set $\beta = 0.01$ to balance steering efficacy and prevent degeneration; see App. B for the head-selection procedure. With the steered attention in place, the transformer generates $\hat{y}$ followed by the rationale Y_R, now naturally centered on the conflict- or agreement spans that drive its uncertainty.

3 Experimental Setup

3.1 Datasets

We select two fact-checking datasets, one specific to the health domain, HealthVer (Sarrouti et al., 2021), and one closer to a real-world fact-checking scenario, DRUID (Hagström et al., 2024). These datasets were chosen because they provide multiple evidence pieces per claim, making them well-suited to our goal of explaining model uncertainty arising from the inter-evidence conflicts and agreements. For experiments, we select six hundred instances that consist of a claim and multiple pieces of evidence, and a golden label $y \in \{\text{SUPPORTS}, \text{REFUTES}, \text{NEUTRAL}\}$ from each dataset.[2]

3.2 Models

We compare three generation strategies for NLEs of model uncertainty:

- **`Prompt`$_{\text{Baseline}}$**: A three-shot prompt baseline extending prior few-shot NLE work (Stammbach and Ash, 2020; Zeng and Gao, 2024; Zhao et al., 2024) by explicitly asking the model to highlight conflicting or supporting spans that shape its uncertainty (see prompt template in App. G.1).
- **`CLUE-Span`**: The instruction-driven variant of our CLUE method where the extracted span interactions are filled into a three-shot prompt to guide the explanation generation (§2.4; App. G.2).
- **`CLUE-Span+Steering`**: The attention steering variant of our CLUE method in which the same

[2]While DRUID has six fine-grained fact-checking labels, we merge the labels into the above three categories to balance the label categories.

prompt as `CLUE-Span` is used. Additional attention steering is applied to instinctively guide the model's explanation generation toward the identified spans (§2.4; App. G.2).

We run the main experiments on multiple recent open-weight LLMs of comparable scale: Qwen2.5-14B-Instruct (Qwen Team, 2024), Gemma-2-9B-IT (Gemma Team, 2024), and OLMo-2-1124-13B-Instruct (Team OLMo et al., 2024) and DeepSeek-R1-Distill-Qwen-14B (DeepSeek-AI) (model cards in App. D). Each backbone is run through CLUE on a single NVIDIA A100-SXM 80GB GPU; we choose these models to balance instruction-following/reasoning ability with inference efficiency (see cost report in App. E).

For span-interaction relation labeling, we instantiate L_{rel} with GPT-4o (OpenAI Team, 2024) to reduce relation-label noise (see App. C.3 for error propagation analysis). This step is optional: replacing L_{rel} with the open-weight Qwen2.5-14B-Instruct yields comparable downstream NLE quality with a small drop (App. C.2); we discuss this modularity and future directions in Limitations.

4 Automatic Evaluation

4.1 Faithfulness

To assess whether the NLEs produced by CLUE are faithful to the model's uncertainty, we adapt the Correlational Counterfactual Test (CCT) (Siegel et al., 2024) and propose *Entropy-CCT*. Entropy-CCT retains CCT's perturb-and-correlate design, but replaces model's answer probability-shift (e.g., TVD) with *predictive entropy change* to directly target uncertainty, which is the object our NLEs aim to explain (see the App. F.2 for comparison).

Following Siegel et al. (2024), for each instance X we construct n perturbations X' by inserting a random modifier token (adjective before a noun, or adverb before a verb; App. F.1). Let $u(X)$ denote the model's uncertainty score defined by Eq. 2. We measure the perturbation's impact on uncertainty using Absolute Entropy Change (AEC):

$$\Delta u(X) = \left| u(X) - u(X') \right| . \tag{8}$$

For each perturbation X', we also record whether the inserted word appears in the generated NLE, using its presence as a proxy for importance. This yields a binary mention flag $m \in \{0, 1\}$ (Siegel et al., 2024; Atanasova et al., 2023).

Let D_m denote the set of perturbed examples where the NLE *mentions* the inserted word and $D_{\neg m}$ the complementary set where it does not. We compute the point-biserial correlation r_{pb} (Tate, 1954) between the continuous variable Δu and the binary flag m over all perturbations. The Entropy-CCT statistic is:

$$\text{CCT}_{\text{entropy}} = r_{\text{pb}} = \frac{\mathbb{E}_m[\Delta u] - \mathbb{E}_{\neg m}[\Delta u]}{\text{Std}(\Delta u)} \cdot \sqrt{\frac{|D_m| \cdot |D_{\neg m}|}{(|D_m| + |D_{\neg m}|)^2}} . \tag{9}$$

Here, $\mathbb{E}_m[\Delta u]$ and $\mathbb{E}_{\neg m}[\Delta u]$ are the mean AEC values for the two groups, and $\text{Std}(\Delta u)$ is the standard deviation across all perturbations. Higher Entropy-CCT indicates that the NLE more often mentions tokens whose insertion induces larger changes in predictive uncertainty, and thus is more faithful to reflect the model's uncertainty.

4.2 Span-Coverage

An uncertainty explanation should surface *all* information conveyed by the selected span interactions. We therefore compute **Span-Coverage**: the fraction of reference interactions that are explicitly mentioned in the generated NLE. Let S_{NLE} be the set of span interactions extracted from the explanation, and let $S_R(k)$ be the reference set supplied in the prompt (See §2.4). Then

$$\text{Span-Coverage} = \frac{|S_{\text{NLE}} \cap S_R(k)|}{|S_R(k)|} . \tag{10}$$

A higher value indicates the NLE covers a higher proportion of the information supplied by the extracted span interactions.

4.3 Span-Extraneous

Ideally, the explanation should mention *only* the provided interactions and avoid introducing extraneous information. We measure the proportion of mentioned interactions that *do not* belong to the reference set, denoted **Span-Extraneous**:

$$\text{Span-Extraneous} = \frac{|S_{\text{NLE}} \setminus S_R(k)|}{|S_{\text{NLE}}|} . \tag{11}$$

A lower value indicates closer alignment with the intended span interactions.

4.4 Label-Explanation Entailment

We evaluate the extent to which the uncertainty explanation agrees with the model's predicted label by formulating the task as a natural-language inference (NLI) problem. First, we convert the predicted label into a hypothesis using the template *"The claim is supported by / refuted by / neutral to the evidence."* The generated explanation serves

as the premise. The resulting premise–hypothesis pair is fed to a widely used off-the-shelf language-inference model, DeBERTa-v3 (He et al., 2023). The Label-Explanation Entailment (LEE) score is the proportion of examples for which the NLI model predicts ENTAILMENT.

4.5 Results

Here, we present the results of our automatic NLE evaluation for the scenario where the input consists of one *Claim* and two pieces of *Evidence*[3]. For brevity, we refer to Qwen2.5-14B-Instruct, OLMo-2-1124-13B-Instruct, and Gemma-2-9B-it simply as Qwen, OLMo, and Gemma, respectively[4].

Faithfulness. We use Entropy-CCT, a point–biserial correlation r_{pb} (Eq. 9) bounded by $[-1, 1]$ to measure the faithfulness of NLEs to the model's uncertainty (§4.1). When $r_{\text{pb}} = 0$, the explanation mentions high- and low-impact perturbation words equally often; every $+0.01$ adds roughly *one percentage point (pp)* to the chance that the explanation names a token that is *truly influential for the model's predictive uncertainty* (App. F.3).

Table 1 shows that **`Prompt`$_{\text{Baseline}}$ is *non-faithful* in all six settings** with all r_{pb} values negative ranging from -0.03 to -0.13. Thus its NLEs mention truly influential tokens 3–13 pp *less* often than uninfluential ones—the opposite of faithful behaviour. **Both variants of our CLUE reverse this trend.** Presenting span interactions in the prompt (**`CLUE-Span`**) raises every correlation to non-negative values and peaks at $r_{\text{pb}} = 0.089$ on the DRUID–Qwen setting. This means the explanation now mentions about 17 pp more often than **`Prompt`$_{\text{Baseline}}$** ($r_{\text{pb}} = -0.080$). Adding attention steering (**`CLUE-Span+Steering`**) lifts the r_{pb} scores to 0.033 on HEALTHVER and 0.102 on DRUID with Qwen model, i.e., net gains of +6 pp and +18 pp over **`Prompt`$_{\text{Baseline}}$**. Moreover, four of the six positive correlations produced by **`CLUE-Span+Steering`** are significant at $p < 0.01$ (Table 7 in App. F.3), confirming that the improvements are both substantial and statistically reliable. **Particularly large jumps of OLMo on Druid dataset (up to $\Delta r_{\text{pb}} = +0.23 \approx +23$ pp)** suggest that span-interaction guidance from our CLUE framework is most beneficial for models that initially struggle to align explanations with its uncertainty.

Other Properties We evaluate three further properties of the generated NLEs: (i) **Span-Coverage** of extracted conflict-/agreement- span interactions (§4.2), (ii) **Span-Extraneous**: mention of non-extracted spans (§4.3), and (iii) **Label-Explanation Entailment** with the generated fact-checking label (§4.4). As Table 1 shows, **`CLUE-Span+Steering` outperforms `CLUE-Span` in both Span-Coverage and Span-Extraneous**, consistent with the attention steering method's effectiveness in directing the model to focus on provided spans during generation (Zhang et al., 2024b). Absolute numbers, however, remain modest (peak Span-Coverage: .44, Span-Extraneous: .20 with Qwen). A Span-Coverage of 1 means the NLE cites every extracted interaction, while a Span-Extraneous score of 0 means it adds none beyond them. This gap highlights considerable headroom for better integrating critical span interactions into the explanations. Among the three backbones, **Qwen attains the highest Span-Coverage and the lowest Span-Extraneous scores**, a trend that likely reflects its stronger instruction-following ability (see benchmark scores in App. A), and thus larger or more capable models might further narrow the gap. **Both variants of our framework achieve stronger label-explanation entailment scores than the baseline**, yielding explanations logically consistent with the predicted labels while remaining faithful to the model's uncertainty.

5 Human Evaluation

5.1 Method

We recruited N=12 participants from Prolific (`https://www.prolific.com/`) to evaluate 120 explanations generated by **`Prompt`$_{\text{Baseline}}$**, **`CLUE-Span`**, **`CLUE-Span+Steering`** for 40 unique claims (20 from DRUID, 20 from HealthVer) (see details of participants and setup in App. I.1). Adapting Atanasova et al. (2020), participants ranked explanations in descending order (1st, 2nd, 3rd) according to five criteria, complementary to our automatic evaluation metrics:

- **Helpfulness.** The explanation offers information that aids readers to fact-check the claim.
- **Coverage.** The explanation captures *all* salient information in the input that matters for the fact check, distinct from Span-Coverage (§4.2), which counts overlap with pre-extracted spans.

[3] See experiments with three pieces of evidence in App. H.1

[4] See experiments on the reasoning model DeepSeek-R1-Distill-Qwen-14B in App. H.2

Model	Method	HealthVer				DRUID			
		Faith ↑	Cov. ↑	Ext. ↓	LEE. ↑	Faith. ↑	Cov. ↑	Ext. ↓	LEE. ↑
Qwen2.5-14B	**Prompt$_{\text{Baseline}}$**	-0.028	*n/a*	*n/a*	0.74	-0.080	*n/a*	*n/a*	0.60
	CLUE-Span	0.006	0.33	0.68	0.75	0.089	0.20	0.38	**0.78**
	CLUE-Span+Steering	**0.033**	**0.44**	**0.53**	**0.80**	**0.102**	**0.28**	**0.20**	0.77
OLMo-2-1124-13B	**Prompt$_{\text{Baseline}}$**	-0.100	*n/a*	*n/a*	0.55	-0.130	*n/a*	*n/a*	0.53
	CLUE-Span	0.005	0.10	0.83	0.61	0.014	0.08	0.79	0.65
	CLUE-Span+Steering	**0.020**	**0.23**	**0.77**	**0.68**	**0.099**	**0.15**	**0.70**	**0.69**
Gemma-2-9B-IT	**Prompt$_{\text{Baseline}}$**	-0.105	*n/a*	*n/a*	0.66	-0.120	*n/a*	*n/a*	0.57
	CLUE-Span	0.007	0.34	0.59	0.82	0.043	0.23	**0.43**	0.76
	CLUE-Span+Steering	**0.021**	**0.39**	**0.50**	**0.85**	**0.098**	**0.30**	0.47	**0.81**

Table 1: Automatic uncertainty-NLE evaluation. Higher is better for Faith.(§4.1)/Cov. (§4.2)/LEE. (§4.4) and lower is better for Ext. (§4.3). *n/a* indicates Cov./Ext. are undefined for **Prompt$_{\text{Baseline}}$** because no reference spans are provided. Best results per metric for each dataset–model pair are in bold.

- **Non-redundancy.** The explanation does not offer irrelevant or repetitive information to the input, distinct from Span-Extraneous (§4.3) which counts mentions outside the extracted spans.
- **Consistency.** The explanation contains logically consistent statements to the input, distinct from Label-Explanation Entailment (§4.4), which measures label-explanation alignment.
- **Overall Quality.** Ranking of explanations by considering all criteria above.

5.2 Results

Our evaluation shows that explanations generated by CLUE are consistently preferred to Prompt$_{\text{Baseline}}$ (Table 2). Similar to prior work that has evaluated NLEs (Atanasova et al., 2020; Huang et al., 2024; Solano et al., 2024), annotator agreement was moderate to low (App. I.2.1), which we attribute to the complexity of the task.

The explanations generated by our CLUE framework are ranked as participants' first preference 67-78% of the time, compared to 24-34% of the time for Prompt$_{\text{Baseline}}$. The **explanations generated by CLUE are rated as most helpful, containing the least amount of redundant information, highest coverage, consistency, and overall quality approximately twice as often as those generated using Prompt$_{\text{Baseline}}$.**

Although both CLUE-Span and CLUE-Span+Steering outperform Prompt$_{\text{Baseline}}$ in automatic metrics (§4.5) and are ranked higher than Prompt$_{\text{Baseline}}$ in the human study (Table 11 in App. I.2.2), we observe a mild *faithfulness-plausibility* trade-off when comparing the two CLUE variants. **CLUE-Span+Steering** achieves the highest automatic faithfulness, whereas **CLUE-Span** is ranked slightly higher in human-perceived overall quality. A possible reason is that **CLUE-Span+Steering** adheres closely to the top-k=3 span interactions (higher Span-Coverage and lower Span-Extraneous), which improves faithfulness but can reduce fluency/naturalness when some extracted spans are fragmentary. In contrast, **CLUE-Span** may capture additional points that participants deemed important, but not captured by the extracted span interactions (Ray Choudhury et al., 2023). This pattern aligns with the well-documented trade-off between faithfulness and plausibility (Agarwal et al., 2024; Atanasova et al., 2023; Lu and Ma, 2024), and calls for future work to enhance both aspects for improved NLE generation, e.g., by improving the quality of the extracted span interactions.

6 Related Work

6.1 Uncertainty Quantification in LLMs

Recent work estimates LLM uncertainty primarily with white-box, logit-derived measures: predictive entropy of answer distributions (Kadavath et al., 2022; Yang et al., 2025), aggregation across generations (Malinin and Gales, 2021), and response consistency via semantic similarity (Duan et al., 2024; Kuhn et al., 2023; Nikitin et al., 2024). For closed-source models, uncertainty is typically elicited through verbalised confidence (Lin et al., 2022; Mielke et al., 2022), which is overconfident and unreliable (Yona et al., 2024; Tanneru et al., 2024), or approximated by diversity under paraphrased prompts (Zhang et al., 2024a; Chen and Mueller, 2024), which is compute-intensive and conflates prompt noise with model uncertainty. Accordingly,

	$\texttt{Prompt}_{\texttt{Baseline}}$	CLUE
Helpfulness		
Overall	0.281	**0.73**
DRUID	0.312	**0.688**
HealthVer	0.25	**0.772**
Consistency		
Overall	0.290	**0.721**
DRUID	0.309	**0.691**
HealthVer	0.27	**0.751**
Non-redundancy		
Overall	0.252	**0.762**
DRUID	0.261	**0.739**
HealthVer	0.242	**0.784**
Coverage		
Overall	0.275	**0.722**
DRUID	0.283	**0.717**
HealthVer	0.266	**0.727**
Overall Quality		
Overall	0.325	**0.678**
DRUID	0.313	**0.688**
HealthVer	0.336	**0.667**

Table 2: Proportion of times explanations were ranked as participants' first preference for each evaluation metric. Explanations generated by **Qwen2.5-14B-Instruct** (chosen for their high faithfulness; see §4.5)

our method focuses on open-source models and adopts *predictive entropy*, a simple, interpretable, efficient white-box metric computed from answer logits, avoiding prompt-induced noise.

6.2 Linguistic Expressions of Uncertainty

Numerical uncertainty estimates are difficult for end-users to act upon as they do not address sources of uncertainty (Warren et al., 2025). Linguistic expressions of uncertainty, e.g., phrases such as "I'm sure" (Mielke et al., 2022; Tian et al., 2023; Xiong et al., 2023; Ji et al., 2025; Farquhar et al., 2024; Kim et al., 2024), may be more intuitive to understand than numerical ones (Zimmer, 1983; Wallsten et al., 1993; Windschitl and Wells, 1996). However, they are not necessarily faithful reflections of the model's uncertainty (Yona et al., 2024; Tanneru et al., 2024) risking misleading users (Steyvers et al., 2025). Moreover, they do not explain *why* the model is uncertain, as our CLUE method does.

6.3 Generating Natural Language Explanations for Fact-Checking

Natural language explanations (NLEs) justify model predictions for lay readers in fact-checking (Wei Jie et al., 2024), complementing summarization and social-media work that provides graphs, manipulated spans as rationales (Ribeiro et al., 2022; Chan et al., 2023; Huang et al., 2025). In fact-checking, early systems extracted key sentences from supplied articles as explanations (Atanasova et al., 2020); later work improved NLE fluency via post-editing (Jolly et al., 2022) and on data from fact-checking websites (Feher et al., 2025; Raffel et al., 2020; Beltagy et al., 2020). Recently, few-shot approaches avoid fine-tuning by prompting GPT-3 to produce evidence summaries (Brown et al., 2020; Stammbach and Ash, 2020), inserting a planning step (Zhao et al., 2024), and utilizing retrieval-augmented language models (Zeng and Gao, 2024). However, existing methods are often unfaithful to model (Atanasova et al., 2023; Siegel et al., 2024, 2025), have limited utility in fact-checking practice (Schmitt et al., 2024), and fail to address model uncertainty (Warren et al., 2025). Our framework addresses these gaps by explicitly explaining sources of uncertainty in automatic fact-checking, yielding explanations that are both faithful to model uncertainty and provide actionable insights for fact-checkers.

7 Conclusion

We present the first framework, CLUE, for generating NLEs of model uncertainty by referring to the conflicts and agreements between claims and multiple pieces of evidence in a fact-checking task. Our method, evaluated across three language models and two datasets, demonstrates significant improvements in both faithfulness to model uncertainty and label consistency compared to standard prompting. Evaluations by human participants further demonstrate that the explanations generated by CLUE are more helpful, more informative, less redundant, and more logically consistent with the input. This work establishes a foundation for explainable fact-checking systems, providing end users with grounded, faithful explanations that reflect the model's uncertainty. Furthermore, our approach may prove useful for future work in a wide range of information-seeking and retrieval-augmented tasks (e.g., question answering) in which explaining uncertainty given conflicting context is critical.

Limitations

Our paper proposes a novel framework for generating NLEs towards the model's uncertainty by explicitly pointing to the conflicts or agreements between the claim and multiple pieces of evidence. While our framework demonstrates improved explanation quality through rigorous evaluation across multiple language models and datasets, we acknowledge several limitations.

Our experiments are constrained to medium-sized models (Qwen2.5-14B-Instruct, Gemma2-9B-It, OLMo-2-1124-13B-Instruct, and DeepSeek-R1-Distill-Qwen-14B), which are selected based on computational limitations. Although these models show significant improvements over baseline performance, our results suggest that larger models (e.g., 70B parameter scale) may further enhance explanation quality, particularly in terms of coverage and redundancy. Our framework's modular design readily accommodates such scaling.

CLUE includes an auxiliary relation labeler L_{rel} that assigns each extracted span interaction one of{AGREE, DISAGREE, UNRELATED}. In the main experiments we use GPT-4o for this step to reduce relation-label noise, but this choice is optional: L_{rel} is modular, and a fully open-weight variant (e.g., Qwen2.5-14B-Instruct) yields comparable downstream NLE quality with only a small drop (App. C.2). Reducing this remaining gap while remaining fully open-weight may require improving relation-label prompting (more local context, higher-quality shots) or lightly fine-tuning an open-weight model on a compact relation-judgment set (App. C.4).

Like most multi-stage pipelines, CLUE can occasionally propagate upstream errors: manual inspection reveals failure modes such as fragmentary spans and occasional relation mislabels, which can in turn reduce NLE faithfulness (App. C.3). These issues could be mitigated by encouraging more complete semantic spans during extraction (e.g., mild constraints during community detection), improving relation-label robustness (prompting or lightweight fine-tuning), and adding lightweight generation-time checks that down-weight unreliable span interactions before producing the final explanation (App. C.4).

In this study, we focus on the HealthVer and DRUID datasets, which pair claims with discrete pieces of evidence, making them ideal for studying evidence-conflict scenarios. Our experiments primarily focus on setups where one claim and two or three pieces of evidence are presented. Although we did not explore setups involving more complex claim-evidence interactions, when multiple claims and/or pieces of evidence are introduced, our framework, grounded in pairwise interactions, can seamlessly accommodate more complex scenarios by simply increasing the number of input claim/evidence pieces; no methodological changes are required.

CLUE targets settings where users need to understand *why* the model is uncertain by grounding uncertainty in concrete claim-evidence and inter-evidence conflicts/agreements. This focus is motivated by qualitative findings that professional fact-checkers want systems to "show the work" behind a decision, especially when evidence-based tensions drive uncertainty, so that users can decide what to verify next and when to trust the output (Warren et al., 2025). While in this paper, we conduct a human study and find that uncertainty explanations generated by our proposed CLUE framework are more helpful than those from the simple prompting method, we do not study whether these uncertainty-oriented explanations are preferable in every context compared to alternative methods for communicating uncertainty (e.g., numeric confidence (Zimmer, 1983; Wallsten et al., 1993; van der Waa et al., 2020; Liu et al., 2020) or verbal hedges (Lin et al., 2022; Mielke et al., 2022; Yona et al., 2024; Kim et al., 2024)). In-situ evaluation with professional fact-checkers, potentially including interface-level ablation studies (e.g., tuning the number of surfaced interactions k and using compact/expandable formats), would be needed to study the effects of uncertainty explanations on efficiency, error detection, and calibrated trust in other contexts.

Our work is limited to the scope of explaining model uncertainty arising from evidence conflicts. While this captures a critical subset of cases, real-world uncertainty may also stem from other sources, including insufficient evidence, knowledge gaps in the model, and context-memory conflicts. We view this work as a foundational step toward broader research on model uncertainty explanation.

Ethical Considerations

This work concerns automated fact-checking, which aims to reduce the harm and spread of mis-

information, but nevertheless has the potential for harm or misuse through model inaccuracy, hallucination, or deployment for censorship. Our current work aims to provide explanations that allow users to examine the outputs of these systems more critically, and so we do not see any immediate risks associated with it.

Our work is limited to examining claims, evidence, and explanations in English, and so our results may not be generalisable to other languages. As the task involved complex reasoning about technical subjects, we screened our participants to be native English speakers to ensure that they could fully understand the material and increase the chances of high-quality responses (see I.1 for details). However, this criteria may also introduce or reinforce existing biases and limit the generalisability of our findings. Participants were informed about the study and its aims before agreeing to provide informed consent. No personal data was collected from participants and they received fair payment for their work (approximately 9 GBP/hour).

References

Chirag Agarwal, Sree Harsha Tanneru, and Himabindu Lakkaraju. 2024. Faithfulness vs. plausibility: On the (un)reliability of explanations from large language models. *Preprint*, arXiv:2402.04614.

Pepa Atanasova, Oana-Maria Camburu, Christina Lioma, Thomas Lukasiewicz, Jakob Grue Simonsen, and Isabelle Augenstein. 2023. Faithfulness tests for natural language explanations. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 283–294, Toronto, Canada. Association for Computational Linguistics.

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.

Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The long-document transformer.

Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural Language Processing with Python*. O'Reilly Media.

Vincent D. Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. 2008. Fast Unfolding of Communities in Large Networks. *Journal of statistical mechanics: theory and experiment*, 2008(10):P10008.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D 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 Ziegler, Jeffrey Wu, Clemens Winter, Chris 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. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Hou Pong Chan, Qi Zeng, and Heng Ji. 2023. Interpretable automatic fine-grained inconsistency detection in text summarization. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 6433–6444, Toronto, Canada. Association for Computational Linguistics.

Jiuhai Chen and Jonas Mueller. 2024. Quantifying uncertainty in answers from any language model and enhancing their trustworthiness. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5186–5200, Bangkok, Thailand. Association for Computational Linguistics.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

DeepSeek-AI. Deepseek-r1-distill-qwen-14b. `https://huggingface.co/deepseek-ai/DeepSeek-R1-Distill-Qwen-14B`. HuggingFace model card; accessed 2025-10-05.

Jinhao Duan, Hao Cheng, Shiqi Wang, Alex Zavalny, Chenan Wang, Renjing Xu, Bhavya Kailkhura, and Kaidi Xu. 2024. Shifting attention to relevance: Towards the predictive uncertainty quantification of free-form large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5050–5063, Bangkok, Thailand. Association for Computational Linguistics.

Sebastian Farquhar, Jannik Kossen, Lorenz Kuhn, and Yarin Gal. 2024. Detecting Hallucinations in Large Language Models using Semantic Entropy. *Nature*, 630(8017):625–630.

Darius Feher, Abdullah Khered, Hao Zhang, Riza Batista-Navarro, and Viktor Schlegel. 2025. Learning to Generate and Evaluate Fact-Checking Explanations with Transformers. *Engineering Applications of Artificial Intelligence*, 139:109492.

Nicolo' Fontana, Francesco Corso, Enrico Zuccolotto, and Francesco Pierri. 2025. Evaluating open-source large language models for automated fact-checking. *Preprint*, arXiv:2503.05565.

Gemma Team. 2024. Gemma: Open models based on gemini research and technology.

Lucas Graves. 2017. Anatomy of a fact check: Objective practice and the contested epistemology of fact checking. *Communication, culture & critique*, 10(3):518–537.

Lovisa Hagström, Sara Vera Marjanović, Haeun Yu, Arnav Arora, Christina Lioma, Maria Maistro, Pepa Atanasova, and Isabelle Augenstein. 2024. A Reality Check on Context Utilisation for Retrieval-Augmented Generation. *Preprint*, arXiv:2412.17031.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2023. Debertav3: Improving deberta using electra-style pretraining with gradient-disentangled embedding sharing. *Preprint*, arXiv:2111.09543.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. In *International Conference on Learning Representations*.

Matthew Honnibal and Ines Montani. 2017. spacy 2: Natural language understanding with bloom embeddings, convolutional neural networks and incremental parsing.

Fan Huang, Haewoon Kwak, Kunwoo Park, and Jisun An. 2024. ChatGPT rates natural language explanation quality like humans: But on which scales? In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 3111–3132, Torino, Italia. ELRA and ICCL.

Kung-Hsiang Huang, Hou Pong Chan, Kathleen McKeown, and Heng Ji. 2025. ManiTweet: A new benchmark for identifying manipulation of news on social media. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 11161–11180, Abu Dhabi, UAE. Association for Computational Linguistics.

Ziwei Ji, Lei Yu, Yeskendir Koishekenov, Yejin Bang, Anthony Hartshorn, Alan Schelten, Cheng Zhang, Pascale Fung, and Nicola Cancedda. 2025. Calibrating Verbal Uncertainty as a Linear Feature to Reduce Hallucinations. *arXiv preprint arXiv:2503.14477*.

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

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, et al. 2022. Language Models (Mostly) Know What They Know. *arXiv preprint arXiv:2207.05221*.

Maurice G Kendall and B. Babington Smith. 1939. The problem of m rankings. *The annals of mathematical statistics*, 10(3):275–287.

Sunnie S. Y. Kim, Q. Vera Liao, Mihaela Vorvoreanu, Stephanie Ballard, and Jennifer Wortman Vaughan. 2024. "i'm not sure, but...": Examining the impact of large language models' uncertainty expression on user reliance and trust. In *Proceedings of the 2024 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '24, page 822–835, New York, NY, USA. Association for Computing Machinery.

Neema Kotonya and Francesca Toni. 2020. Explainable Automated Fact-Checking: A Survey. *arXiv preprint*. ArXiv:2011.03870 [cs].

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. 2023. Semantic Uncertainty: Linguistic Invariances for Uncertainty Estimation in Natural Language Generation. *arXiv preprint arXiv:2302.09664*.

Stephanie C. Lin, Jacob Hilton, and Owain Evans. 2022. Teaching Models to Express Their Uncertainty in Words. *Transactions on Machine Learning Research*. https://openreview.net/forum?id=8s8K2UZGTZ.

Dawn Liu, Marie Juanchich, Miroslav Sirota, and Sheina Orbell. 2020. The Intuitive Use of Contextual Information in Decisions Made with Verbal and Numerical Quantifiers. *Quarterly Journal of Experimental Psychology*, 73(4):481–494.

Xiaolei Lu and Jianghong Ma. 2024. Does faithfulness conflict with plausibility? an empirical study in explainable ai across nlp tasks. *arXiv preprint arXiv:2404.00140*.

Andrey Malinin and Mark J. F. Gales. 2021. Uncertainty Estimation in Autoregressive Structured Prediction. In *Proceedings of the 9th International Conference on Learning Representations (ICLR 2021)*.

Ana Marasovic, Iz Beltagy, Doug Downey, and Matthew Peters. 2022. Few-shot self-rationalization with natural language prompts. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 410–424, Seattle, United States. Association for Computational Linguistics.

Nicholas Micallef, Vivienne Armacost, Nasir Memon, and Sameer Patil. 2022. True or false: Studying the work practices of professional fact-checkers. *Proc. ACM Hum.-Comput. Interact.*, 6(CSCW1).

Sabrina J. Mielke, Arthur Szlam, Emily Dinan, and Y-Lan Boureau. 2022. Reducing conversational agents' overconfidence through linguistic calibration. *Transactions of the Association for Computational Linguistics*, 10:857–872.

George A. Miller. 1995. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

Alexander Nikitin, Jannik Kossen, Yarin Gal, and Pekka Marttinen. 2024. Kernel language entropy: Fine-grained uncertainty quantification for llms from semantic similarities. 37:8901–8929.

OpenAI Team. 2024. Gpt-4o system card. *Preprint*, arXiv:2410.21276.

Qwen Team. 2024. Qwen2.5: A party of foundation models.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *Journal of machine learning research*, 21(140):1–67.

Sagnik Ray Choudhury, Pepa Atanasova, and Isabelle Augenstein. 2023. Explaining interactions between text spans. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12709–12730, Singapore. Association for Computational Linguistics.

Leonardo F. R. Ribeiro, Mengwen Liu, Iryna Gurevych, Markus Dreyer, and Mohit Bansal. 2022. FactGraph: Evaluating factuality in summarization with semantic graph representations. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3238–3253, Seattle, United States. Association for Computational Linguistics.

Mourad Sarrouti, Asma Ben Abacha, Yassine Mrabet, and Dina Demner-Fushman. 2021. Evidence-based fact-checking of health-related claims. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 3499–3512, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Michael Schlichtkrull, Nedjma Ousidhoum, and Andreas Vlachos. 2023. The intended uses of automated fact-checking artefacts: Why, how and who. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 8618–8642, Singapore. Association for Computational Linguistics.

Vera Schmitt, Luis-Felipe Villa-Arenas, Nils Feldhus, Joachim Meyer, Robert P. Spang, and Sebastian Möller. 2024. The role of explainability in collaborative human-ai disinformation detection. In *Proceedings of the 2024 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '24, page 2157–2174, New York, NY, USA. Association for Computing Machinery.

Noah Siegel, Oana-Maria Camburu, Nicolas Heess, and Maria Perez-Ortiz. 2024. The probabilities also matter: A more faithful metric for faithfulness of free-text explanations in large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 530–546, Bangkok, Thailand. Association for Computational Linguistics.

Noah Y Siegel, Nicolas Heess, Maria Perez-Ortiz, and Oana-Maria Camburu. 2025. Faithfulness of LLM Self-Explanations for Commonsense Tasks: Larger Is Better, and Instruction-Tuning Allows Trade-Offs but Not Pareto Dominance. *arXiv preprint arXiv:2503.13445*.

Jesus Solano, Mardhiyah Sanni, Oana-Maria Camburu, and Pasquale Minervini. 2024. SparseFit: Few-shot prompting with sparse fine-tuning for jointly generating predictions and natural language explanations. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2053–2077, Bangkok, Thailand. Association for Computational Linguistics.

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.

Mark Steyvers, Heliodoro Tejeda, Aakriti Kumar, Catarina Belem, Sheer Karny, Xinyue Hu, Lukas W Mayer, and Padhraic Smyth. 2025. What large language models know and what people think they know. *Nature Machine Intelligence*, pages 1–11.

Jingyi Sun, Pepa Atanasova, and Isabelle Augenstein. 2025. Evaluating input feature explanations through a unified diagnostic evaluation framework. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 10559–10577, Albuquerque, New Mexico. Association for Computational Linguistics.

Sree Harsha Tanneru, Chirag Agarwal, and Himabindu Lakkaraju. 2024. Quantifying Uncertainty in Natural Language Explanations of Large Language Models. In *Proceedings of The 27th International Conference on Artificial Intelligence and Statistics*, volume 238 of *Proceedings of Machine Learning Research*, pages 1072–1080. PMLR.

Robert F Tate. 1954. Correlation between a Discrete and a Continuous Variable. Point-Biserial Correlation. *The Annals of mathematical statistics*, 25(3):603–607.

Team OLMo, Pete Walsh, Luca Soldaini, Dirk Groeneveld, Kyle Lo, Shane Arora, Akshita Bhagia, Yuling Gu, Shengyi Huang, Matt Jordan, Nathan Lambert, Dustin Schwenk, Oyvind Tafjord, Taira Anderson, David Atkinson, Faeze Brahman, Christopher Clark, Pradeep Dasigi, Nouha Dziri, Michal Guerquin, Hamish Ivison, Pang Wei Koh, Jiacheng Liu, Saumya Malik, William Merrill, Lester James V. Miranda, Jacob Morrison, Tyler Murray, Crystal Nam, Valentina Pyatkin, Aman Rangapur, Michael Schmitz, Sam Skjonsberg, David Wadden, Christopher Wilhelm, Michael Wilson, Luke Zettlemoyer, Ali Farhadi, Noah A. Smith, and Hannaneh Hajishirzi. 2024. 2 olmo 2 furious.

Katherine Tian, Eric Mitchell, Allan Zhou, Archit Sharma, Rafael Rafailov, Huaxiu Yao, Chelsea Finn, and Christopher Manning. 2023. Just ask for calibration: Strategies for eliciting calibrated confidence

scores from language models fine-tuned with human feedback. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5433–5442, Singapore. Association for Computational Linguistics.

Jasper van der Waa, Tjeerd Schoonderwoerd, Jurriaan van Diggelen, and Mark Neerincx. 2020. Interpretable confidence measures for decision support systems. *International Journal of Human-Computer Studies*, 144:102493.

Thomas S. Wallsten, David V. Budescu, Rami Zwick, and Steven M. Kemp. 1993. Preferences and Reasons for Communicating Probabilistic Information in Verbal or Numerical Terms. *Bulletin of the Psychonomic Society*, 31(2):135–138.

Yuxia Wang, Revanth Gangi Reddy, Zain Muhammad Mujahid, Arnav Arora, Aleksandr Rubashevskii, Jiahui Geng, Osama Mohammed Afzal, Liangming Pan, Nadav Borenstein, Aditya Pillai, Isabelle Augenstein, Iryna Gurevych, and Preslav Nakov. 2024. Factcheck-bench: Fine-grained evaluation benchmark for automatic fact-checkers. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 14199–14230, Miami, Florida, USA. Association for Computational Linguistics.

Greta Warren, Irina Shklovski, and Isabelle Augenstein. 2025. Show me the work: Fact-checkers' requirements for explainable automated fact-checking. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*, CHI '25, New York, NY, USA. Association for Computing Machinery.

Yeo Wei Jie, Ranjan Satapathy, Rick Goh, and Erik Cambria. 2024. How interpretable are reasoning explanations from prompting large language models? In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 2148–2164, Mexico City, Mexico. Association for Computational Linguistics.

Sarah Wiegreffe, Ana Marasović, and Noah A. Smith. 2021. Measuring association between labels and free-text rationales. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 10266–10284, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Paul D Windschitl and Gary L Wells. 1996. Measuring Psychological Uncertainty: Verbal versus Numeric Methods. *Journal of Experimental Psychology: Applied*, 2(4):343.

Miao Xiong, Zhiyuan Hu, Xinyang Lu, Yifei Li, Jie Fu, Junxian He, and Bryan Hooi. 2023. Can LLMs Express Their Uncertainty? An Empirical Evaluation of Confidence Elicitation in LLMs. *arXiv preprint arXiv:2306.13063*.

Yongjin Yang, Haneul Yoo, and Hwaran Lee. 2025. MAQA: Evaluating uncertainty quantification in LLMs regarding data uncertainty. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 5846–5863, Albuquerque, New Mexico. Association for Computational Linguistics.

Gal Yona, Roee Aharoni, and Mor Geva. 2024. Can large language models faithfully express their intrinsic uncertainty in words? In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 7752–7764, Miami, Florida, USA. Association for Computational Linguistics.

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

Caiqi Zhang, Fangyu Liu, Marco Basaldella, and Nigel Collier. 2024a. LUQ: Long-text uncertainty quantification for LLMs. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 5244–5262, Miami, Florida, USA. Association for Computational Linguistics.

Qingru Zhang, Chandan Singh, Liyuan Liu, Xiaodong Liu, Bin Yu, Jianfeng Gao, and Tuo Zhao. 2024b. Tell your model where to attend: Post-hoc attention steering for LLMs. In *Proceedings of the Twelfth International Conference on Learning Representations (ICLR 2024)*.

Xiaoyan Zhao, Lingzhi Wang, Zhanghao Wang, Hong Cheng, Rui Zhang, and Kam-Fai Wong. 2024. PACAR: Automated Fact-Checking with Planning and Customized Action Reasoning Using Large Language Models. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 12564–12573.

Alf C Zimmer. 1983. Verbal vs. Numerical Processing of Subjective Probabilities. In *Advances in psychology*, volume 16, pages 159–182. Elsevier.

A Backbone model performance on public benchmarks

Table 3 summarises the publicly reported five-shot results on two standard reasoning benchmarks. All figures are taken verbatim from the official model cards or accompanying technical reports. Figures are copied from the official model cards.

These numbers corroborate our claim that Qwen2.5-14B-Instruct is the strongest of the three for instruction-following and reasoning.

B Method: Selecting attention heads to steer

Following Zhang et al. (2024b), we steer only a selected subset of attention heads rather than all of them, because targeted steering yields larger gains in output quality. Our selection criterion, however,

Model	Params	MMLU	GSM8K
Qwen2.5-14B-Instruct (Qwen Team, 2024)	14.7 B	79.7	90.2
Gemma-2-9B-IT (Gemma Team, 2024)	9.0 B	71.3	68.6
OLMo-2-1124-13B-Instruct (Team OLMo et al., 2024)	13 B	67.5	54.2

Table 3: Benchmark scores on MMLU (Hendrycks et al., 2021) and GSM8K (Cobbe et al., 2021) are used to characterize instruction-following and reasoning strength.

differs from theirs: instead of ranking heads by their impact on task accuracy, we rank them by how strongly they affect the model's *predictive uncertainty* during fact-checking.

Concretely, for each fact-checking dataset chosen in this work(see details in §3.1), D, we draw a validation subset D_d with $|D_d| = 300$ examples. For every input $X \in D_d$, we compute the model's baseline uncertainty score $u(X)$ when it predicts the fact-checking label as stated in §2.2. Then, for each attention head identified by layer ℓ and index h, we zero out that head, re-run the model, and measure the absolute change in uncertainty

$$\Delta u(X, \ell, h) \;=\; \big|u(X) \;-\; u_{/o(l,h)}(X)\big|.$$

Averaging $\Delta u(X, l, h)$ over all $X \in D_d$ yields a single importance score for head (ℓ, h). We rank the heads by this score and keep the top t heads for each dataset and each model. Note that we set $t = 100$ in line with the recommendation of Zhang et al. (2024b) and to balance steering effectiveness against the risk of degeneration.

C Conflict and Agreement Identification Details

C.1 Prompt Example for Assigning Relation Labels to Captured Span Interactions

To identify agreements and conflicts between the claim and the two evidence passages, we use the prompt in Figure 3 to label each extracted span interaction (see §2.3).

```
You are a helpful assistant. Your task:

1. Read the claim and its two evidence passages (E1,
      E2).
2. For each supplied span interaction, decide
     whether the two spans
   AGREE, DISAGREE, or are UNRELATED, taking the
        full context into account.
3. Output the span pairs exactly as given, followed
     by
   "relation: agree|disagree|unrelated".

Return format:
  1. "SPAN A" - "SPAN B"  relation: <agree|disagree|
       unrelated>
  2. ...
  3. ...

### SHOT 1  (annotated example)
Claim: [...]
Evidence 1: [...]
Evidence 2: [...]

Span interactions (to be labelled):
  1. "[...]" - "[...]"
  2. "[...]" - "[...]"
  3. "[...]" - "[...]"

Expected output:
  1. "[...]" - "[...]"  relation: ...
  2. "[...]" - "[...]"  relation: ...
  3. "[...]" - "[...]"  relation: ...

### SHOT 2   % omitted for brevity
### SHOT 3   % omitted for brevity

### NEW INSTANCE  (pre-filled for each new example)
Claim: {CLAIM}
Evidence 1: {E1}
Evidence 2: {E2}
Span interactions:
  1. "{SPAN1-A}" - "{SPAN1-B}"
  2. "{SPAN2-A}" - "{SPAN2-B}"
  3. "{SPAN3-A}" - "{SPAN3-B}"
```

Figure 3: Prompt template for span interaction relation labelling.

C.2 Open-weight Alternative for Span-Interaction Relation Labeling

This section studies how the choice of relation labeler affects downstream NLE quality in CLUE. While we use GPT-4o as the default relation labeler to reduce labeling noise, we show that a strong open-weight alternative is also feasible, with only a small performance drop.

As discussed in Sec. 2.3, CLUE uses a relation labeler L_{rel} to assign each extracted span interaction one of $\{\text{AGREE}, \text{DISAGREE}, \text{UNRELATED}\}$. We considered two labeler options: (i) a closed-API model, which can reduce label noise at the cost of API usage, and (ii) an open-weight model, which is fully local but may yield slightly noisier labels and thus introduce additional noise in downstream NLE generation.

To quantify the impact of the labeler choice, we reran the pipeline with one open-weight alternative for L_{rel}: Qwen2.5-14B-Instruct, keeping all other components fixed. Table 4 reports the results.

Overall, Qwen2.5-14B-Instruct yields similar, though slightly weaker, downstream NLE quality

Relation labeler	HealthVer				DRUID			
	Faith. (↑)	Span-Cov. (↑)	Span-Ext. (↓)	LEE (↑)	Faith. (↑)	Span-Cov. (↑)	Span-Ext. (↓)	LEE (↑)
GPT-4o	**0.033**	**0.44**	**0.53**	**0.80**	**0.102**	**0.28**	**0.20**	**0.77**
Qwen2.5-14B-Instruct	0.025	0.41	0.58	0.78	0.076	0.25	0.25	0.75

Table 4: Effect of span-interaction *relation labeler* choice. We rerun CLUE while swapping only L_{rel} (GPT-4o vs. Qwen2.5-14B-Instruct) and keeping all other components fixed (backbone: Qwen2.5-14B-Instruct; variant: CLUE-Span+Steering; k=3). An open-weight labeler is feasible with only a small drop in downstream NLE quality. Bold indicates the better result per metric within each dataset.

compared to GPT-4o across both datasets. For instance, on HealthVer, Faithfulness drops from 0.033 to 0.025 and LEE from 0.80 to 0.78; on DRUID, Faithfulness drops from 0.102 to 0.076 and LEE from 0.77 to 0.75.

These results indicate that an open-weight labeler is feasible in CLUE, and the framework is not tied to a proprietary component. In this work, we use GPT-4o in the main experiments because it provides consistently stronger relation labeling and reduces the risk of error propagation from noisy relation labels to the final explanations. We further discuss how upstream labeling errors can propagate and mitigation strategies in App. C.3.

A natural concern is whether using a closed-API labeler compromises transparency. We emphasize that CLUE is *white-box with respect to the backbone model*: uncertainty scoring, span-interaction extraction, and attention steering require access to the backbone model internals, which closed APIs do not provide. Accordingly, running the *entire* pipeline with GPT-4o would forgo these white-box signals and reduce the method to surface-level prompting; prior work shows such verbalized uncertainty can be unreliable, and fact-checking practice requires systems to "show the work" behind a decision (Yona et al., 2024; Tanneru et al., 2024; Warren et al., 2025).

GPT-4o is used only for the narrow task of pairwise relation judgment *after* span interactions have already been extracted from backbone internals; as such, it does not affect the transparency of the core framework.

C.3 Error Propagation Analysis

We manually inspected 50 HealthVer instances (top-3 span pairs per instance; N=150 span pairs) to assess (i) span extraction quality and (ii) relation labeling accuracy when using GPT-4o as L_{rel}.

Span extraction quality. Most extracted spans are very short: 105/150 span pairs contain two-token spans (often salient nouns such as subjects/objects). In 35/150 cases, at least one span is incomplete (e.g., "D" instead of "vitamin D") or semantically underspecified (e.g., "is not"). Overall, the extractor tends to favor short, high-salience tokens, suggesting potential for future work to encourage more complete semantic units.

Relation labelling accuracy. On these 150 span pairs, GPT-4o achieves 86% accuracy (129/150 correct). Among the 21 errors, 12 are cases where an AGREE relation is mislabelled as UNRELATED, typically when one span is incomplete (e.g., "D" vs. "vitamin D"), indicating that span completeness is a primary failure mode.

Propagation to downstream NLEs. Using Qwen2.5-14B-Instruct with CLUE-Span+Steering, 13/21 mislabeled interactions are explicitly mentioned in the generated NLEs, showing that upstream errors can propagate into explanations. Moreover, the subset of instances with mislabeled interactions exhibits lower faithfulness (mean Entropy-CCT 0.020) than the overall HealthVer–Qwen CLUE-Span+Steering average (0.033; Table 1), suggesting that incorrect relation labels can reduce explanation faithfulness.

This analysis suggests that fragmentary spans and relation-label noise can be an upstream error source and carry forward to the final NLEs, motivating mitigation strategies discussed in App. C.4 for future work.

C.4 Mitigating Error Propagation

We outline several directions to mitigate error propagation in future work: (1) **Improve span extraction:** add mild constraints during community detection so extracted spans more consistently capture complete semantic units rather than clipped tokens. (2) **Improve relation labeling robustness:** use stronger prompting (broader local context, higher-quality example shots) or lightly fine-tune an open-weight model on a compact relation-

judgment set. (3) **Generation-time sanity checks:** add a lightweight consistency check that flags or down-weights span pairs judged internally inconsistent with the claim/evidence before producing the final NLE.

D Resources and Model Cards

Model cards: Qwen2.5-14B-Instruct (Hugging Face); Gemma-2-9B-IT (Hugging Face); OLMo-2-1124-13B-Instruct (Hugging Face); DeepSeek-R1-Distill-Qwen-14B (Hugging Face); DeBERTa-v3 NLI (Hugging Face); GPT-4o (OpenAI).

E Computational Cost and Resource Requirements

Setup. We report the computational overhead of CLUE compared to $\texttt{Prompt}_{\texttt{Baseline}}$. Unless otherwise stated, measurements are collected using Qwen2.5-14B-Instruct on the HealthVer dataset with a single NVIDIA A100 80GB GPU. All values are reported *per instance* (averaged over the evaluation set).

E.1 Latency

Table 5 reports the average wall-clock time per instance for the three methods. Both `CLUE-Span` and `CLUE-Span+Steering` are roughly 2 seconds slower per instance than $\texttt{Prompt}_{\texttt{Baseline}}$, and this overhead is entirely due to the additional span-interaction extraction and relation labeling stage.

E.2 GPU memory usage

Table 6 reports peak GPU memory usage (VRAM). During NLE generation, VRAM usage is similar across all methods. `CLUE-Span+Steering` requires about 0.32 GiB more VRAM than $\texttt{Prompt}_{\texttt{Baseline}}$, primarily because the prompt is longer (it includes the extracted span interactions; see App. G.2). Notably, `CLUE-Span+Steering` does not require more VRAM than `CLUE-Span`, since during generation KV-cache and activations dominate memory and steering does not add a separate memory footprint beyond the base forward pass.

E.3 Memory requirement for steering 100 heads

For selecting the most important 100 attention heads (App. B), the peak VRAM is 29.8 GiB. For steering these heads during NLE generation (`CLUE-Span+Steering`), the average peak VRAM per instance is 30.84 GiB. Thus, a GPU with 40GB of memory satisfies the requirement for attention steering across 100 heads.

E.4 GPT-4o API cost for relation labeling

When using GPT-4o as L_{rel}, the total relation-labeling cost scales linearly with the number of labeled instances. In our main experiments, the cumulative cost across datasets and backbones was approximately \$30 (about \$0.003 per instance), with an average input length of $\sim$1k tokens and an average output length of $\sim$50 tokens per call.

F Methodology Supplement: Faithfulness Evaluation Details

This supplement provides implementation and statistical details for the Entropy-CCT faithfulness metric introduced in §4.1. We describe (i) how counterfactual perturbations are generated, (ii) how Entropy-CCT relates to the original CCT, and (iii) how we test statistical significance and report results.

F.1 Entropy-CCT Perturbation Generation

To evaluate how faithfully each NLE reflects model uncertainty, we generate multiple counterfactuals per instance, following Atanasova et al. (2020) and Siegel et al. (2024) (see §4.1). For every input, comprising one claim and two evidence passages, we first tag part-of-speech with spaCy (Honnibal and Montani, 2017) (`en_core_web_sm`), then choose four random insertion sites (uniformly among tokens tagged as NOUN or VERB). At each site we insert either (i) a random adjective before a noun or (ii) a random adverb before a verb. The candidate modifiers are drawn uniformly from WordNet (Miller, 1995) using NLTK (Bird et al., 2009): we construct adjective/adverb candidate lists by collecting all lemma names from all WordNet adjective/adverb synsets, respectively. Because we sample three random candidates for each of the four positions, this procedure yields $4 \times 3 = 12$ perturbations per instance, providing a sufficient set for the subsequent Entropy-CCT evaluation, in which we check whether the NLE mentions the inserted word and compute the correlation between that mention and the uncertainty change induced by each perturbation.

F.2 Entropy-CCT vs. Original CCT

CCT (Siegel et al., 2024) evaluates faithfulness by perturbing the input and measuring the resulting

Approach	Uncertainty Scoring (s)	Span Extraction + Labeling (s)	NLE Generation (s)	Total (s)
$\texttt{Prompt}_{\texttt{Baseline}}$	0.364	–	14.8969	15.2609
CLUE-Span	0.364	2.0452	14.7730	17.1822
CLUE-Span+Steering	0.364	2.0452	14.7874	17.1966

Table 5: Average time cost per instance for the three NLE generation methods on HealthVer using Qwen2.5-14B-Instruct on a single NVIDIA A100 80GB GPU. "–" indicates that span extraction/labeling is not used by $\texttt{Prompt}_{\texttt{Baseline}}$.

Approach	Uncertainty Scoring (GiB)	Span Interaction Extraction (GiB)	NLE Generation (GiB)
$\texttt{Prompt}_{\texttt{Baseline}}$	29.71	–	30.52
CLUE-Span	29.71	29.72	30.84
CLUE-Span+Steering	29.71	29.72	30.84

Table 6: Average peak VRAM per instance for the three NLE generation methods (HealthVer, Qwen2.5-14B-Instruct, NVIDIA A100 80GB).

change in the model's predicted label distribution. Specifically, it uses the Total Variation Distance (TVD) between the pre- and post-perturbation distributions $P(\cdot \mid X)$ and $P(\cdot \mid X')$:

$$\mathrm{TVD}(P, Q) = \frac{1}{2} \sum_i |P_i - Q_i|. \qquad (12)$$

Entropy-CCT is a minimal adaptation of CCT that preserves the same perturbation procedure and the same correlational test: we correlate the perturbation impact with whether the inserted token is mentioned in the NLE. The only change is the signal we correlate. Instead of TVD (probability-shift), we use Absolute Entropy Change (AEC), i.e., the change in predictive entropy, to directly target uncertainty (our object of explanation).

F.3 Statistical Testing Significance

We test statistical significance of r_{pb} using the standard Pearson correlation t-test. Each perturbation constitutes one observation: for each dataset we evaluate 600 instances with 12 perturbations per instance, yielding $n = 600 \times 12 = 7{,}200$ observations (App. F.1). Because the point-biserial correlation is algebraically equivalent to Pearson correlation between a continuous and a binary variable, we use the standard Pearson t-test for $H_0 : r_{\text{pb}} = 0$.

Interpreting r_{pb} and Δr_{pb}. The Entropy-CCT score is the point-biserial correlation (Tate, 1954) between the absolute entropy change $|\Delta u|$ and the binary mention flag m. Because it is mathematically identical to a Pearson r computed between one continuous and one binary variable, it obeys $-1 \leq r_{\text{pb}} \leq 1$. When $r_{\text{pb}} = 0$, it means the high- and low-impact perturbations are mentioned equally often. If the two strata are roughly balanced, every $+0.01$ in r_{pb} increases the probability that a truly uncertainty-influential token is mentioned by about one percentage point (pp). A *gain* Δr_{pb} therefore translates to an *absolute* improvement of $\approx |\Delta r_{\text{pb}}| \times 100$ pp in mention rate. For instance, moving from -0.08 to $+0.06$ is a swing of 0.14, corresponding to about 14 pp.

Significance testing. Because the point-biserial is a Pearson correlation, the familiar t–test applies:

$$t = r_{\text{pb}} \sqrt{\frac{n-2}{1-r_{\text{pb}}^2}}, \qquad (13)$$

$$t \sim t_{(n-2)} \qquad \text{under } H_0\colon r_{\text{pb}} = 0. \qquad (14)$$

With $n = 7{,}200$ we have $\mathrm{df} = 7{,}198$; the critical two-sided values are $|t| > 1.96$ for $p < 0.05$ and $|t| > 2.58$ for $p < 0.01$.

Faithfulness with significance results. Table 7 shows the point-biserial coefficients r_{pb}, which is our faithfulness measurement for model uncertainty(See, Eq.9), the associated t statistics, and two-sided p values for every model–method pair. Values that meet the stricter $p < 0.01$ criterion are highlighted in bold.

Across both datasets and all three backbones, the $\texttt{Prompt}_{\texttt{Baseline}}$ exhibits negative correlations, implying a *non-faithful* tendency to highlight low-impact tokens in the generated NLEs, with mean $= -0.094$. The prompt-only variant of our CLUE framework **CLUE-Span** neutralises this bias and turns the average into $+0.027$; two of its coefficients are clear $p < 0.01$, indicating a modest but significant improvement regarding faithfulness.

Model	Method	r_{pb}	t	p
	HealthVer			
Qwen2.5-14B-Instruct	**Prompt$_{\text{Baseline}}$**	-0.028	-2.38	1.7×10^{-2}
	CLUE-Span	$+0.006$	$+0.51$	6.1×10^{-1}
	CLUE-Span+Steering	$+0.033$	$+2.80$	$\mathbf{5.1 \times 10^{-3}}$
OLMo-2-1124-13B-Instruct	**Prompt$_{\text{Baseline}}$**	-0.100	-8.53	$\mathbf{< 10^{-15}}$
	CLUE-Span	$+0.005$	$+0.42$	6.7×10^{-1}
	CLUE-Span+Steering	$+0.020$	$+1.70$	9.0×10^{-2}
Gemma-2-9B-IT	**Prompt$_{\text{Baseline}}$**	-0.105	-8.96	$\mathbf{< 10^{-15}}$
	CLUE-Span	$+0.007$	$+0.59$	5.5×10^{-1}
	CLUE-Span+Steering	$+0.021$	$+1.78$	7.5×10^{-2}
	DRUID			
Qwen2.5-14B-Instruct	**Prompt$_{\text{Baseline}}$**	-0.080	-6.81	$\mathbf{9.8 \times 10^{-12}}$
	CLUE-Span	$+0.089$	$+7.58$	$\mathbf{3.4 \times 10^{-14}}$
	CLUE-Span+Steering	$+0.102$	$+8.70$	$\mathbf{< 10^{-15}}$
OLMo-2-1124-13B-Instruct	**Prompt$_{\text{Baseline}}$**	-0.130	-11.12	$\mathbf{< 10^{-15}}$
	CLUE-Span	$+0.014$	$+1.19$	2.3×10^{-1}
	CLUE-Span+Steering	$+0.099$	$+8.44$	$\mathbf{< 10^{-15}}$
Gemma-2-9B-IT	**Prompt$_{\text{Baseline}}$**	-0.120	-10.26	$\mathbf{< 10^{-15}}$
	CLUE-Span	$+0.043$	$+3.65$	$\mathbf{2.6 \times 10^{-4}}$
	CLUE-Span+Steering	$+0.098$	$+8.35$	$\mathbf{< 10^{-15}}$

Table 7: Detailed faithfulness evaluation results for baseline method **Prompt$_{\text{Baseline}}$**, and two variants of our CLUE framework **CLUE-Span** and **CLUE-Span+Steering** on Healthver and Druid dataset based on Qwen2.5-14B-Instruct(Qwen Team (2024)), OLMo-2-1124-13B-Instruct(Team OLMo et al. (2024))and Gemma-2-9B-IT(Gemma Team (2024)). Point-biserial correlation r_{pb} is our Entropy-CCT measurement(§4.1), along with t statistic and two-sided p-value for each model–method pair ($n = 7{,}200$, $df = 7{,}198$). Entries with $p < 0.01$ are bold.

The full **CLUE-Span+Steering** variant pushes the mean to $+0.062$ and achieves $p < 0.01$ in four of six settings. Interpreting these numbers using the discussion above, the switch from -0.094 to $+0.062$ yields an *absolute* increase of $(0.062 - (-0.094)) \times 100! \approx 16$, pp in the probability that a truly uncertainty-influential token is named in the NLE, which is easily noticeable in qualitative inspection.

The consistently positive, statistically significant gains therefore substantiate the claim made in the main text: CLUE produces markedly more faithful NLEs towards model uncertainty than the **Prompt$_{\text{Baseline}}$**, and the steer variant is particularly beneficial for models that initially struggle with uncertainty attribution.

G Prompt template for Prompt$_{\text{Baseline}}$, CLUE-Span and CLUE-Span+Steering on Healthver and Druid dataset

We designed two prompt templates for our experiments. The baseline prompt (Figure 4) gives the model no span interactions; instead, it must first identify the relevant agreements or conflicts and then discuss them in its explanation. In contrast, the prompt used by our CLUE framework (Figure 5) supplies the three pre-extracted span interactions (§2.3). The model is explicitly instructed to base its explanation on these spans, ensuring that the rationale remains grounded in the provided evidence.

G.1 Prompt template for Prompt$_{\text{Baseline}}$

To generate NLEs about model uncertainty without span-interaction guidance, we craft a three-shot prompt that instructs the model to identify the interactions most likely to affect its uncertainty and to explain how these relations they represent affect it. (See Figure 4).

G.2 Prompt template for CLUE-Span and CLUE-Span+Steering

To generate NLEs about model uncertainty with the span-interaction guidance, we craft a three-shot prompt that instructs the model to discuss how these interactions, along with the relations they represent, affect its uncertainty. (See Figure 5).

```
You are a helpful assistant. Your tasks:
1. Determine the relationship between the claim and
     the two evidence passages.
2. Explain your prediction's uncertainty by
     identifying the three most
   influential span interactions from Claim-Evidence
         1, Claim-Evidence 2,
   and Evidence 1-Evidence 2, and describing how
        each interaction's relation
   (agree, disagree, or unrelated) affects your
        overall confidence.
Return format: [Prediction] [Explanation]

### SHOT 1
Input
  Claim: [...]
  Evidence 1: [...]
  Evidence 2: [...]
Output
  [Prediction: ...] [Explanation: ...]

### SHOT 2   % omitted for brevity
### SHOT 3   % omitted for brevity

### NEW INSTANCE
Claim: {CLAIM}
Evidence 1: {E1}
Evidence 2: {E2}
Your answer:
```

Figure 4: Three-shot prompt for **Prompt$_{\text{Baseline}}$** (Shots 2–3 omitted) on the HealthVer and DRUID datasets.

Metrics	Prompt$_{\text{Base}}$	CLUE-S	CLUE-Steer
Faith. (↑)	0.08	0.10	**0.13**
Span-Cov. (↑)	–	0.25	**0.41**
Span-Ext. (↓)	–	**0.23**	0.27
LEE (↑)	0.80	**0.86**	**0.86**

Table 8: Automatic evaluation for three types of NLE on HealthVer using **Qwen2.5-14B-Instruct** when one claim and three pieces of evidence are involved: **Prompt$_{\text{Base}}$**(**Prompt$_{\text{Baseline}}$**) is the baseline method, **CLUE-S** (**CLUE-Span**) and **CLUE-Steer**(**CLUE-Span+Steering**) are the two variants of our CLUE framework.

H Extended Evaluation Results

H.1 Extended Evaluation Results for Scenarios Involving Three Evidence pieces

To verify the generalizability of our framework in scenarios involving multiple evidence pieces, we conduct an experiment on the Healthver dataset using the Qwen2.5-14B-Instruct model, pairing each claim with three pieces of evidence instead of two (as shown in Tab. 8). Note that we select the top-$k = 6$ span interactions for NLE generation, taking the top-1 interaction from each pairwise combination among the C, E_1, E_2, and E_3 input parts.

Overall, we observe a similar trend with three pieces of evidence as with two (see §4.5): **CLUE-Span+Steering** and **CLUE-Span** show better Faithfulness and Label–Explanation Entailment than **Prompt$_{\text{Baseline}}$**. Interestingly, the Faithfulness of all three explanation types increases relative to the two-evidence setting, from -0.028 to 0.08 for **Prompt$_{\text{Baseline}}$**, from 0.006 to 0.10 for **CLUE-Span**, and from 0.033 to 0.13 for **CLUE-Span+Steering**. This may be because, when NLEs are grounded in richer evidence, they capture more nuances in both claim–evidence and inter-evidence interactions, thereby more faithfully reflecting the model's fact-checking decision.

It is notable that **CLUE-Span+Steering** includes more targeted span interactions (higher Span-Coverage: 0.41 > 0.27) as well as more redundant span interactions (higher Span-Extraneous: 0.27 > 0.23) than **CLUE-Span**. This suggests that, although **CLUE-Span+Steering** better focuses on target interactions, it still suffers from redundancy due to the inclusion of more irrelevant interactions,

```
You are a helpful assistant. Your tasks:
1. Determine the relationship between the claim and
     the two evidence passages.
2. Explain your prediction's uncertainty by
     referring to the three span
   interactions provided below (Claim-Evidence 1,
        Claim-Evidence 2,
   Evidence 1-Evidence 2) and describing how each
        interaction's relation
   (agree, disagree, or unrelated) affects your
        overall confidence.
Return format: [Prediction] [Explanation]

### SHOT 1
Input:
  Claim: [...]
  Evidence 1: [...]
  Evidence 2: [...]
  Span interactions:
    1. ''[...]'' - ''[...]''  (C-E1)  relation:
         [...]
    2. ''[...]'' - ''[...]''  (C-E2)  relation:
         [...]
    3. ''[...]'' - ''[...]''  (E1-E2) relation:
         [...]
Output:
  [Prediction: ...] [Explanation: ...]

### SHOT 2   % omitted for brevity
### SHOT 3   % omitted for brevity

### NEW INSTANCE
Claim: {CLAIM}
Evidence 1: {E1}
Evidence 2: {E2}
Span interactions (pre-filled):
    1. ''{SPAN1-A}'' - ''{SPAN1-B}''  (C-E1)
         relation: {REL1}
    2. ''{SPAN2-A}'' - ''{SPAN2-B}''  (C-E2)
         relation: {REL2}
    3. ''{SPAN3-A}'' - ''{SPAN3-B}''  (E1-E2)
         relation: {REL3}
Your answer:
```

Figure 5: Three-shot prompt for **CLUE-Span** and **CLUE-Span+Steering** (Shots 2–3 omitted) on the HealthVer and DRUID datasets.

Metrics	$\texttt{Prompt}_{\texttt{Base}}$	CLUE-S	CLUE-Steer
Faith. (↑)	0.12	0.16	**0.18**
Span-Cov. (↑)	–	0.58	**0.67**
Span-Ext. (↓)	–	**0.59**	0.64
LEE (↑)	0.88	**0.90**	0.89

Table 9: Automatic evaluation for three types of NLE on HEALTHVER using **DeepSeek-R1-Distill-Qwen-14B** when one claim and two pieces of evidence are involved: $\texttt{Prompt}_{\texttt{Base}}$($\texttt{Prompt}_{\texttt{Baseline}}$) is the baseline method, **CLUE-S** (**CLUE-Span**) and **CLUE-Steer**(**CLUE-Span+Steering**) are the two variants of our CLUE framework.

which we call for future work to address.

H.2 Extended Evaluation Results on Reasoning Model

To demonstrate that our framework generalizes to reasoning models trained primarily via large-scale reinforcement learning, and designed to encourage chain-of-thought exploration, we adopt *DeepSeek-R1-Distill-Qwen-14B*[5] (DeepSeek-AI) as the base model and evaluate it on the HealthVer dataset in the setting with one claim and two pieces of evidence (see Tab. 9).

Overall, both variants of our CLUE model, **CLUE-Span** and **CLUE-Span+Steering**, show higher Faithfulness and Label-Explanation Entailment than the baseline method $\texttt{Prompt}_{\texttt{Baseline}}$, confirming the effectiveness of our framework. It is also notable that, compared with the evaluation results on other models in Tab. 1, all three types of explanations generated with the DeepSeek-R1-Distill-Qwen-14B model achieve the best performance across metrics, especially the highest Faithfulness and Span-Coverage. The former implies that reinforcement-learning-trained reasoning models may be more capable of generating highly faithful explanations with the thinking step; the latter suggests that this type of model follows our interaction-guided instructions more effectively. Consequently, both **CLUE-Span** and **CLUE-Span+Steering** include more target-span interactions, yielding higher Span-Coverage.

Comparing the two variants, **CLUE-Span+Steering** exhibits higher Span-Coverage (0.67 > 0.58) than **CLUE-Span**, but also more Span-Extraneous (0.64 > 0.59), suggesting that **CLUE-Span+Steering** tends to include redundant interactions while focusing on target interactions. **CLUE-Span+Steering** also achieves the highest Faithfulness (0.18 > 0.16), consistent with its higher Span-Coverage, but yields similar Label-Explanation Entailment to **CLUE-Span** (0.89 vs. 0.90), likely due to the increased Span-Extraneous. Addressing this trade-off is left for future work.

[5] https://huggingface.co/deepseek-ai/DeepSeek-R1-Distill-Qwen-14B

I Human Evaluation Details

I.1 Participants and Materials

Participants We recruited N=12 participants from Prolific, screened to be native English speakers from Australia, Canada, Ireland, New Zealand, the United Kingdom, and the United States. The study was approved by our institution's Research Ethics Committee (reference number anonymised).

Materials Forty claims (20 from DRUID, 20 from HealthVer) were selected at random for evaluation, while 120 explanations were evaluated in total. For each instance, participants were provided with a claim, two evidence documents, model verdict, model numerical certainty, and three alternative explanations (see Figure 6 in I.6). The explanations presented to participants were those generated using Qwen2.5-14b-instruct (Qwen Team, 2024) based on its automatic evaluation performance.

Each participant evaluated explanations for 10 instances (5 labelled 'True', 5 labelled 'False'), in addition to two attention check instances which were used to screen responses for quality. To mitigate the difficulty and cognitive demands of the task, we kept the number of instances per participant low, and removed participants who failed attention checks from the data. In order to minimise the risk of bias, each explanation presented to participants was assigned a neutral label: 'Explanation A', 'Explanation B', 'Explanation C', so that participants could not infer anything about how the explanations were generated or indicate that one should be favoured over another. Each explanation was also presented in a different, random order for each instance (e.g., ABC for Claim 1, CAB for Claim 2, etc.) to mitigate order effects such as primacy bias or recency bias. The order in which the response options 'Explanation A', 'Explanation B', 'Explanation C' also appeared in the same random order.

Procedure Participants read information about the study (see I.3) and provided informed consent

	DRUID		HealthVer	
	Set A	Set B	Set A	Set B
Helpfulness	.016	.079	.003	.013
Consistency	.44	.058	.017	.016
Non-redundancy	.005	.084	.005	.019
Coverage	.494	.113	.018	.027
Overall Quality	.005	.158	.01	.002

Table 10: Interrater agreement (Kendall's W) for human evaluation

(see I.4) before reading detailed task instructions and completing a practice example of the task (see I.5). Participants then progressed through the study at their own pace. The task took approximately 20 minutes, and participants were paid £3 for their work.

I.2 Human Evaluation Results

I.2.1 Interrater agreement

In line with similar NLE evaluations carried out by previous studies (e.g., (Atanasova et al., 2020)), interrater agreement (Kendall's W (Kendall and Smith, 1939)) was moderate to low (see Table 10). We attribute this to the relative complexity of the task and individual differences in how the information was perceived.

I.2.2 CLUE Variant preferences

Table 11 shows the mean rank assigned by participants to each explanation variant. The explanations generated by CLUE were preferred by our participants to those generated using **Prompt$_{\text{Baseline}}$**: the **explanations generated by CLUE-Span+Steering were rated as most helpful, highest coverage, and containing the least amount of redundant information**, while **those from CLUE-Span were judged to have the highest consistency and overall quality**.

Finally, we observed slight variation between datasets: **CLUE-Span+Steering** tended to be rated higher than **CLUE-Span** for DRUID, and vice versa for HealthVer. This may arise from differences in length and complexity of the input: DRUID evidence documents, retrieved from heterogeneous online sources and often consisting of longer form new articles, may have benefited from attention steering more than HealthVer evidence documents which consist of focused, shorter extracts from scientific abstracts.

	Prompt$_{\text{Base}}$	CLUE-S	CLUE-Steer
Helpfulness			
Overall	2.025	1.892	**1.867**
DRUID	1.9	1.917	**1.767**
HealthVer	2.15	**1.867**	1.967
Consistency			
Overall	1.875	**1.783**	1.817
DRUID	1.717	1.75	**1.617**
HealthVer	2.033	**1.817**	2.017
Non-redundancy			
Overall	2.05	1.908	**1.833**
DRUID	1.983	1.983	**1.683**
HealthVer	2.117	**1.833**	1.983
Coverage			
Overall	1.967	1.775	**1.758**
DRUID	1.767	1.75	**1.617**
HealthVer	2.167	**1.8**	1.9
Overall Quality			
Overall	1.967	**1.908**	1.925
DRUID	1.9	1.9	**1.817**
HealthVer	2.033	**1.917**	2.033

Table 11: Mean Average Rank (MAR) for the five human-evaluation criteria applied to explanations from **Qwen2.5-14B-Instruct** on the HEALTHVER and DRUID datasets (chosen for its high faithfulness; see §4.5). **Prompt$_{\text{Baseline}}$**, **CLUE-Span** (**CLUE-S**), and **CLUE-Span+Steering** (**CLUE-Steer**) are compared. Lower MAR means a better (higher) average rank; the best score in each row is boldfaced.

I.3 Human Evaluation Information Screen

Thank you for volunteering to participate in this study! Before you decide whether you wish to take part, please read this information screen carefully.

1. What is the project about?

Our goal is to make sure that AI fact-checking systems can explain the decisions they produce in ways that are understandable and useful to people. This survey is part of a project to help us understand [anonymised]

2. What does participation entail?

You are invited to help us explore what kinds of explanations work better in fact-checking. In this task you will see claims, an AI system's prediction about whether this claim is true or false and corresponding evidence used to make the prediction. You will also see an explanation for why the AI system is certain or uncertain about its prediction to help you decide how to interpret the true/false prediction. We ask you to evaluate the explanations along 5 different dimensions (the detailed explana-

tion of the task is on the next page). All participants who complete the survey will receive a payment of £3. There is no cost to you for participating. You may refuse to participate or discontinue your involvement at any time without penalty.

3. Source of funding

This project has received funding from [anonymised]

4. Consenting to participate in the project and withdrawing from the research

You can consent to participating in this study by ticking the box on the next page of the study. Participation in the study is completely voluntary. Your decision not to consent will have no adverse consequences. Should you wish to withdraw during the experiment you can simply quit the webpage. All incomplete responses will be deleted. After you have completed the study and submitted your responses, it will no longer be possible to withdraw from the study, as your data will not be identifiable and able to linked to you.

5. Possible benefits and risks to participants

By participating in this study you will be contributing to research related to understanding what kinds of explanations are useful to people who use or who are impacted by automated fact checking systems. This is a long-term research project, so the benefits of the research may not be seen for several years. It is not expected that taking part will cause any risk, inconvenience or discomfort to you or others.

6. What personal data does the project process?

The project does not process any personal data.

7. Participants' rights under the General Data Protection Regulation (GDPR)

As a participant in a research project, you have a number of rights under the GDPR. Your rights are specified in [anonymised]

8. Person responsible for storing and processing of data

[anonymised]

Please click 'Next' to read more about consenting to participate in the study.

I.4 Human Evaluation Consent Form

We hereby request your consent for processing your data. We do so in compliance with the General Data Protection Regulation (GDPR). See the information sheet on the previous screen for more details about the project and the processing of your data.

- I confirm that I have read the information sheet and that this forms the basis on which I consent to the processing of my data by the project.
- I hereby give my consent that [anonymised] may register and process my data
- I understand that any data I provide will be anonymous and not identifiable to me.
- I understand that my anonymous response data will be retained by the study team.
- I understand that after I submit my responses at the end of the study, they cannot be destroyed, withdrawn, or recalled, because they cannot be linked with me.
- I understand that there are no direct benefits to me from participating in this study
- I understand that anonymous data shared through publications or presentations will be accessible to researchers and members of the public anywhere in the world, not just the EU.
- I give my consent that the anonymous data I provided may be stored in a database for new research projects after the end of this project.
- I give permission for my anonymous data to be stored for possible future research related to the current study without further consent being required.
- I understand I will not be paid for any future use of my data or products derived from it.

By checking this box, I confirm that I agree to the above and consent to take part in this study.

☐ I consent

I.5 Evaluation Task Instructions

What do I have to do?

In this study you will see claims, an AI system's prediction about whether this claim is true or false, how certain the system is about its label, and the corresponding evidence used to make the prediction. You will also see three different explanations for why the AI system is certain or uncertain about its prediction. These explanations are intended to help you decide how to interpret the true/false prediction.

Your task is to **evaluate the quality of the explanations** provided, **not** the credibility of the claims and evidence.

What information will I be shown?
You will be shown examples of claims, evidence document, verdicts and explanations.

- A claim is some statement about the world. It may be true, false, or somewhere in between.
- Additional information is typically necessary to verify the truthfulness of a claim - this is referred to as evidence or evidence document. An evidence document consists of one or several sentences extracted from an external source for the particular claim. In this study, you will see two evidence documents that have been retrieved for a claim. These evidence documents may or may not agree with each other.
- Based on the available evidence, a verdict is reached regarding whether a claim is true or false.
- Uncertainty often arises when evaluating the claim and evidence to reach a verdict. Each verdict is accompanied by a numerical uncertainty score which represents the AI system's confidence that its predicted verdict is correct.
- You will see 3 alternative explanations for where uncertainty arises with regard to the verdict. Note that these explanations focus on the AI system's uncertainty, not the verdict itself.
- You are asked to evaluate the explanations according to 5 different properties. The properties are as follows:

  Helpfulness. The explanation contains information that is helpful for evaluating the claim and the fact check.

  Coverage. The explanation contains important, salient information and does not miss any important points that contribute to the fact check.

  Non-redundancy. The explanation does not contain any information that is redundant/repeated/not relevant to the claim and the fact check.

  Consistency. The explanation does not contain any pieces of information that are contradictory to the claim and the fact check.

  Overall Quality. Rank the explanations by their overall quality.

- Please rank the explanations in descending order. For example, you should rank the explanation that you think is most helpful as '1', and the explanation that you think is least helpful as '3'. If two explanations appear almost identical, you can assign them the same ranking, but as a general rule, you should try rank them in hierarchical order.
- The three explanations, Explanation A, Explanation B, and Explanation C, will appear in a different order throughout the study, so you may need to pay some attention to which is which.

Important: Please only consider the provided information (claim, evidence documents, and explanations) when evaluating explanations. Sometimes you will be familiar with the claim, but we ask you to approach each claim as new, whether or not you have seen it before. It doesn't matter whether you personally agree or disagree with the claim or evidence – we are asking you to evaluate what the AI produces: if you were to see this claim for the first time, would you find the explanation provided by the AI useful? On the next page, you will see an example of the task.

I.6 Example of human evaluation set-up

Here is an example of what you will see during the study. First, you will see a **Claim**, and two pieces of **Evidence**, along with an AI system's predicted **Verdict** and the system's **Certainty** that its prediction is correct.

The **parts of the claim and evidence that are most important to the AI system's certainty are highlighted.** Parts of the Claim are Red, parts of Evidence 1 are Blue, and parts of Evidence 2 are Green.

Underneath, you will see **three alternative explanations for the AI system's certainty**, Explanation A, Explanation B, and Explanation C. The parts of each explanation that refer to the claim and evidence are colour coded in the same way (Claim = Red, Evidence 1 = Blue, Evidence 3 = Green).

Your task is to read the claim, evidence, and explanations, and rank each explanation based on five properties.

Now, you can try this example below!

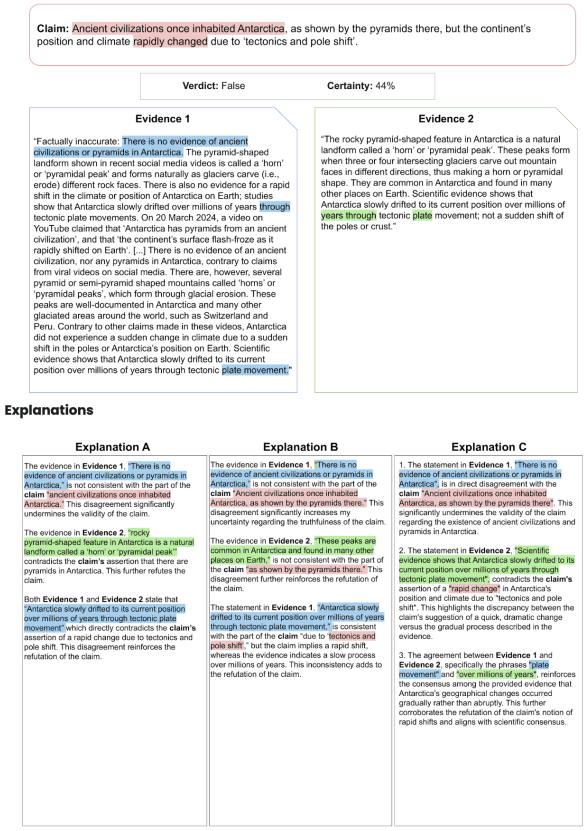

Figure 6: Example of human evaluation set-up. Explanation A was generated using $\text{Prompt}_{\text{Baseline}}$, Explanation B by CLUE-Span, and Explanation C by CLUE-Span+Steering