IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 22 June 2026, accepted 19 July 2026, date of publication 31 July 2026, date of current version 24 August 2026.

Digital Object Identifier 10.1109/ACCESS.2026.3719082

APPLIED RESEARCH

Explaining the Role of Evidence in Data-Driven Fact-Checking

JEAN-FLAVIEN BUSSOTTI[1], ANDREA BARALDI[2], FRANCESCO GUERRA[2], AND PAOLO PAPOTTI[3]

[1]Megagon Labs, Mountain View, CA 94041, USA
[2]DIEF, Universitá di Modena e Reggio Emilia, 41121 Modena, Italy
[3]EURECOM, 06410 Biot, France

Corresponding author: Paolo Papotti (Paolo.Papotti@eurecom.fr)

This work was supported in part by the ANR Project ATTENTION under Grant ANR-21-CE23-0037, and in part by the 3IA Côte d'Azur Investments in the IA-Cluster Project under Grant ANR-23-IACL-0001.

ABSTRACT Fact-checking is crucial for combating misinformation, and computational methods are essential for scalability. The most effective approaches leverage neural models that use domain-specific evidence to validate claims. However, these models often act as black boxes, providing labels without explaining the rationale behind their decisions. In this work, we introduce a comprehensive evaluation protocol to assess how fact-checking verifiers attribute decisions to individual evidence pieces. We evaluate whether existing explainable AI methods, such as LIME and SHAP, can be adapted to perform evidence-level attribution and to classify evidence relevance across four established datasets. Our findings show that post-hoc attribution methods can support the analysis of how verifier predictions change under evidence perturbations, thereby improving transparency by highlighting patterns and potential issues in model behavior. This was achieved through the evaluation of five fact-checking systems, showing that for verifiers amenable to perturbation-based analysis, post-hoc attribution methods often provide higher-quality evidence-level signals than native explainers.

INDEX TERMS Explainable artificial intelligence, fact checking, NLP.

I. INTRODUCTION

Online misinformation endangers democracy and trust in news sources. While fact-checking mitigates this, human-based methods are resource-intensive and cannot scale with the rapid content influx on social media [1]. Automated methods have been proposed to address this scalability challenge, offering cost-effective alternatives. Computational fact-checking involves establishing the relationship between a textual claim and a body of evidence to determine if the evidence supports, refutes, or is insufficient to verify it [2], [3]. Evidence can be structured tabular data, unstructured text, or their combination [1].

Evidence-based fact-checking typically relies on a two-stage ML pipeline: given a claim, a retriever identifies relevant evidence from a document corpus, and a verifier evaluates the claim's veracity using that evidence. Correctly labeling the claim is crucial, but inference alone is insufficient: most existing solutions operate as black boxes, providing accurate predictions without explaining which

Claim	The Eagle AC-7 Eagle 1 is a military aircraft manufactured by Windecker Industries.
Evidence	The Eagle AC-7 Eagle 1 (USAF designation YE-5) is an aircraft. Windecker Industries was an American aircraft manufacturer founded in 1962. [...] It was manufactured by Windecker Industries. Wei Hang holds the rights and the type certificate to produce the military aircraft.
Label	SUPPORTS

FIGURE 1. Example from the Feverous dataset with a claim, its retrieved evidence and the corresponding ground truth label. Our work labels the supporting evidence in green, noise in gray, and refuting in red.

The associate editor coordinating the review of this manuscript and approving it for publication was Massimo Cafaro.

evidence drove their decisions. Since fact-checkers discard outputs that do not clearly expose the supporting evidence [4], transparency is essential for users to verify and assess the model's conclusions [1].

Our claim is that it is not only crucial to retrieve relevant evidence, as re-rankers do in the retrieval stage [5], but also essential to determine how individual evidence pieces influence the verifier's classification. The idea is to have systems able to label evidence in a manner analogous to the colored annotations in Figure 1: supporting sentences are highlighted in green, refuting evidence in red (e.g., a sentence stating that production rights are held by *Wei Hang*, not *Windecker Industries*), and noisy or irrelevant content in gray. By doing so, we can identify which pieces of evidence led the model to lean toward a specific claim label, supporting, refuting, or indicating insufficient information, providing an interpretable explanation of the verifier's reasoning.

In this work, we investigate whether existing post-hoc explanation methods are capable of clarifying how fact-checking models use individual pieces of evidence to support or refute claims. Rather than proposing a new framework, our goal is to assess the extent to which current systems provide meaningful evidence-level explanations and to determine whether the domain requires further methodological developments. To this end, we enable users to analyze the outputs of a fact-checking tool by deriving explanations for individual evidence pieces, which allow us to address the following three questions:

Q1: What are the evidence pieces that steer the decision? Can the verifier discern noise, identifying the most important evidence in claim evaluation?

Q2: To what extent do the verifiers rely on the evidence and the claim to make their decisions? Does this reliance vary across claim labels?

Q3: How is the impact of different pieces of evidence on the verifier distributed? Do verifiers use all the available evidence?

Our work establishes a comprehensive benchmark for assessing the explainability of current verifiers, setting expected standards for evidence attribution and serving as a reference point for developing new models. Beyond benchmarking, we provide actionable directions for improving verifiers: assessing whether models genuinely attend to supporting or refuting evidence, distinguishing reliance on supplied evidence versus internal knowledge, and evaluating how they handle cases labeled Not Enough Information by recognizing the absence of evidence. These insights directly guide the refinement of more transparent and reliable verification systems.

This work explores these questions using two post-hoc explanation methods [6] (LIME [7] and SHAP [8], in Section III) over four popular datasets (Feverous, SciFact, FM2 and AVTC, introduced in Section IV-A) and five verifier models, four of which include a ''native explainer'' providing explanations for their predictions (RoBERTa, GFCE and an LLM-prompting verifier based on LLaMA3.1, GPT-4.1 Mini and Claude 3.5 Sonnet described in Section IV-B). Our experimental evaluation demonstrates that post-hoc methods enable the analysis of a broad class of verifiers, including those based on fine-tuned PLMs, providing evidence-based justifications of a quality comparable to those produced by native explainers (Section V).

II. RELATED WORK

Automated fact-checking has been investigated for many years, and survey papers consistently characterize the dominant architecture as a three-stage pipeline with claim detection component coupled with an evidence retriever followed by a verifier that predicts claim veracity or a rationale-backed label (retriever–verifier) [1], [4]. Only some existing approaches include *explanation functionalities*, i.e., they provide reasons for their predictions [9]. Several of these approaches integrate a native explainer component within their verification models.

Some of these works leverage generative methods, such as summarization, to produce explanations [10], [11]. These approaches face the challenge of selecting an appropriate level of detail to be persuasive [12]. Related methods use question answering as a proxy for explainable fact-checking [13]. However, generative explanations may reduce trustworthiness, as they do not directly expose original evidence sources. Other works [14], [15] adopt a dual-transformer architecture, where a verifier and a dedicated explainer are jointly trained. The explainer operates at the evidence or token level by assigning labels to the input, rather than producing free-text explanations.

Other systems decide each piece of evidence's usefulness before making a claim label prediction [16], [17] or rely on the training of an explainer [18], [19]. However, in these approaches the explanation mechanism is not directly derived from the verifier's decision process, as the explainer is architecturally decoupled from the predictor. As a consequence, they cannot guarantee that the verifier actually relies on the evidence selected by the explainer, nor provide insights into the role played by each evidence piece in the final decision. Furthermore, since these explainers are tightly integrated into specific model architectures, they cannot be readily applied to arbitrary pre-existing verifiers.

In addition to these approaches, other paradigms have been explored for explainable fact-checking. Large Language Models (LLMs) have been used to analyze the relevance and impact of evidence pieces in a multimodal context, including text and tables [20], [21], [22]. For instance, the Win Rate approach estimates which paragraphs or table cells are most influential in driving a model's decision, providing insights into evidence importance even when LLMs struggle to verbalize their reasoning. Decision Tree-based approaches [23] have also been employed to determine which user comments or textual inputs are necessary for verifying a claim, using semantic similarity and metadata features to both take and explain the decision. Knowledge

Graph-based methods [24] exploit structured information to infer claim veracity, though they typically provide rule-based reasoning without highlighting which specific pieces of evidence are most decisive, in contrast with the Win Rate method.

Post-hoc explainers represent an alternative to native explainers. Simons [25] compare different post-hoc analysis methods on [14]'s verifier. Similarly, [26] proposes a full pipeline with fact extraction, involving post-hoc analysis to determine which tokens guided the decision. However, due to the analysis being conducted at the token level their approach cannot be used to evaluate "black-box" verifier explainers, such as those based on RoBERTa. In contrast, we focus on explaining black-box verifiers, such as popular verification models based on pre-trained language models, which have competitive performance and the best trade-off when considering cost/energy issues [27]. Recent work has investigated the faithfulness of explanations produced by large language models, highlighting that plausible natural-language explanations may not accurately reflect the model's internal reasoning process (e.g., [28], [29]). Our work complements this line of research by analyzing evidence-level attribution in fact-checking systems using model-agnostic post-hoc explainers.

In this paper, we employ several methods: three LLM-based, one lexical-similarity-based, and two local post-hoc attribution ones, LIME and SHAP, which explain a prediction by quantifying the contribution of different parts of the input—in our case, individual evidence pieces—to the model's decision. Both methods are model-agnostic and probe the verifier through input perturbations, making them suitable for black-box settings. LIME randomly removes features of an input example and trains a linear interpretable surrogate model to predict the model's output for those perturbed instances in the local neighborhood of the original input. SHAP attributes the contribution of each feature to the model's prediction by estimating Shapley values over multiple feature coalitions, following principles from cooperative game theory. It should be noted that gradient-based attribution methods, such as Integrated Gradients (IG) [30] and Layerwise Relevance Propagation (LRP) [31], could be used as alternatives to SHAP or LIME. In Appendix A, we report a comparison between these approaches. However, gradient-based methods require access to internal model representations and are therefore not applicable to heterogeneous architectures or closed-source LLM endpoints, which are central to our benchmark. For this reason, we adopt SHAP and LIME in our main experiments.

III. THE FRAMEWORK

The Fact-checking Problem. We define the evidence set $\mathbf{e} = \{s_1, \ldots, s_n\}$ as a non-empty set of evidence pieces, where each s_i is either a sentence from the document or the content of a table cell. The value n denotes the total number of available sentences and cells in the document.

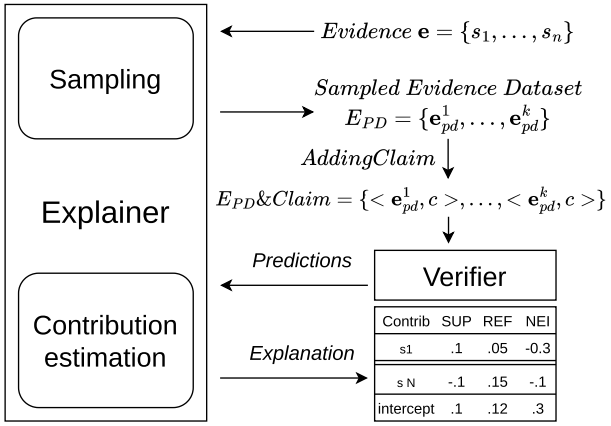

FIGURE 2. The explanation framework.

Algorithm 1 Evidence Contribution Estimation

Require: Claim c, evidence set $e = \{s_1, \ldots, s_n\}$, verifier $\mathcal{F}$, explainer $\mathcal{X}$ (LIME or SHAP)

Ensure: Evidence contribution scores $\{\phi(s_1), \ldots, \phi(s_n)\}$ and intercept ϕ_0

1: $\mathcal{X}$ generates a set of sampled evidence subsets E_{PD} by sampling different subsets of evidence pieces from e, such that each sampled set $e^i_{pd} \subseteq e$ ▷ Perturbations for LIME, feature coalitions for SHAP
2: **for** each sampled evidence set $e^i_{pd} \in E_{PD}$ **do**
3: Pair e^i_{pd} with the original claim c
4: Compute prediction $y^{(i)} = \mathcal{F}(e^i_{pd}, c)$
5: **end for**
6: Use explainer $\mathcal{X}$ to estimate evidence contributions $\phi(s_j)$ for each $s_j \in e$, and the intercept ϕ_0
7: **return** Evidence contribution scores and intercept

A supervised verifier model $\mathcal{F}$ assesses whether a textual claim c finds support or contradiction w.r.t. a given evidence $\mathbf{e}$. Formally, $\mathcal{F}$ takes as input a data pair $<\mathbf{e}, c>$ and outputs a verdict in the space $\mathcal{L}$. In Figure 1 we show an example of evidence $\mathbf{e}$, claim c, and label. In this work, we consider two settings. In the binary (2-label) setting $\mathcal{L} = \{Supports, Refutes\}$, while in the full (3-label) setting $\mathcal{L} = \{Supports, Refutes, Not\ Enough\ Information\}$. The *Not Enough Information* (NEI) label is for cases where there is not enough evidence to state if the claim is supported or refuted.

Explanation in Fact-checking. Given a claim c and an associated evidence set $\mathbf{e}$, an explanation is defined as a function that assigns a contribution score to each evidence piece $s_i \in \mathbf{e}$, reflecting its impact on the verifier's prediction for c.

Our Proposal.[1] Adapting local post-hoc explainers to the fact-checking setting raises two main challenges. First, standard explainers such as LIME and SHAP assume as input a single instance represented by a set of features

[1]The code is available on the project GitHub https://github.com/softlab-unimore/explainable-fact-checking.git

and associated with a model prediction. In fact-checking, however, the verifier operates on structured inputs composed of a textual claim and a set of evidence pieces. This mismatch requires defining how claims and evidence should be represented within the explainer framework, as well as how the evidence set can be mapped to interpretable features. Second, local explainers typically include an intercept term, which captures the portion of the prediction that is not explained by explicit feature contributions. In the fact-checking setting, the semantic interpretation of the intercept is non-trivial and must therefore be carefully defined.

Figure 2 presents our framework, which addresses these challenges by adapting the input representation used by local post-hoc explainers, such as LIME and SHAP, to the fact-checking setting. The overall procedure is formalized in Algorithm 1. Given an input example $\langle \mathbf{e}, c \rangle$, the explainer samples different subsets of evidence pieces to generate alternative evidence inputs $\mathbf{e}_s \subseteq \mathbf{e}$. These sampled subsets correspond to input perturbations in the case of LIME and to feature coalitions in the case of SHAP. Each sampled evidence subset $\mathbf{e}_s$ is then paired with the original claim and provided as input to the verifier. Pairing each sampled evidence subset with the original claim is essential to preserve the semantic context of the verification task, ensuring that the contribution of each evidence piece is evaluated with respect to the specific claim it is intended to support or contradict.

The explainer uses the verifier's predictions on the sampled examples to estimate a contribution score for each evidence piece, treated as an individual feature, as well as an intercept term. The intercept captures the portion of the verifier's prediction that is not explained by the explicit contributions of the evidence pieces. Since the claim is kept fixed across all sampled inputs, the intercept captures the part of the local approximation that is not assigned to explicit evidence-piece contributions. We refer to this term as a claim/baseline component: it is influenced by the fixed claim, but it may also absorb model priors, default predictions, and systematic behavior unrelated to individual evidence pieces. Therefore, it should not be interpreted as a purely causal contribution of the claim. Accordingly, the verifier's prediction can be approximated as a function of the contributions defined by the explanation:

$$\text{Pred} \approx \sum_{i=1}^{N} \text{contribution}(e_i) + \text{Intercept}. \tag{1}$$

IV. DATASETS AND MODELS

A. DATASETS

We select four fact-checking datasets. Each example consists of a claim written by a human, a label, and the golden evidence used to classify the claim. We refer to the ground-truth evidence manually annotated in the fact-checking dataset for a given claim as *gold evidence*. Such evidence represents the clean set of relevant evidence required to deduce $\mathcal{L}$. To evaluate model behavior under realistic conditions, we additionally construct controlled noisy versions of each dataset by augmenting gold evidence with non-relevant evidence, hereafter referred to as *noisy* evidence. Statistics about the datasets after pre-processing are reported in Table 1, while retrieval quality and the resulting noise characteristics are summarized in Table 2. When the original test set is private, we use the original validation split as the fixed evaluation set, ensuring that all reported metrics are computed on the same deterministic test data.

TABLE 1. Statistics of datasets used for training and testing the fact-checking models. The claim counts correspond to the number of *'Supports'*, *'NEI'*, and *'Refutes'*.

Dataset	#Claim	T.	|Claim|	#Ev.	#Noise
Fev.3L	71.2k(27.2k/2.2k/41.8k)	train	25.3	1.6	16.8
	7.9k(3.5k/0.5k/3.9k)	test	24.9	1.4	17.6
Fev.2L	69k(27.2k/0/41.8k)	train	25.3	1.6	16.8
	7.4k(3.5k/0/3.9k)	test	24.9	1.4	17.5
SciFact	0.8k(0.2k/0.3k/0.3k)	train	12.3	1.3	9.0
	0.3k(0.1k/0.1k/0.1k)	test	12.5	1.3	8.7
AVTC	2.8k(1.7k/0.3k/0.8k)	train	17.1	2.6	5.6
	0.5k(0.3k/0.04k/0.1k)	test	14.4	2.6	5.5
FM2	10.4k(5.3k/0/5.1k)	train	13.7	1.3	9.1
	1.4k(0.7k/0/0.7k)	test	13.8	1.3	9.1

FEVERous [32] is an extension of FEVER [33] featuring complex claims derived from both textual and tabular Wikipedia evidence. To handle tabular content, we linearize tabular evidence rather than providing full tables, using the format: $Cell_{Value}$ <context>$Cell_{Header}$</context>. We consider two dataset variants, with and without the *'Not Enough Information'* label, denoted as **Fev.3L** and **Fev.2L**. We construct a controlled noisy version of the dataset by applying the provided retriever to both training and test sets, retrieving five documents per claim. From each document, we select five sentences and three tables, with an arbitrary number of cells per table. Gold evidence is not always retrieved (0.36 Recall), and irrelevant evidence is included (0.07 Precision), potentially leaving some claims unverifiable.

SciFact [34] contains expert-written claims paired with evidence from scientific papers, consisting exclusively of textual sources. We create a controlled noisy version of the dataset by appending up to 20 sentences sampled from the five provided distractor abstracts to the gold evidence.

FM2 [35] is obtained from an online multiplayer game in which users create claims from a list of Wikipedia evidence, aiming to make them difficult to fact-check by other players. Evidence is textual only, and the *'Not Enough Information'* label is not present. We construct a controlled noisy version by treating the sentences not used to form the claim as non-relevant evidence and explicitly adding them to the example.

AVeriTeC (AVTC) [36] contains real-world claims verified using Web evidence in the form of question–answer pairs supported by online content. We treat each question–answer pair as an individual textual evidence. We construct a controlled noisy version of the dataset using the provided

TABLE 2. Performance of the retrievers used to build the datasets (Precision, Recall, F1). A dataset is *Sufficient* if, for each claim, it contains all gold-standard evidence, i.e., the dataset contains the original evidence along with additional evidence that act as noise. Precision and recall are computed based on the retrieval of the top 5 documents for FEVEROUS, or for the other datasets, based on the precision and recall that a retriever would achieve when retrieving all gold evidence along with the distractor evidence, as described in the text.

Dataset	Precision	Recall	F1	Sufficient
Fev. 2L	0.07	0.36	0.12	✗
Fev. 3L	0.07	0.36	0.12	✗
SciFact	0.13	1	0.23	✓
AVeriTeC	0.32	1	0.48	✓
FM2	0.12	1	0.22	✓

question–answer generator: from the generated candidates, we select the least relevant pairs according to BM25 similarity with the claim, adding on average 5.5 noisy pairs per example. The additional label Conflicting Evidence/Cherry-picking is excluded because it is not predicted by any of the evaluated verifiers. This exclusion affects a small but non-negligible portion of the original AVeriTeC data: 233 examples in the considered train/development split, corresponding to 6.53% overall, with 6.36% in training and 7.60% in development. As a consequence, our AVeriTeC evaluation focuses on the common Supports/Refutes/Not Enough Information setting and may underrepresent mixed-evidence cases in which both supporting and refuting evidence are present.

B. MODELS AND THEIR TRAINING

We evaluate two categories of approaches: models fine-tuned within our experimental framework and pre-trained models used as baselines. The baselines include pre-trained large language models (both locally deployed and API-accessed) and a TF–IDF baseline that scores evidence by lexical similarity to the claim.

Fine-tuned Models. We consider two verifiers that are trained or fine-tuned within our experimental framework: RoBERTa and GFCE.

RoBERTa [37] is a transformer-based language model. We fine-tune the model on relevant NLI datasets [38] as in FEVERous, using a learning rate of $1e^{-5}$ for datasets with three labels and $1e^{-7}$ for datasets with two labels. We train the model for one epoch on FEVERous and AVeriTeC, and for three epochs on FM2 and SciFact.

GFCE [15] jointly trains veracity prediction and explanation generation using a fine-tuned version of DistilBERT. It leverages both claim texts and supporting evidence to produce justifications. We use a learning rate of $1e^{-5}$ for all datasets and train the model for three epochs on FM2, FEVERous, and SciFact, and for two epochs on AVeriTeC. Since the original work does not fully specify training details for our experimental setting, we adopt dataset-specific calibration strategies. In particular, for AVeriTeC we mitigate class imbalance at training time by equalizing the number of *Supported* and *Refuted* claims.

> You are an expert fact-checker. I will provide you a claim and a list of evidence. Based on them, you should answer in two steps.
> In the first step, you should repeat every evidence that is useful to predict a label for the claim. Repeat only those evidence pieces, one evidence piece per line starting with their index, as they are presented to you. Start this step by the text: `Useful evidence:`
> The second step should contain your predicted label. The label `[PREDICTED LABEL]` can be `$LABELS_TAG$`. Answer even if you are not sure. The format of the second step should be `Label:[PREDICTED LABEL]`

FIGURE 3. Prompt template used for all evaluated LLMs. `$LABELS_TAG$` **is a placeholder for the possible labels of the given dataset.**

Pre-trained LLM. In addition to models fine-tuned within our framework, we evaluate pre-trained large language models used in a zero-shot prompting setting. These models differ in deployment mode: LLaMA is locally deployed, while other LLMs are accessed via API.

LLaMA. We evaluate a locally deployed LLaMA-3.1-70B model used as a verifier through prompting. The prompt includes the task description, the claim, and its associated evidence, and the model outputs a veracity label together with the evidence supporting the decision.

API-based LLMs. We evaluate large language models accessed via API and used in a zero-shot prompting setting. The prompt includes the task description, the claim, and its associated evidence, and the model outputs a veracity label together with the evidence supporting the decision. This joint formulation allows us to directly interpret the generated evidence as an implicit evidence usefulness classification. The prompt template used for all LLMs is shown in Figure 3. For comparison with attribution-based explanations, we consider explanations directly generated by GPT-4.1-Mini [39] and Claude 3.5-Sonnet [40].

TF–IDF Baseline. To estimate the intrinsic difficulty of evidence usefulness identification, we include a TF–IDF baseline that scores evidence based on lexical similarity with the claim. As with other attribution methods, its decision threshold is calibrated using the procedure described in Appendix B.

C. CONFIGURATIONS OF THE EXPLAINERS

We use LIME and SHAP to analyze the verifier outputs. For the GFCE and RoBERTa models, we set 500 perturbation samples per explanation, while for LLaMA, we reduced it to 100 because of its computational cost. This reduced budget is a computational trade-off required by the cost of repeatedly querying LLaMA 3.1 70B. Consequently, LLaMA attribution results should be interpreted primarily as qualitative diagnostic evidence rather than as precise estimates directly comparable in variance to the 500-sample

RoBERTa and GFCE settings. We explain a subset of the datasets: we exclude examples exceeding 512 tokens (too long for RoBERTa) and sample up to 1,000 examples from each dataset. Since GPT-4.1-Mini and Claude 3.5-Sonnet are accessed via API, the large-scale perturbation required by SHAP and LIME is impractical; therefore, we do not compute post-hoc attribution scores for them.

V. EXPERIMENTS

The experimental evaluation addresses three main questions:

- **Q1: SHAP and LIME based Noise Detection.** We investigate how SHAP- and LIME-based explanations distinguish noisy from useful evidence (Section V-B).
- **Q2: Model Reliance on Evidence and Claim.** We analyze how verifiers rely on the evidence and the claim when predicting a label (Section V-C).
- **Q3: Contribution Spread on Evidence.** We study the importance given by verifiers to the evidence for each label individually (Section V-D).

In addition, although this work primarily focuses on the insights provided by SHAP and LIME rather than on production-oriented deployment considerations, we conduct an efficiency analysis of the considered explanation methods, reported in Appendix C.

A. EXPERIMENTAL SETUP

The experiments are performed on a system running Ubuntu 20.04.6 LTS with a kernel version of 5.4.0-177-generic. The hardware configuration consists of an AMD EPYC 7272 12-Core Processor with 128GB of RAM, and a NVIDIA GeForce RTX 3090 GPU with 24GB of memory.

The software setup includes Python 3.8.10, LIME 0.2.0.1, SHAP 0.45.0. To perform inference on LLaMA 3.1 70B, we used a more powerful server running Ubuntu 20.04.6 LTS with a kernel version of 5.15.0-122-generic. The hardware configuration of this server consists of an AMD EPYC 7742 64-Core Processor with 512GB of RAM, and a NVIDIA A100-SXM with 80GB of memory. The software setup includes Docker 23.0.3 where we execute in a container LLaMA 3.1 using Python 3.9.

B. Q1: SHAP AND LIME BASED NOISE DETECTION

To assess how effectively models distinguish relevant from irrelevant evidence, we evaluate evidence labeling as a binary task distinguishing *Useful* from *Noise* evidence.

GFCE and LLM-based verifiers (GPT-4.1-Mini, Claude 3.5 Sonnet, and LLaMA) output a binary label for each evidence piece together with a label for the overall claim. For LLM-based verifiers, these evidence labels correspond to prompted self-reported evidence selections produced by the model, rather than to external post-hoc attributions of the model's internal decision process. However, such outputs do not necessarily specify the causal role of each evidence piece in the decision (e.g., whether it steers the verifier toward a *Supports* or *Refutes* prediction). While prompting can elicit this information, it does not guarantee that the verifier internally relies on the evidence in that manner. For SHAP and LIME, we rely on attribution scores to derive binary evidence labels. For each evidence piece, we compute the sum of absolute contribution values as a relevance score. The underlying hypothesis is that noisy evidence should leave the model's prediction largely unchanged, whereas useful evidence should significantly influence the claim label.

To convert continuous attribution scores into binary labels, we infer a dataset–model–explainer-specific threshold. For each configuration, we calibrate the threshold using explanation scores for all evidence pieces associated with 100 claim-level examples from the training set. The threshold is selected to maximize the sum of F1 scores for both the ''Useful'' and ''Noise'' categories. The learned threshold is then applied to explanations generated on the test set, ensuring that calibration is tailored to each configuration while avoiding information leakage. Continuous attribution scores are converted into binary Useful/Noise labels using the threshold obtained. Additional details on threshold computation are provided in Appendix B.

Discussion.

Table 4 summarizes the experimental results. Each row corresponds to a specific dataset–verifier pair, representing the observed model under analysis. The *Verifier* columns report the claim-labeling performance of each model. The *Explainer* columns present the evidence-level performance (Accuracy, F1 Useful, and F1 Noise) achieved by native explainer components, when available. Finally, the *SHAP/LIME* columns report the same evidence-level metrics computed using post-hoc attribution methods. The SHAP/LIME evidence-labeling metrics in Table 4 should be interpreted as the output of a thresholded attribution pipeline. Therefore, these metrics reflect the combined effect of the attribution method and the calibrated threshold, rather than attribution quality alone. The threshold is learned only from training examples and then fixed on the test set, avoiding test leakage while still making threshold choice part of the evaluated procedure.

Overall, SHAP generally provides more accurate results than LIME across datasets, particularly for Accuracy and F1 Noise. Across dataset–model pairs, SHAP achieves higher scores than LIME in 62%, 69%, and 54% of the comparisons for Accuracy, F1 Useful, and F1 Noise, respectively.

When focusing on verifiers for which post-hoc attribution is computed (RoBERTa, GFCE, and LLaMA), post-hoc attribution methods remain competitive with native explainers. GFCE provides an illustrative case: despite being explicitly designed with a native explainer, post-hoc methods outperform it in 13 out of 15 evidence-level comparisons. Across these verifiers, post-hoc methods frequently match or exceed native evidence-labeling performance, while LLaMA's native explainer achieves particularly strong performance in identifying useful evidence.

We note that the evidence-labeling scores reported for LLM-based verifiers and those reported for SHAP/LIME

TABLE 3. Performance comparison across different models and datasets in terms of claim verification (left), native or prompted evidence selection (middle), and post-hoc SHAP/LIME attribution after calibrated thresholding (right). Best evidence-level score for every dataset and observed model pair in bold. For LLM-based verifiers, evidence-labeling scores correspond to prompted self-reported evidence selection and should not be interpreted as fully comparable to SHAP/LIME attribution scores.

DS	Model	Verifier				Explainer			SHAP/LIME		
		Acc.	F1 Sup	F1 Nei	F1 Ref	Acc.	F1 Useful	F1 Noise	Acc.	F1 Useful	F1 Noise
Fev.2l	Roberta	0.61	0.73	-	0.35	-	-	-	**0.88**/0.83	0.24/0.30	0.93/0.90
	GFCE	0.48	0.03	-	0.64	0.78	0.31	0.87	0.56/0.83	0.27/**0.40**	0.69/0.90
	LLaMA	0.69	0.70	-	0.68	0.88	0.47	0.93	0.88/0.86	0.21/0.20	**0.94**/0.92
	TF-IDF	-	-	-	-	**0.90**	0.28	**0.95**	-	-	-
	Claude 3.5 Sonnet	0.69	0.65	-	0.72	**0.90**	0.49	0.94	-	-	-
	GPT-4.1 Mini	0.66	0.62	-	0.70	**0.90**	**0.50**	0.94	-	-	-
Fev.3l	Roberta	0.60	0.71	0	0.41	-	-	-	**0.89**/0.88	0.35/**0.40**	**0.94**/0.93
	GFCE	0.59	0.69	0	0.46	0.79	0.33	0.88	0.80/0.82	0.36/0.34	0.88/0.89
	LLaMA	0.57	0.70	0.17	0.44	0.88	0.46	0.93	0.83/0.84	0.33/0.26	0.90/0.91
	TF-IDF	-	-	-	-	0.90	0.28	**0.95**	-	-	-
	Claude 3.5 Sonnet	0.47	0.59	0.19	0.55	**0.91**	0.52	**0.95**	-	-	-
	GPT-4.1 Mini	0.50	0.59	0.19	0.58	**0.91**	**0.53**	**0.95**	-	-	-
SciFact	Roberta	0.67	0.69	0.75	0.49	-	-	-	0.83/0.86	0.28/0.23	0.91/0.92
	GFCE	0.37	0.52	0	0.23	0.38	0.20	0.50	0.66/0.58	0.21/0.20	0.79/0.72
	LLaMA	0.76	0.80	0.71	0.72	0.80	0.51	0.88	**0.87**/0.85	**0.29**/0.26	**0.93**/0.92
	TF-IDF	-	-	-	-	0.41	0.49	0.30	-	-	-
	Claude 3.5 Sonnet	0.85	0.86	0.85	0.82	**0.88**	**0.60**	**0.93**	-	-	-
	GPT-4.1 Mini	0.83	0.88	0.82	0.78	0.86	0.58	0.91	-	-	-
FM2	Roberta	0.70	0.72	-	0.68	-	-	-	0.87/**0.88**	0.27/0.16	**0.93/0.93**
	GFCE	0.58	0.59	-	0.56	0.60	0.32	0.71	0.87/0.83	0.09/0.20	**0.93**/0.90
	LLaMA	0.87	0.88	-	0.87	0.85	0.58	0.91	**0.88**/0.85	**0.40**/0.38	**0.93**/0.92
	TF-IDF	-	-	-	-	0.81	0.16	0.89	-	-	-
	Claude 3.5 Sonnet	0.87	0.88	-	0.87	**0.90**	**0.68**	**0.94**	-	-	-
	GPT-4.1 Mini	0.86	0.86	-	0.86	0.88	0.64	0.93	-	-	-
AVTC	Roberta	0.79	0.86	0.52	0.78	-	-	-	**0.78/0.78**	0.49/0.49	**0.86/0.86**
	GFCE	0.59	0.50	0.50	0.69	0.37	0.49	0.17	0.70/0.69	0.48/0.44	0.79/0.79
	LLaMA	0.76	0.77	0.44	0.85	0.88	**0.81**	0.92	0.77/0.75	**0.63**/0.54	0.84/0.83
	TF-IDF	-	-	-	-	0.83	0.36	0.90	-	-	-
	Claude 3.5 Sonnet	0.76	0.84	0.46	0.84	**0.89**	0.80	**0.93**	-	-	-
	GPT-4.1 Mini	0.75	0.82	0.42	0.86	**0.89**	0.80	0.92	-	-	-

measure different types of evidence-level signals. For LLaMA, GPT-4.1-Mini, and Claude 3.5 Sonnet, evidence labels correspond to prompted self-reported evidence selections, whereas SHAP and LIME estimate post-hoc attributions of a verifier's output under evidence perturbations.

When considering LLM-based Verifiers and Explanations, prompted self-reported evidence selections achieve strong evidence-labeling scores, especially for the API-based models. Claude 3.5 Sonnet achieves the strongest overall performance, ranking first in F1-Noise and evidence-labeling accuracy on four out of five datasets, while tying with the top-performing models on F1-Useful. These findings suggest that prompted LLM evidence selection can be effective at distinguishing noisy evidence and identifying useful evidence. However, these scores should be interpreted as the LLMs' selected useful evidence rather than as external attribution of the internal decision process, and they do not establish whether the selected evidence drove the model prediction. Interestingly, this trend does not hold for LLaMA 3.1 70B. We attribute this effect to stronger intrinsic evidence discrimination capabilities in the API-based LLMs considered here and to the fact that SHAP is evaluated with respect to the verifier's outputs, which may not fully align with the LLM's internal rationale. Applying SHAP directly to large LLMs could provide more comparable attribution signals, but this remains computationally demanding and is left for future work.

The TF–IDF baseline reveals a consistent pattern across datasets: identifying useful evidence (F1-Useful) is substantially more challenging than detecting noise. Although TF–IDF achieves overall accuracy comparable to more sophisticated approaches in several cases, its F1-Useful scores are markedly lower on four out of five datasets. In the only dataset where TF–IDF attains competitive F1-Useful performance, this improvement is accompanied by a noticeable drop in F1-Noise. These results highlight the limitations of purely lexical similarity measures, which struggle to capture the semantic relevance required to

distinguish genuinely useful evidence from superficially related but noisy content, without capturing whether such evidence supports or refutes the claim.

Beyond raw performance comparisons, an important insight emerges. Since post-hoc attribution is estimated from perturbations of the verifier's outputs, one would expect SHAP and LIME to expose weaknesses in how the verifier reacts to evidence removal or inclusion. In the case of GFCE, however, post-hoc explainers frequently outperform the native explainer, suggesting that the underlying verifier encodes useful information for useful-versus-noise separation that is not fully captured by its jointly trained explanation component. These results indicate that evidence-level shortcomings may stem more from the explainer design than from the verifier itself. The analysis in Section V-C further supports this interpretation.

More generally, the strong performance of post-hoc methods on F1-Noise suggests that fact-checking models are relatively effective at distinguishing irrelevant evidence. In contrast, identifying genuinely useful evidence remains substantially more challenging.

Take-away Q1: Post-hoc methods effectively discern noisy evidence, indicating that fact-checking models capture signals for filtering irrelevant content. At the same time, identifying genuinely useful evidence remains a challenging problem.

C. Q2: MODEL RELIANCE ON EVIDENCE AND CLAIM

We analyze the extent to which verifier models rely on the claim, the gold evidence, and injected noise when making predictions. We focus on LLaMA, RoBERTa, and GFCE, for which post-hoc attribution analyses were computed.

We consider separately the examples labeled by a model as *Supports*, *Refutes* or *Not Enough Information*. For each example, we compute three contributions: the one from the useful evidence (obtained by averaging the contributions for all useful evidence pieces of an example), the one from the noisy evidence (by averaging them as well), and the one from the intercept, which we denote as the claim/baseline component. This component is useful for diagnosing the extent to which predictions are not explained by explicit evidence-piece attributions, but it should not be read as a purely causal claim-only contribution. We report the analysis for each dataset and each verifier for which post-hoc attribution was computed, resulting in a total of 15 plots. The distribution of these contributions for all examples in the test set is shown in Figure 4 (SHAP) and Figure 5 (LIME) as useful (evidence), noise (evidence), and claim/baseline component. Computing the average contribution for useful and noise lets us analyze their contribution independently from their unbalance in quantity. Even though models may assign greater contributions to useful evidence than over noise, in some experiments the higher number of noisy evidence makes the sum of their contribution greater than one for useful.

Discussion. The observed behavior is consistent across SHAP (Figure 4) and LIME (Figure 5). In the experiments, the average contribution of useful evidence in claim labeling is generally greater than that from noisy evidence. The exception is the *NEI* label, where noise contributes as much as useful evidence. For this label, verifiers show a stronger claim/baseline component. This behavior indicates that predictions are less explained by individual evidence-piece attributions, consistent with the lack of sufficient useful evidence and with the results in Table 4.

In RoBERTa, for Fev. 2L most of the contribution to the *Refutes* label comes from the useful evidence. In the *Supports* case, the median of the claim/baseline component is the only score clearly deviating from zero. This may suggest that the model labels a claim as Supports when it cannot find any evidence to refute it, rather than relying on explicit supporting evidence. On Fev. 3L, the model displays a similar behavior as on Fev. 2L. In the AVeriTeC dataset, RoBERTa shows a dominant claim/baseline component across all labels. We observe the same behavior for GFCE. This reliance on the claim/baseline component, rather than on useful evidence, may undermine the verifier's trustworthiness, as reflected in the mediocre results in claim labeling in Table 4.

For LLaMA, the post-hoc attribution patterns suggest that evidence pieces play a substantial role in the verifier's predictions. However, because the LLaMA analysis uses a reduced perturbation budget, these results should be interpreted as qualitative diagnostic patterns rather than as precise attribution estimates directly comparable in variance to the 500-sample RoBERTa and GFCE settings. Comparing LLaMA's behavior on datasets with 2 and 3 labels suggests a shift in the attribution patterns: the introduction of the *NEI* label is associated with higher evidence-level contributions for the *Refutes* class. This suggests a transition from an implicit *Refutes* behavior to a more nuanced *NEI*-aware pattern, which appears consistent with the intended role of the additional label.

Take-away Q2: Using post-hoc analysis, we uncover that verifiers rely differently on evidence and on the claim/baseline component when making decisions. This approach suggests significant differences in how models weigh explicit evidence-piece attributions and non-evidence baseline components in their decision-making process.

D. Q3: CONTRIBUTION SPREAD ON EVIDENCE

We assess whether attribution scores are evenly distributed across evidence pieces. For each example, we rank evidence sentences in descending order of contribution and compute the average contribution at each rank across the dataset. Our analysis focuses on LLaMA, RoBERTa, and GFCE, for which post-hoc attribution scores are available.

Discussion. Results in Figure 6 suggest that, as observed for Q2, verifiers assign evidence-level contributions differently across models and datasets. GFCE is weakly influenced by the evidence, with a maximum contribution of the useful evidence of 0.2 across every dataset on the *Supports* and

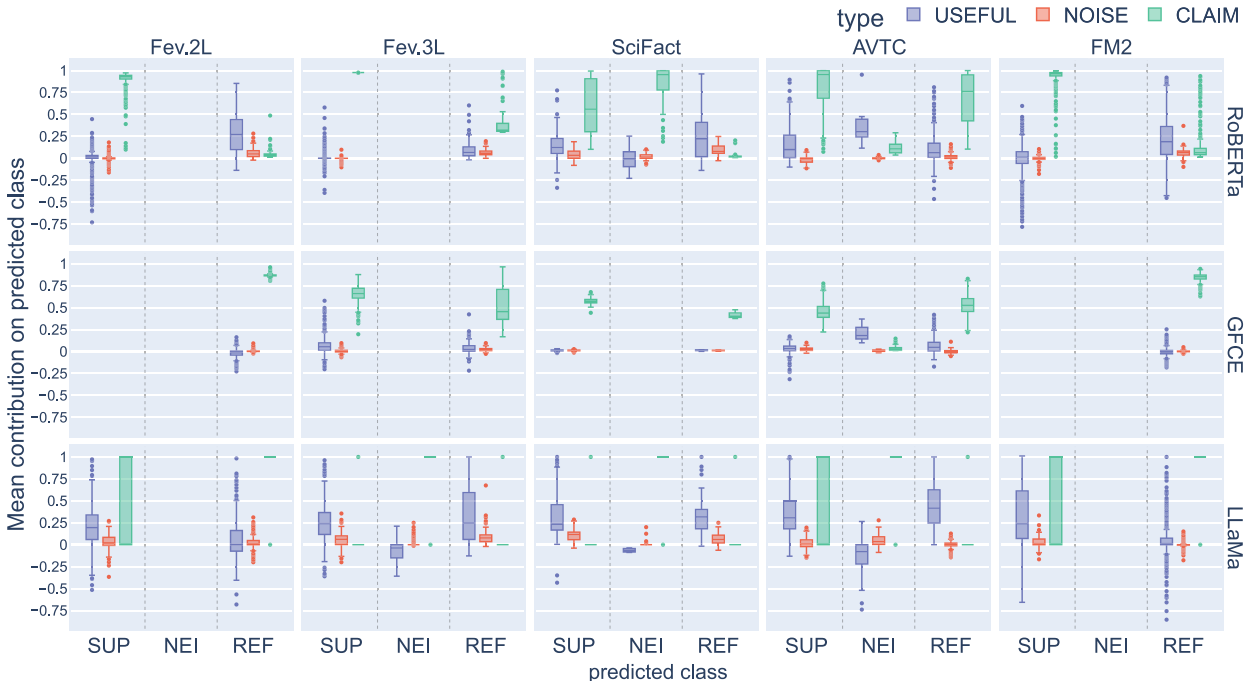

FIGURE 4. Distribution of contribution scores obtained with SHAP of the useful evidence, noisy evidence, and claim/baseline component on the predicted classes (Supports, Refutes, NEI). We spread horizontally the five datasets and vertically the three observed models. Every plot reports the predicted class over the x-axis and the prediction contributions for useful evidence, noise, and the claim/baseline component. We run the explanation on the whole test set, including the incorrect predictions.

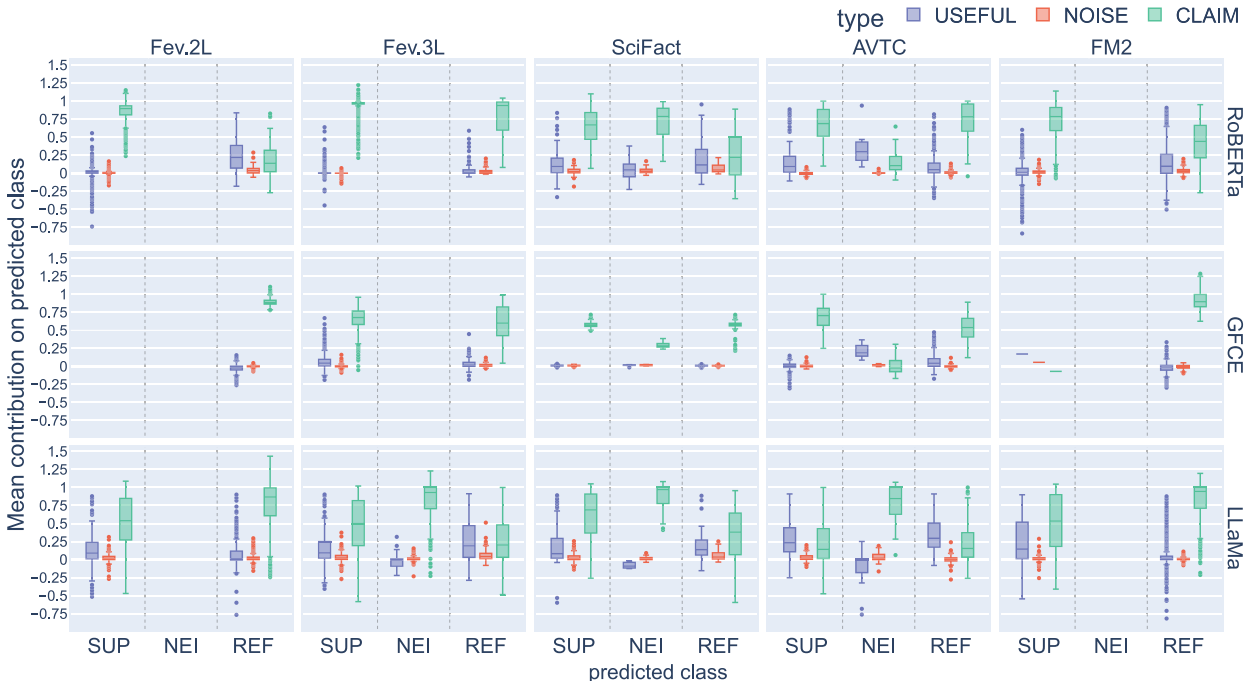

FIGURE 5. Distribution of contribution scores obtained with LIME of the useful evidence, noisy evidence, and claim/baseline component on the predicted classes (Supports, Refutes, NEI). We spread horizontally the five datasets and vertically the three observed models. Every plot reports the predicted class over the x-axis and the prediction contributions for useful evidence, noise, and the claim/baseline component. We run the explanation on the whole test set, including the incorrect predictions.

Refutes labels. When the predicted label is *NEI*, GFCE and RoBERTa on AVeriTeC show a high contribution for the most useful evidence, suggesting that the models switch decision to *NEI* based on a single piece of evidence. This pattern is

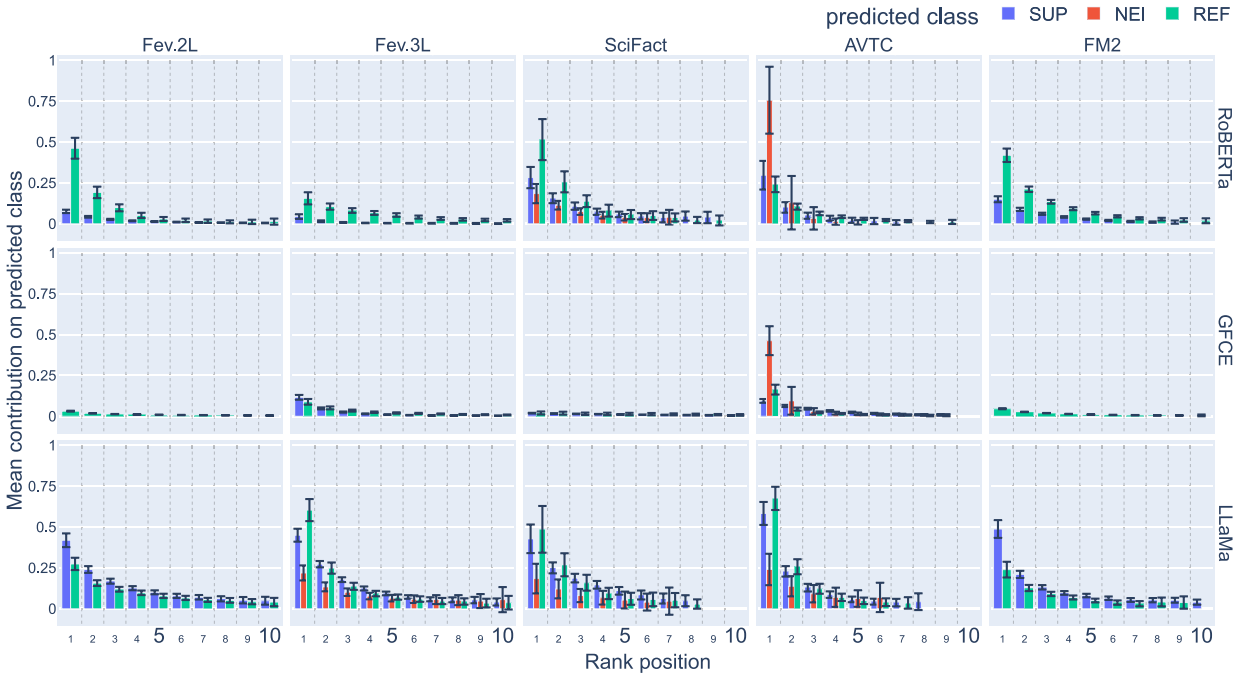

FIGURE 6. Mean contribution score on the predicted class for the evidence pieces in an example. Scores are grouped by predicted class and in descending order, removing evidence pieces contributing negatively to the predicted class. Experiment run on each verifier (rows) and each dataset (columns), scores obtained with SHAP.

counterintuitive, since adding evidence can lead the model toward a lack-of-information prediction. This suggests that the verifier did not correctly learn the implications of the NEI label, likely due to the small percentage of such claims in the training data.

Next, we look at contribution ranking for evidence across models and datasets. For LLaMA, the attribution patterns suggest a more distributed use of evidence than for the other verifiers, especially for the *Supports* and *Refutes* labels. However, because LLaMA is analyzed with a reduced perturbation budget, these results should be interpreted as qualitative diagnostic patterns rather than as precise attribution estimates directly comparable in variance to the 500-sample RoBERTa and GFCE settings. Across datasets, the highest-ranked evidence contribution for LLaMA is consistently substantial, suggesting that the model tends to assign non-negligible importance to at least one evidence piece. This is noteworthy for two reasons. First, LLaMA has not been fine-tuned for the fact-checking task, in contrast with RoBERTa and GFCE. Second, the LLM's extensive knowledge base could enable it to disregard the provided evidence. One possible explanation is that this behavior is related to the size of the model (70B), as the same model with only 8B shows lower results (see Appendix D). We hypothesize that larger LLMs may be better able to exploit relevant parts of the provided context, whereas task-specific models may be more prone to shortcut behavior.

Finally, looking at the results by label, the evidence plays a key role for RoBERTa when it predicts *Refutes*, while this behavior is less evident for *Supports*. On the other hand, LLaMA shows a more balanced attribution pattern between the two labels. In GFCE, the low contribution of the evidence involves all labels.

Take-away Q3: Post-hoc analysis suggests that evidence impact is unevenly distributed across verifiers, with models primarily focusing on few evidence pieces rather than utilizing all available information equally.

VI. CONCLUSION

We address the black-box nature of neural fact-checking models by applying post-hoc XAI methods to identify useful evidence and its contribution to claim classification across four established datasets. Our findings suggest that post-hoc attribution methods provide useful evidence-level diagnostic signals, helping to identify when verifier predictions are associated with specific evidence pieces, when noisy evidence has limited influence, and when predictions appear to depend more strongly on baseline components not attributable to individual evidence. We demonstrate that post-hoc attribution methods provide evidence-level signals that are competitive with native explainers across fine-tuned and locally deployed verifiers. We also show that lexical similarity is not enough for correct evidence attribution. Our work is a step toward more trustworthy verifiers, by enabling a comprehensive benchmark for assessing their explainability. This benchmark serves as a critical reference point for researchers aiming to develop new explainable models, particularly in setting expected standards for evidence attribution.

Limitations and Future Work. While our benchmark provides a first evidence-level comparison of post-hoc explainers across multiple verifier models and datasets, several limitations remain. First, our results are obtained on a fixed set of domains and evidence retrieval pipelines; the stability of evidence attributions under domain shift and different retrieval settings warrants a dedicated generalization study. Second, LIME and SHAP are post-hoc methods and may not always produce faithful rationales, e.g., they can be sensitive to perturbation choices and may reflect spurious correlations rather than causal evidence. Moreover, the binary Useful/Noise evaluation depends on a calibrated threshold, and the LLaMA attribution analysis uses a reduced perturbation budget for computational reasons. These choices make the analysis practical across several datasets and verifiers, but they should be considered when interpreting fine-grained numerical comparisons. In addition, due to computational constraints, perturbation-based analysis is applied only to locally accessible models, limiting direct comparability with API-based LLMs. Finally, excluding the Conflicting Evidence/Cherry-picking label in AVeriTeC narrows the evaluation to the labels supported by all considered verifiers and may underrepresent mixed-evidence cases in which both supporting and refuting evidence are present. Future work should extend the benchmark to verifiers and attribution settings that explicitly support such labels. Explainability methods can also inherit or amplify dataset and model biases, potentially affecting which pieces of evidence are highlighted for different topics or entities. Future work can extend the benchmark to additional attribution techniques (e.g., Integrated Gradients, LRP) and to broader model families, and incorporate stronger reliability analyses (e.g., uncertainty estimates, sensitivity tests, and human-centered evaluations) to better characterize robustness and trustworthiness of evidence-level explanations.

APPENDIX A
SHAP AND LIME VERSUS INTEGRATED GRADIENTS AND LRP

As an additional sanity check for our approach, we compare evidence attribution quality obtained with SHAP and LIME against Integrated Gradients (IG) [30] and Layerwise Relevance Propagation (LRP) [31]. For the latter, we specifically employ the transformer-adapted LXT-LRP variant [41], originally developed to extend relevance decomposition techniques from BERT [42] to modern attention-based architectures. For both IG and LRP, we compute token-level relevance for each evidence passage, average scores per evidence segment, and apply the threshold calibration method detailed in Appendix B. Unlike SHAP and LIME, which can operate on both local and hosted API models, IG and LRP require full access to internal model activations, making them unsuitable for closed-source large-model endpoints. Our results show that SHAP and LIME outperform IG and LRP on FEVEROUS when paired with RoBERTa, as presented in Table 5. This comparison is intended as a representative sanity check on FEVEROUS with RoBERTa, not as a general ranking of attribution methods. Extending the IG/LRP comparison to all datasets and models would require access to internal activations for each verifier and is outside the scope of the present benchmark. In this FEVEROUS–RoBERTa setting, SHAP/LIME achieve the strongest evidence-level scores, particularly for identifying useful evidence. This result should be interpreted as supporting evidence for the selected benchmark configuration, rather than as a general ranking across attribution methods.

TABLE 4. Evidence-level comparison of four attribution strategies on FEVEROUS.

Metric	SHAP	LIME	IG	LRP-LXT
Accuracy	**0.89**	0.88	0.85	0.76
F1 Useful	0.35	**0.40**	0.18	0.19
F1 Noise	**0.94**	0.93	0.92	0.86

Algorithm 2 Threshold Calibration Using 100 Training Examples

Require: Training set $\mathcal{D}$, max calibration size $N{=}100$
Require: Attribution method Score(claim, evidence)

Select first N claim–level examples: $\mathcal{D}_{calib} \leftarrow \{(c_i, E_i, y_i)\}_{i=1}^{N}$, where c_i is the claim i, $E_i = \{e_{i1}, \ldots, e_{ik}\}$ are evidence pieces and $y_{ij} \in \{0, 1\}$ are gold evidence tags.
Initialize lists $S \leftarrow []$ and $Y \leftarrow []$ ▷ S scores, Y labels
for $i = 1$ to N **do**
  for each evidence e_{ij} in E_i **do**
    $s_{ij} \leftarrow \text{Score}(c_i, e_{ij})$ ▷ SHAP/LIME/IG/...
    Append s_{ij} to S, and y_{ij} to Y
  end for
end for
Define threshold search range:
$T = \{t \mid t \text{ evenly spaced between } \min(S) \text{ and } \max(S)\}$
for each $t \in T$ **do**
  $\hat{y}_{ij}(t) = \mathbb{1}[s_{ij} \geq t]$ ▷ Binary useful/noise tag
  $F1_{\text{use}}(t)$ = F1-score for class 1 (Useful)
  $F1_{\text{noise}}(t)$ = F1-score for class 0 (Noise)
  $M(t) = \frac{1}{2}\left(F1_{\text{use}}(t) + F1_{\text{noise}}(t)\right)$
end for
return best threshold: $t^{\star} = \arg\max_{t \in T} M(t)$

APPENDIX B
COMPUTATION OF THRESHOLD FOR USEFUL/NOISE

Our SHAP and LIME pipelines, as well as Integrated Gradients [30], TF–IDF, and LRP [31], require selecting a threshold to distinguish useful evidence from noise. This threshold is calibrated using 100 examples, as shown in Algorithm 2. The calibration step is part of the evaluated evidence-labeling pipeline: continuous explanation or relevance scores are first produced by the corresponding method and are then converted into binary Useful/Noise labels through the

TABLE 5. Average execution time (seconds) to compute one explanation with LIME/SHAP. Values are mean (standard deviation) after applying the conservative outlier-removal rule described in Appendix C.

Explainer	Model	AVTC	FM2	Fev.2L	Fev.3L	SciFact
LIME	GFCE	60.96 (24.7)	53.66 (7.8)	40.60 (7.3)	44.17 (10.3)	19.75 (2.3)
	LLaMA	40.65 (10.0)	45.55 (8.1)	29.80 (19.5)	42.63 (16.5)	33.29 (13.1)
	RoBERTa	9.74 (3.3)	7.05 (1.6)	8.96 (3.0)	8.05 (3.0)	6.34 (1.9)
SHAP	GFCE	65.72 (22.8)	66.76 (13.9)	51.33 (14.1)	51.32 (13.9)	55.49 (17.3)
	LLaMA	45.37 (11.2)	56.04 (13.7)	51.93 (18.3)	53.28 (23.5)	31.70 (17.4)
	RoBERTa	5.87 (3.5)	6.37 (1.7)	5.91 (3.6)	6.11 (3.8)	3.48 (3.0)

calibrated threshold. Consequently, the evidence-labeling F1 scores reported in Table 4 should be interpreted as the result of the score-plus-threshold pipeline, rather than as an isolated measurement of attribution quality alone.

In this algorithm, we calibrate the threshold on a small subset of the training data (first $N = 100$ claim-level examples). For each claim–evidence pair, we compute a method-specific score (e.g., a SHAP/LIME attribution score or a TF–IDF relevance score) and collect the corresponding gold label (useful vs. noise). We then evaluate a set of candidate thresholds evenly spaced between the minimum and maximum observed scores. For each threshold t, scores are converted into binary predictions (useful if $s \geq t$, noise otherwise), and we compute the F1-score for both classes. The final threshold $t^\star$ is selected by maximizing the average of the F1-scores for useful and noise evidence, ensuring a balanced trade-off between identifying relevant evidence and filtering noise. The selected threshold is then kept fixed and applied to the corresponding test set. This prevents test-set leakage, while making the thresholding strategy an explicit component of the evaluated procedure. Different calibration strategies could lead to different Useful/Noise trade-offs; a broader sensitivity analysis over thresholding choices is therefore left for future work.

APPENDIX C
RUNTIME CONSIDERATIONS FOR POST-HOC ATTRIBUTION

A practical limitation of perturbation-based explainers is runtime. Both LIME and SHAP require repeatedly querying the verifier on multiple perturbed evidence subsets (or feature coalitions), so their cost scales approximately linearly with (i) the number of perturbation samples, (ii) the verifier inference latency, and (iii) the average input size (claim + evidence) and the number of evidence segments that can be masked. In our setup, we use 500 samples per explanation for RoBERTa and GFCE, while we reduce the budget to 100 for LLaMA to keep the analysis tractable; in contrast, applying these explainers to hosted API endpoints would require hundreds of API calls per instance and is therefore impractical at scale.

Table 6 reports the average wall-clock execution time required to produce a single evidence-level explanation across datasets and verifiers. Before computing the reported summary statistics, we applied a conservative outlier-removal rule uniformly to all runtime configurations. Specifically, for each configuration, we removed only values outside the interval $[Q_1 - 10 \cdot IQR, Q_3 + 10 \cdot IQR]$, where Q_1 and Q_3 are the first and third quartiles and $IQR = Q_3 - Q_1$. This rule is substantially wider than the standard IQR rule and is intended to remove only extreme anomalous executions, such as cases potentially caused by machine stalls or transient system-level issues. In our experiments, this correction affected only the LIME + GFCE configuration on FM2.

Two trends are apparent. First, runtimes are dominated by the underlying verifier: RoBERTa yields explanations in a few seconds, while larger verifiers require tens of seconds per instance even with a reduced perturbation budget. Second, runtimes vary across datasets and configurations, which is partly explained by differences in evidence cardinality across examples. When an instance contains only a small number of evidence segments, the perturbation procedure may generate repeated (or identical) masked configurations across samples. In such cases, the effective number of distinct model queries is reduced: a repeated configuration only incurs the cost of a single forward pass, so the explainer behaves as if it had fewer *virtual* samples and therefore fewer total predictions. This yields non-uniform runtimes across instances and contributes to the observed standard deviations. Overall, these results motivate (i) reducing the perturbation budget for expensive verifiers, (ii) batching perturbed instances when possible, and (iii) restricting large-scale attribution studies to local/open models when cost or latency constraints are binding.

APPENDIX D
LLAMA UNDER LOW-RESOURCES CONSTRAINTS

LLaMA 3.1 70B requires a high-end GPU such as NVIDIA A100. In a low-resource setting, a user might want to be able to use an 8B variant of the model. We show in Table 6 the impact of this decision. The clear impact of the model switch, with more than 20 points lost in some cases, makes this smaller model not an ideal choice for the user; both in terms of evidence and claim labeling.

APPENDIX E
INFORMATION ABOUT USE OF AI ASSISTANTS

In the preparation of this manuscript, we utilized an AI assistant to aid in various aspects, including coding, rewriting, and providing suggestions.

TABLE 6. Performance metrics of LLaMA 3.1 70B versus 8B (scores in parenthesis) for Claim labeling and Evidence labeling across datasets.

Claim labeling				
Dataset	**Accuracy**	**F1 Sup**	**F1 NEI**	**F1 Ref**
Fev. 3L	0.57 (0.54)	0.70 (0.61)	0.17 (0.12)	0.44 (0.53)
Fev. 2L	0.69 (0.60)	0.70 (0.64)	-	0.68 (0.53)
SciFact	0.76 (0.50)	0.80 (0.69)	0.71 (0.18)	0.72 (0.41)
FM2	0.87 (0.65)	0.88 (0.52)	-	0.87 (0.73)
AVeriTeC	0.76 (0.63)	0.77 (0.49)	0.44 (0.27)	0.85 (0.75)
Evidence labeling				
Dataset		**Accuracy**	**F1 Ev.**	**F1 Noise**
Fev. 3L		0.88 (0.75)	0.46 (0.28)	0.93 (0.85)
Fev. 2L		0.88 (0.76)	0.47 (0.29)	0.93 (0.85)
SciFact		0.80 (0.53)	0.51 (0.30)	0.88 (0.64)
FM2		0.85 (0.59)	0.58 (0.30)	0.91 (0.71)
AVeriTeC		0.88 (0.60)	0.81 (0.54)	0.92 (0.64)

Throughout this process, we maintained a critical review of the AI's contributions, thoroughly double-checking and revising the text to uphold the quality and integrity of the final work. The human authors remained in full control, ensuring that the manuscript reflects their expertise and scholarly standards.

REFERENCES

[1] Z. Guo, M. Schlichtkrull, and A. Vlachos, "A survey on automated fact-checking," *Trans. Assoc. Comput. Linguistics*, vol. 10, pp. 178–206, Feb. 2022, doi: 10.1162/tacl_a_00454.

[2] A. Saakyan, T. Chakrabarty, and S. Muresan, "COVID-fact: Fact extraction and verification of real-world claims on COVID-19 pandemic," in *Proc. 59th Annu. Meeting Assoc. Comput. Linguistics, 11th Int. Joint Conf. Natural Lang. Process.*, 2021, pp. 2116–2129, doi: 10.18653/v1/2021.acl-long.165.

[3] T. Schuster, A. Fisch, and R. Barzilay, "Get your vitamin C! Robust fact verification with contrastive evidence," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol.*, Jun. 2021, pp. 624–643. [Online]. Available: https://aclanthology.org/2021.naacl-main.52

[4] P. Nakov, D. Corney, M. Hasanain, F. Alam, T. Elsayed, A. Barrón-Cedeño, P. Papotti, S. Shaar, and G. Da San Martino, "Automated fact-checking for assisting human fact-checkers," in *Proc. 30th Int. Joint Conf. Artif. Intell.*, Aug. 2021, pp. 4551–4558, doi: 10.24963/ijcai.2021/619.

[5] M. Saeed, G. Alfarano, K. Nguyen, D. Pham, R. Troncy, and P. Papotti, "Neural re-rankers for evidence retrieval in the FEVEROUS task," in *Proc. 4th Workshop Fact Extraction Verification (FEVER)*, Nov. 2021, pp. 108–112. [Online]. Available: https://aclanthology.org/2021.fever-1.12

[6] A. Madsen, S. Reddy, and S. Chandar, "Post-hoc interpretability for neural NLP: A survey," 2021, *arXiv:2108.04840*.

[7] M. T. Ribeiro, S. Singh, and C. Guestrin, "'Why should i trust you?': Explaining the predictions of any classifier," 2016, *arXiv:1602.04938*.

[8] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in *Proc. 31st Int. Conf. Neural Inf. Process. Syst.*, 2017, pp. 4768–4777.

[9] N. Kotonya and F. Toni, "Explainable automated fact-checking: A survey," in *Proc. 28th Int. Conf. Comput. Linguistics*, Barcelona, Spain, Dec. 2020, pp. 5430–5443. [Online]. Available: https://aclanthology.org/2020.coling-main.474/

[10] N. Kotonya and F. Toni, "Explainable automated fact-checking for public health claims," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, Nov. 2020, pp. 7740–7754. [Online]. Available: https://aclanthology.org/2020.emnlp-main.623

[11] S. Althabiti, M. A. Alsalka, and E. Atwell, "Generative AI for explainable automated fact checking on the FactEx: A new benchmark dataset," in *Proc. Disinformation Open Online Media*. Cham, Switzerland: Springer, 2023, pp. 1–13.

[12] R. Linder, S. Mohseni, F. Yang, S. K. Pentyala, E. D. Ragan, and X. B. Hu, "How level of explanation detail affects human performance in interpretable intelligent systems: A study on explainable fact checking," *Appl. AI Lett.*, vol. 2, no. 4, p. 49, Dec. 2021. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/ail2.49

[13] J. Yang, D. Vega-Oliveros, T. Seibt, and A. Rocha, "Explainable fact-checking through question answering," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2022, pp. 8952–8956, doi: 10.1109/ICASSP43922.2022.9747214.

[14] Z. Zhang, K. Rudra, and A. Anand, "Explain and predict, and then predict again," in *Proc. 14th ACM Int. Conf. Web Search Data Mining*, Mar. 2021, pp. 418–426, doi: 10.1145/3437963.3441758.

[15] P. Atanasova, J. G. Simonsen, C. Lioma, and I. Augenstein, "Generating fact checking explanations," in *Proc. 58th Annu. Meeting Assoc. Comput. Linguistics (ACL)*, Jul. 2020, pp. 7352–7364, doi: 10.18653/v1/2020.acl-main.656.

[16] J. Bussotti and P. Papotti, "Refining attention for explainable and noise-robust fact-checking with transformers," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2025, pp. 25476–25488, doi: 10.18653/v1/2025.emnlp-main.1295.

[17] P. P. S. Dammu, H. Naidu, M. Dewan, Y. Kim, T. Roosta, A. Chadha, and C. Shah, "ClaimVer: Explainable claim-level verification and evidence attribution of text through knowledge graphs," 2024, *arXiv:2403.09724*.

[18] J. Si, Y. Zhu, and D. Zhou, "Exploring faithful rationale for multi-hop fact verification via salience-aware graph learning," in *Proc. AAAI Conf. Artif. Intell.*, Jun. 2023, vol. 37, no. 11, pp. 13573–13581, doi: 10.1609/aaai.v37i11.26591.

[19] J. Si, Y. Zhu, and D. Zhou, "Consistent multi-granular rationale extraction for explainable multi-hop fact verification," 2023, *arXiv:2305.09400*.

[20] N. Deng, Z. Sun, R. He, A. Sikka, Y. Chen, L. Ma, Y. Zhang, and R. Mihalcea, "Tables as texts or images: Evaluating the table reasoning ability of LLMs and MLLMs," 2024, *arXiv:2402.12424*.

[21] B. Xu, A. Yang, J. Lin, Q. Wang, C. Zhou, Y. Zhang, and Z. Mao, "ExpertPrompting: Instructing large language models to be distinguished experts," 2023, *arXiv:2305.14688*.

[22] A. Wan, E. Wallace, and D. Klein, "What evidence do language models find convincing?" 2024, *arXiv:2402.11782*.

[23] L. Wu, Y. Rao, Y. Zhao, H. Liang, and A. Nazir, "DTCA: Decision tree-based co-attention networks for explainable claim verification," 2020, *arXiv:2004.13455*.

[24] N. Ahmadi, J. Lee, P. Papotti, and M. Saeed, "Explainable fact checking with probabilistic answer set programming," 2019, *arXiv:1906.09198*.

[25] A. Simons. (Feb. 2023). *Evaluating Feature Attribution Methods: A Usecase on a Neural Fact-Checking Model*. [Online]. Available: https://repository.tudelft.nl/record/uuid:ccc39e3a-f233-4279-a230-dbc33ce4147e

[26] F. Vargas, I. Salles, D. Alves, A. Agrawal, T. A. S. Pardo, and F. Benevenuto, "Improving explainable fact-checking via sentence-level factual reasoning," in *Proc. 7th Fact Extraction Verification Workshop (FEVER)*, Miami, FL, USA, Nov. 2024, pp. 192–204. [Online]. Available: https://aclanthology.org/2024.fever-1.23/

[27] L. Pan, X. Wu, X. Lu, A. T. Luu, W. Y. Wang, M.-Y. Kan, and P. Nakov, "Fact-checking complex claims with program-guided reasoning," in *Proc. 61st Annu. Meeting Assoc. Comput. Linguistics*, Toronto, ON, Canada, 2023, pp. 6981–7004, doi: 10.18653/v1/2023.acl-long.386.

[28] A. Madsen, S. Chandar, and S. Reddy, "Are self-explanations from large language models faithful?" in *Proc. Findings Assoc. Comput. Linguistics (ACL)*, 2024, pp. 295–337.

[29] K. Matton, R. O. Ness, J. V. Guttag, and E. Kiciman, "Walk the talk? Measuring the faithfulness of large language model explanations," in *Proc. ICLR*, 2025.

[30] M. Sundararajan, A. Taly, and Q. Yan, "Axiomatic attribution for deep networks," 2017, *arXiv:1703.01365*.

[31] A. Binder, G. Montavon, S. Bach, K.-R. Müller, and W. Samek, "Layer-wise relevance propagation for neural networks with local renormalization layers," 2016, *arXiv:1604.00825*.

[32] R. Aly, Z. Guo, M. S. Schlichtkrull, J. Thorne, A. Vlachos, C. Christodoulopoulos, O. Cocarascu, and A. Mittal, "FEVEROUS: Fact extraction and verification over unstructured and structured information," in *Proc. Neural Inf. Process. Syst.*, Dec. 2021, pp. 1–13. [Online]. Available: https://datasets-benchmarks-proceedings.neurips.cc/paper/2021/hash/68d30a9594728bc39aa24be94b319d21-Abstract-round1.html

[33] J. Thorne, A. Vlachos, C. Christodoulopoulos, and A. Mittal, "FEVER: A large-scale dataset for fact extraction and verification," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol.*, vol. 1, New Orleans, LA, USA, Jun. 2018, pp. 809–819, doi: 10.18653/v1/n18-1074.

[34] D. Wadden, S. Lin, K. Lo, L. L. Wang, M. van Zuylen, A. Cohan, and H. Hajishirzi, "Fact or fiction: Verifying scientific claims," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, Nov. 2020, pp. 7534–7550. [Online]. Available: https://aclanthology.org/2020.emnlp-main.609

[35] J. M. Eisenschlos, B. Dhingra, J. Bulian, B. Börschinger, and J. Boyd-Graber, "Fool me twice: Entailment from Wikipedia gamification," 2021, *arXiv:2104.04725*.

[36] M. Schlichtkrull, Z. Guo, and A. Vlachos, "AVeriTeC: A dataset for real-world claim verification with evidence from the Web," 2023, *arXiv:2305.13117*.

[37] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "RoBERTa: A robustly optimized BERT pretraining approach," 2019, *arXiv:1907.11692*.

[38] Y. Nie, A. Williams, E. Dinan, M. Bansal, J. Weston, and D. Kiela, "Adversarial NLI: A new benchmark for natural language understanding," 2019, *arXiv:1910.14599*.

[39] OpenAI. (2025). *GPT-4.1 Technical Report*. [Online]. Available: https://openai.com/research

[40] Anthropic. (2025). *Claude 3.5 Sonnet Model Card*. [Online]. Available: https://www.anthropic.com

[41] R. Achtibat, S. M. V. Hatefi, M. Dreyer, A. Jain, T. Wiegand, S. Lapuschkin, and W. Samek, "AttnLRP: Attention-aware layer-wise relevance propagation for transformers," in *Proc. 41st Int. Conf. Mach. Learn.*, vol. 235, Jul. 2024, pp. 135–168. [Online]. Available: https://proceedings.mlr.press/v235/achtibat24a.html

[42] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol.*, vol. 1, Minneapolis, MN, USA, Jun. 2019, pp. 4171–4186. [Online]. Available: https://aclanthology.org/N19-1423/

JEAN-FLAVIEN BUSSOTTI received the M.Sc. degree in computer science from Télécom Paris, in 2021, and the joint Ph.D. degree from Sorbonne Université and EURECOM, France, in 2024. He is currently a Research Associate with Megagon Labs, USA. His research interests include natural language processing (NLP) and generative AI, with a particular focus on explainable AI, model architectures, and multimodal synthetic dataset generation.

ANDREA BARALDI received the B.S., M.Sc., and Ph.D. degrees from the Department of Engineering "Enzo Ferrari," University of Modena and Reggio Emilia, in 2018, 2020, and 2025, respectively.

He is currently a Data Scientist with Accenture. His main research interests include eXplainable artificial intelligence (XAI), data integration: studying explainability tools and intrinsically interpretable ML/DL models in the entity matching field; scalability and fairness: studying how to improve time/data efficiency of models that respect fairness constraints.

FRANCESCO GUERRA is currently a Full Professor with the Department of Engineering "Enzo Ferrari," University of Modena and Reggio Emilia, Italy. He teaches software engineering and big data and text analysis. His research interests include scalable data management and integration (with a focus on explainable and fair ML/DL techniques for entity resolution), data mining on big data (e.g., anomaly and novelty detection), text analytics and NLP (automatic analysis of textual and tabular data), the semantic web (ontology alignment and creation), and keyword-based search on structured datasets. For more details https://fguerra73.github.io/

PAOLO PAPOTTI received the Ph.D. degree from Roma Tre University, Italy, in 2007. He had research positions with Qatar Computing Research Institute, Qatar, and Arizona State University, USA. He has been a Professor with EURECOM, France, since 2017, and the holder of a Chair of artificial intelligence with the 3IA Institute, since 2024. His research is focused on data management and NLP.