Refining Attention for Explainable and Noise-Robust Fact-Checking with Transformers

Jean-Flavien Bussotti
Megagon Labs, USA*
jflavien@megagon.ai

Paolo Papotti
EURECOM, France
papotti@eurecom.fr

Abstract

In tasks like question answering and fact-checking, models must discern relevant information from extensive corpora in an "open-book" setting. Conventional transformer-based models excel at classifying input data, but (i) often falter due to sensitivity to noise and (ii) lack explainability regarding their decision process. To address these challenges, we introduce ATTUN, a novel transformer architecture designed to enhance model transparency and resilience to noise by refining the attention mechanisms. Our approach involves a dedicated module that directly modifies attention weights, allowing the model both to improve predictions and to identify the most relevant sections of input data after supervised training. We validate our methodology using fact-checking datasets and show promising results in question answering. Experiments demonstrate improvements of up to 51% in F1 score for detecting relevant context, and gains of up to 18% in task accuracy when integrating ATTUN into a model.[1]

1 Introduction

Transformer-based models are pivotal in numerous applications, notably in tasks like fact-checking, which has gained prominence with the rise of social media platforms (Nakov et al., 2021). In this domain, a claim, denoted as a query q, is examined for its truthfulness using supporting evidence $\hat{e}$ extracted from a vast corpus. The model processes this query-evidence pair to produce an output o, categorically labeling the claim as *Supports*, *Refutes*, or *Not Enough Information* (NEI). Despite their efficacy, traditional encoder models face challenges with noise sensitivity and lack transparency in how these decisions are reached.

Consider, for example, the claim "B. Obama was born in Hawaii". A retriever over Wikipedia pages might collect several passages related to Obama and Hawaii. Among these, the crucial passage confirming Obama's birthplace may not emerge as the top choice. Consequently, the classifier model is typically provided with 20 to 40 passages, which introduces both relevant and noisy information. Enhancing the model's ability to discern and use relevant evidence would improve both accuracy and explainability of its decisions.

In realistic settings, retriever models often generate e, a noisy version of the ideal evidence $\hat{e}$. Therefore, it is crucial to identify which parts of the context e are truly beneficial for decision-making, effectively aiming to expose the $\hat{e}$ used by the model. Human fact-checkers dismiss outputs that lack transparency about the evidence supporting the model's decision (Nakov et al., 2021). Ensuring transparency is vital for users, allowing them to verify the evidence and determine their alignment with the model's conclusions (Guo et al., 2022).

However, traditional models often function as black boxes, lacking explanations for their outputs, which makes it challenging to discern the specific evidence $\hat{e}$ used for decision-making (Bussotti et al., 2024). Current post-hoc methods, like SHAP and LIME (Ribeiro et al., 2016; Lundberg and Lee, 2017), offer insights into model outputs, but are computationally expensive, requiring numerous executions. External models attempt to classify context relevance (Atanasova et al., 2020), but often fall short in faithfully representing the model's internal decision-making. Using LLMs for both output and justification has shown limitations in correctly attributing references (Gao et al., 2023), despite attempts to address these issues through synthetic data and fine-tuning (Huang et al., 2024). While some approaches bypass retrieval by using entire documents as context (Lee et al., 2024), this renders the context e ineffective for explanation.

Attention mechanisms present a promising avenue for generating explainable outputs by allow-

*Work done while at EURECOM.
[1]Code https://github.com/JeFlBu/ATTUN.git

ing transformer models to focus on relevant information within the input. However, attention weights do not always correlate well with feature importance (Liu et al., 2022; Team, 2024). Existing methods use attention to highlight tokens with high attention scores, aiming to show which parts of the text the model considers important (Kotipalli, 2024; MLOps Community, 2024). Yet, these approaches typically neglect the context of the query during classification tasks, which significantly limits the effectiveness of such explanations (Liu et al., 2021). This underscores the need for improved methods that effectively integrate query context into attention-based explanations.

Finally, a significant challenge is the models' lack of robustness to noise (Yoran et al., 2024). While LLMs manage extensive contexts with millions of tokens (Dubey et al., 2024), their outputs can be disrupted by lengthy inputs (Lee et al., 2024). Current explainers and post-hoc methods do not enhance model performance, highlighting the need for novel architectural solutions.

We address these challenges by exploring the following research question: *Can we create lightweight models that justify their output using the input?* We introduce ATTUN (from ATTention tUNing), which offers a solution for Transformer-based models by providing more faithful explanations without increasing computational cost. Our approach integrates a module that evaluates the attention between the context and the query within the classifier's architecture. This module is trained jointly with the classifier, refining its attention parameters based on evidence relevance. This refinement enhances both the quality of explanations and the accuracy of attention values, ultimately improving the classifier's overall performance.

We validate our approach by comparing ATTUN with other explainers through extensive experiments with established systems and LLMs. We assess the scalability of our method by applying it across various datasets, using encoder-only, encoder-decoder, and decoder-only models. Our results demonstrate that ATTUN significantly enhances performance, with an increase in F1 score for detecting relevant contextual elements by up to 51%, and a boost in task accuracy by up to 18% compared to using vanilla architecture.

2 Related Work

Most interpretable fact-checking systems bolt an explainer onto a black-box verifier. Many rank evidence passages before assigning a claim label (Kotonya and Toni, 2020; Dammu et al., 2024; Saeed et al., 2021), while others train a parallel explanation head (Si et al., 2023b,a; Atanasova et al., 2020). Yet the evidence they highlight often differs from what the verifier truly uses. Post-hoc tools such as SHAP and LIME provide more faithful explanations, but at a steep computational cost (Ribeiro et al., 2016; Lundberg and Lee, 2017).

Transformer self-attention (Vaswani et al., 2017) has been probed for interpretability by visualizing attention maps (Sen et al., 2020; Pruthi et al., 2020) and scoring token/head salience (Liu et al., 2021; Voita et al., 2019; Clark et al., 2019)Because raw weights often mis-align with model reasoning (Wiegreffe and Pinter, 2019; Jain and Wallace, 2019; Sun and Lu, 2020), refined variants improve faithfulness for image-to-text (Rong et al., 2024) and ECG diagnosis (Yoo et al., 2021). Look Back Lens uses attention ratios post-hoc to flag hallucinations without extra fine-tuning (Chuang et al., 2024). Yet no prior work trains a single transformer to jointly predict and explain, or alters attention during the forward pass to control evidence selection.

3 Architecture

Our architecture identifies relevant inputs crucial for generating the answer. We now outline the input and output structures used by traditional systems in comparison to our approach. We then explain the modifications introduced to transformer models to enhance their explainability and robustness.

Structure of output explanation. Transformer models process a textual input i to produce an output o. In fact-checking, i comprises a claim q and an evidence set e, which includes both relevant and irrelevant information for labeling q. The evidence set is represented as $e = \{e_i | i \in \{0, ..., k\}\}$, where k is the total number of evidence pieces and e_i denotes each piece. Typically, we input i into the model as a concatenated string "$q|e_0|...|e_k$".

ATTUN retains this input structure but introduces an additional output, el, for evidence labeling. This output identifies relevant components of e that contributed to producing o, represented as $el = \{el_i | i \in \{0, ..., k\}\}$, where el_i is 1 if the evidence is useful, and 0 if it is noise. The purpose of el is to provide a transparent explanation to users

about which evidence influenced the model's decisions. For instance, with q="B. Obama was born in the US" and the evidence set e={"B. Obama was born in Hawaii", "Hawaii is in the US", "B. Obama was elected in 2008"}, $el = \{1, 1, 0\}$. With NEI claims, an evidence is useful if it is necessary to support or refute the claim but not sufficient.

Model refinement. We enhance the architecture of transformer models to directly use attention values A. Ignoring batch size, A is structured as $[l, h, N, N]$, where N is the number of tokens in the input i, and l and h denote the attention layers and heads, respectively. Each element from the context e_i is associated with a specific token range within $[0, N]$. We build a function, f_e, which processes A to derive a specific value for each e_i, considering its role within the entire context. The function f_e computes, for each layer and head, the average attention weight between the context input tokens – claim or full input – and the tokens of each evidence span e_i. This yields a matrix of shape $[l, h]$ per evidence representing attention interactions between the evidence and the overall input across all layers and heads. These matrices are then element-wise divided by a reference matrix computed in the same way over the claim span. The resulting matrices are flattened and passed to a linear classifier to produce a usefulness score for each evidence. Alternative strategies for computing f_e are provided in Appendix A.2. By compiling these labels, we form the set of evidence labels, $\hat{el} = \{\hat{el}_i | i \in \{0, ..., k\}\}$.

We either use a single classifier, or one classifier per label. Having a different classifiers can be beneficial when different labels require different focus on the evidence. For instance, a *Refutes* claim only requires to focus on contradicting elements.

Linear classifiers are included inside the model architecture and trained during the fine-tuning of the model. Their training objective is to minimize the binary cross entropy loss, $loss_e$, between the generated set $\hat{el}$ and the original set el. This loss is then added to the loss of the original model $loss_m$ and multiplied by a tunable coefficient γ. The total loss is $loss = loss_m + \gamma \times loss_e$.

Figure 1 illustrates the pipeline of our system. This novel architecture incorporates a loss function that refines attention parameters by focusing on key input elements, enhancing model robustness and making it internally explainable.

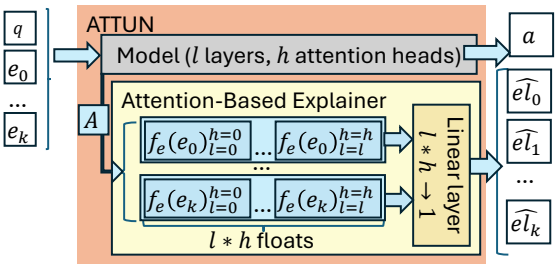

Figure 1: Integration of ATTUN attention-based explainer module into a pretrained transformer. The module processes attention values A to generate evidence labels $\hat{el}$, enhancing interpretability and robustness.

4 Experiments

We conduct experiments to evaluate ATTUN in the context of fact-checking. Additional experiments on question answering are in the Appendix.

Settings. We use four popular datasets for fact-checking: Feverous, AVeriTeC, FM2 and SciFact. We compare ATTUN against models with intrinsic explanations, such as GFCE, LLaMa3, and GPT-4o mini (the latter is evaluated both with and without additional fine-tuning on each dataset). We use both SHAP-based post-hoc explainability and ATTUN explainability on RoBERTa, Phi-4 mini, and DeBERTa-V3 verifiers, enabling direct comparisons. LLaMa3 and GPT-4o mini are prompted to identify relevant evidence prior to classifying the claim, whereas Phi-4 mini outputs a label directly. We also introduce ATTÆX (ATTention Analysis for EXplainability), a simpler variant in which the verifier is fine-tuned independently of its explainer module. Details on models and training procedures are in Appendix A.7. All models are trained and evaluated on their corresponding datasets. The *F1 Useful* metric measures the alignment between predicted and gold evidence sets, while *F1 Noise* is for the evidence correctly predicted as not useful.

Results. Table 1 summarizes our results; we provide the effect of hyperparameter variations in the Appendix. Our model ATTUN consistently achieves state-of-the-art results in evidence identification, with up to 51% improvement in F1 Useful (from 0.49 to 0.74 on Feverous) compared to other approaches. On Feverous, the retriever recall is only 0.36 (Appendix Table 8), meaning that many gold evidence sentences are missing from the input. Yet, ATTUN improves evidence identification showing its capacity to handle incomplete contexts in noisy retrieval settings. Conversely, on datasets where all gold evidence is present (i.e., recall of 1), ATTUN still brings significant gains,

Model	DS	Verifier				Explainer			DS	Verifier				Explainer		
		Acc.	F1 Sup	F1 NEI	F1 Ref	Acc.	F1 Useful	F1 Noise		Acc.	F1 Sup	F1 NEI	F1 Ref	Acc.	F1 Useful	F1 Noise
GFCE	Feverous	0.59	0.69	0.00	0.46	0.79	0.33	0.88	SciFact	0.37	0.52	0.00	0.23	0.38	0.20	0.50
LLaMA		0.57	0.70	0.17	0.44	0.88	0.46	0.93		0.76	0.80	0.71	0.72	0.80	0.51	0.88
GPT-4o mini		0.51	0.65	**0.20**	0.5	0.89	0.49	0.94		0.69	0.73	0.62	0.69	0.84	0.41	0.91
FT GPT-4o mini		0.57	0.68	0.00	0.43	0.93	0.49	0.96		0.69	0.74	0.74	0.40	0.89	0.52	0.94
RoBERTa + SHAP		*0.60*	*0.71*	*0.00*	*0.41*	*0.89*	*0.35*	*0.94*		*0.67*	*0.69*	*0.75*	*0.49*	*0.83*	*0.28*	*0.91*
RoBERTa + ATTÆX		*0.60*	*0.71*	*0.00*	*0.41*	*0.86*	*0.36*	*0.92*		*0.67*	*0.69*	*0.75*	*0.49*	*0.79*	*0.49*	*0.87*
RoBERTa + ATTUN		*0.68*	*0.75*	*0.00*	*0.62*	*0.93*	*0.73*	*0.96*		*0.79*	*0.79*	*0.86*	*0.59*	***0.91***	*0.56*	***0.95***
DeBERTa + SHAP		*0.71*	*0.77*	*0.00*	*0.68*	*0.84*	*0.29*	*0.90*		*0.83*	*0.80*	*0.83*	***0.86***	*0.89*	*0.29*	*0.94*
DeBERTa + ATTÆX		*0.71*	*0.77*	*0.00*	*0.68*	*0.88*	*0.46*	*0.93*		*0.83*	*0.80*	*0.83*	***0.86***	*0.76*	*0.34*	*0.86*
DeBERTa + ATTUN		***0.72***	***0.78***	*0.00*	***0.70***	***0.94***	***0.74***	***0.97***		***0.88***	***0.89***	***0.90***	*0.81*	***0.91***	*0.61*	***0.95***
Phi-4 mini + SHAP		*0.68*	*0.75*	*0.00*	*0.62*	*0.90*	*0.48*	*0.94*		*0.67*	*0.67*	*0.05*	*0.81*	*0.89*	*0.40*	*0.94*
Phi-4 mini + ATTÆX		*0.68*	*0.75*	*0.00*	*0.62*	*0.88*	*0.56*	*0.93*		*0.67*	*0.67*	*0.05*	*0.81*	*0.86*	*0.29*	*0.92*
Phi-4 mini + ATTUN		*0.69*	*0.76*	*0.00*	*0.62*	*0.88*	*0.61*	*0.93*		*0.70*	*0.71*	*0.18*	*0.81*	*0.90*	***0.63***	*0.94*
GFCE	FM2	0.58	0.59	-	0.56	0.60	0.32	0.71	AVeriTeC	0.59	0.50	0.50	0.69	0.37	0.49	0.17
LLaMA		0.87	0.88	-	0.87	0.85	0.58	0.91		0.76	0.77	0.44	0.85	0.88	0.81	0.92
GPT-4o mini		0.81	0.87	-	0.79	0.88	0.64	0.93		0.85	**0.90**	0.51	0.89	0.89	0.80	0.92
FT GPT-4o mini		0.88	0.88	-	0.87	0.95	0.79	**0.97**		**0.90**	**0.90**	0.65	**0.93**	0.85	0.80	0.88
RoBERTa + SHAP		*0.86*	*0.86*	-	*0.86*	*0.87*	*0.27*	*0.93*		*0.79*	*0.86*	*0.52*	*0.78*	*0.78*	*0.49*	*0.86*
RoBERTa + ATTÆX		*0.86*	*0.86*	-	*0.86*	*0.91*	*0.64*	*0.95*		*0.79*	*0.86*	*0.52*	*0.78*	*0.87*	*0.80*	*0.91*
RoBERTa + ATTUN		*0.86*	*0.86*	-	*0.86*	*0.95*	*0.80*	***0.97***		*0.83*	*0.75*	*0.50*	*0.88*	***0.92***	***0.86***	***0.93***
DeBERTa + SHAP		*0.90*	*0.90*	-	*0.90*	*0.89*	*0.47*	*0.94*		*0.79*	*0.84*	*0.00*	*0.79*	*0.77*	*0.48*	*0.85*
DeBERTa + ATTÆX		*0.90*	*0.90*	-	*0.90*	*0.83*	*0.46*	*0.90*		*0.79*	*0.84*	*0.00*	*0.79*	*0.78*	*0.67*	*0.83*
DeBERTa + ATTUN		***0.91***	***0.91***	-	***0.91***	***0.96***	***0.81***	***0.97***		***0.90***	*0.86*	***0.72***	***0.93***	*0.91*	*0.85*	***0.93***
Phi-4 mini + SHAP		*0.89*	*0.89*	-	*0.89*	*0.81*	*0.11*	*0.90*		*0.78*	*0.70*	*0.60*	*0.83*	*0.71*	*0.20*	*0.82*
Phi-4 mini + ATTÆX		*0.89*	*0.89*	-	*0.89*	*0.87*	*0.41*	*0.93*		*0.78*	*0.70*	*0.60*	*0.83*	*0.67*	*0.26*	*0.79*
Phi-4 mini + ATTUN		*0.89*	*0.89*	-	*0.89*	*0.91*	*0.65*	*0.95*		*0.78*	*0.71*	*0.00*	*0.85*	*0.85*	*0.77*	*0.89*

Table 1: Performance comparison across different models and datasets for claim verification (*Verifier*) and explanation through evidence attribution (*Explainer*). Best explanation result for every dataset in **bold**. Models including our architecture refinement are underlined. The tests are performed on test datasets from Table 7. Our primary goal, explainability, is assessed using the F1 Useful Evidence column, with accuracy also being an important factor. The performance of claim classification is primarily evaluated based on its accuracy.

confirming that its benefits extend to high-recall settings as well. These results address our primary research question. In addition to improved explainability, ATTUN yields better claim classification accuracy across several datasets — with improvements up to 18% (from 0.67 to 0.79 on RoBERTa+SciFact) compared to its non-ATTUN counterpart — demonstrating enhanced noise robustness. We attribute ATTUN's superior performances to its architecture. It enables a better evidence labeling, due to a more expressive evidence classifier, as well as an improved claim prediction, since useful evidence tokens receive refined attention scores. Through regularization via multitask learning, it prevents overfitting by penalizing biased shortcuts – a model that relies on spurious signals for claims will struggle to label evidence.

In contrast, models like GFCE – which use separate modules with a joint loss – fail to capture this synergy, likely due to insufficient cross-task signal propagation. Similarly, ATTÆX, where the evidence classifier is decoupled, underperforms, underscoring the value of shared learning in ATTUN. Using attention values, even with the manipulations performed by f_e, it is not enough to provide a relevant selection of evidence as output.

We observe that ATTUN brings the most benefits when applied to encoder-only architectures like RoBERTa and DeBERTa-V3. In contrast, decoder-only models like Phi-4 mini benefit less. We hypothesize this is due to the difference in input representation handling. Encoder models preserve a consistent mapping to input tokens throughout the layers, enabling attention-based methods to faithfully track evidence relevance while Decoder models (like Phi-4 mini), which update token representations autoregressively, shift attention toward generated abstractions, making evidence attribution less reliable. Despite this, ATTUN still improves Phi-4 mini's explainability and classification, showing general robustness across datasets.

Finally, we suggest ATTUN's generalizability by applying it to question answering, where it reaches a strong F1 Useful score of 0.75 in preliminary experiments (Appendix A.4, Table 6).

Takeaways: ATTUN significantly improves evidence attribution performance, our primary objective. Our method sets a new standard for explainability in fact-checking.

5 Conclusion and Future Work

We address noise sensitivity and explainability issues in transformer models through enhanced attention – a crucial focus given the continuous increase in input lengths. Future work could explore the application of ATTUN to instruction-based tasks.

Limitations

Although we apply our method on several fact-checking dataset, we do not evaluate on a broader range of tasks. The main reason for focusing on a limited number of tasks is the lack of annotated datasets. We aim to work on tasks as realistic as possible, sourcing evidence either from question answering or fact-checking based on real retrieved evidence. We show proofs of generalization through a first experiment on Question Answering, relying on a dataset adapted to the task thanks to its noisy passages. Our experiments already demonstrate that adding artificial noise makes the evidence classification task more artificial and biased, as the noise becomes easier to distinguish. Also, given the constraints of a short paper, adding extensive experiments on additional tasks is challenging. We leave other applications as future work.

ATTUN can be adapted to a range of tasks, including extractive QA. However, in extractive settings the explainer offers limited benefit at inference, since the answer span already reveals the relevant context. Its contribution is therefore mainly complementary, reinforcing evidence-focused learning during training, whereas tasks like fact-checking or abstractive QA showcase its full potential.

Including our module in the models complicates the training procedure. Training datasets need to have usefulness labels for their context as part of the data. In case there is none, it is not always possible to create such labels. For instance, human written context would request to use synthetic dataset generation methods to fill the gap. One can use an LLM to add noise, but it might be too trivial compared to real noise.

Ethical considerations

The deployment of fact-checking models comes with inherent risks, including potential misuse and unintended harm. A key concern is the possibility of adversaries exploiting these models to spread disinformation. By analyzing how the models operate, they could manipulate evidence sources to generate misleading yet plausible claims. ATTUN would then expose those false evidence. Mitigating this requires frequent updates to the models and improved detection of evolving misinformation tactics.

Another risk involves bias and fairness. Fact-checking models may unintentionally reinforce biases present in their training data, leading to unfair treatment of certain groups. For example, if specific communities are underrepresented in the evidence base, the models may struggle with claims related to them. Addressing this challenge involves auditing datasets, enhancing data diversity, and incorporating fairness-aware learning methods to reduce bias and ensure equitable performance.

Acknowledgements

This work has been supported by the French government, through the 3IA Côte d'Azur Investments in the IA-cluster project managed by the National Research Agency (ANR) with the reference number ANR-23-IACL-0001. This research has also been supported by the ANR project ATTENTION (ANR-21-CE23-0037).

References

Marah Abdin, Jyoti Aneja, Harkirat Behl, Sébastien Bubeck, Ronen Eldan, Suriya Gunasekar, Michael Harrison, Russell J. Hewett, Mojan Javaheripi, Piero Kauffmann, James R. Lee, Yin Tat Lee, Yuanzhi Li, Weishung Liu, Caio C. T. Mendes, Anh Nguyen, Eric Price, Gustavo de Rosa, Olli Saarikivi, Adil Salim, Shital Shah, Xin Wang, Rachel Ward, Yue Wu, Dingli Yu, Cyril Zhang, and Yi Zhang. 2024. Phi-4 technical report. *Preprint*, arXiv:2412.08905.

Rami Aly, Zhijiang Guo, Michael Sejr Schlichtkrull, James Thorne, Andreas Vlachos, Christos Christodoulopoulos, Oana Cocarascu, and Arpit Mittal. 2021. FEVEROUS: Fact extraction and VERification over unstructured and structured information. In *NeurIPS (Datasets and Benchmarks)*.

Pepa Atanasova, Jakob Grue Simonsen, Christina Lioma, and Isabelle Augenstein. 2020. Generating fact checking explanations. *Preprint*, arXiv:2004.05773.

Payal Bajaj, Daniel Campos, Nick Craswell, Li Deng, Jianfeng Gao, Xiaodong Liu, Rangan Majumder, Andrew McNamara, Bhaskar Mitra, Tri Nguyen, Mir Rosenberg, Xia Song, Alina Stoica, Saurabh Tiwary, and Tong Wang. 2018. Ms marco: A human generated machine reading comprehension dataset. *Preprint*, arXiv:1611.09268.

Jean-Flavien Bussotti, Luca Ragazzi, Giacomo Frisoni, Gianluca Moro, and Paolo Papotti. 2024. Unknown claims: Generation of fact-checking training examples from unstructured and structured data. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, EMNLP 2024, Miami, FL, USA, November 12-16, 2024*, pages 12105–12122. Association for Computational Linguistics.

Yung-Sung Chuang, Linlu Qiu, Cheng-Yu Hsieh, Ranjay Krishna, Yoon Kim, and James Glass. 2024. Lookback lens: Detecting and mitigating contextual hallucinations in large language models using only attention maps. *Preprint*, arXiv:2407.07071.

Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D. Manning. 2019. What does BERT look at? an analysis of BERT's attention. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 276–286, Florence, Italy. Association for Computational Linguistics.

Preetam Prabhu Srikar Dammu, Himanshu Naidu, Mouly Dewan, YoungMin Kim, Tanya Roosta, Aman Chadha, and Chirag Shah. 2024. Claimver: Explainable claim-level verification and evidence attribution of text through knowledge graphs. *Preprint*, arXiv:2403.09724.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, David Esiobu, and Dhruv Choudhary et al. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Julian Martin Eisenschlos, Bhuwan Dhingra, Jannis Bulian, Benjamin Börschinger, and Jordan Boyd-Graber. 2021. Fool me twice: Entailment from wikipedia gamification. *Preprint*, arXiv:2104.04725.

Tianyu Gao, Howard Yen, Jiatong Yu, and Danqi Chen. 2023. Enabling large language models to generate text with citations. *Preprint*, arXiv:2305.14627.

Zhijiang Guo, Michael Sejr Schlichtkrull, and Andreas Vlachos. 2022. A survey on automated fact-checking. *Trans. Assoc. Comput. Linguistics*, 10:178–206.

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

Lei Huang, Xiaocheng Feng, Weitao Ma, Liang Zhao, Yuchun Fan, Weihong Zhong, Dongliang Xu, Qing Yang, Hongtao Liu, and Bing Qin. 2024. Advancing large language model attribution through self-improving. *Preprint*, arXiv:2410.13298.

Sarthak Jain and Byron C. Wallace. 2019. Attention is not explanation. *Preprint*, arXiv:1902.10186.

F. Jelinek, R. L. Mercer, L. R. Bahl, and J. K. Baker. 2005. Perplexity—a measure of the difficulty of speech recognition tasks. *The Journal of the Acoustical Society of America*, 62(S1):S63–S63.

B. Kotipalli. 2024. The role of attention mechanisms in enhancing transparency and interpretability of neural network models in explainable ai. Retrieved from `https://digitalcommons.harrisburgu.edu/dandt/2`.

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

Jinhyuk Lee, Anthony Chen, Zhuyun Dai, Dheeru Dua, Devendra Singh Sachan, Michael Boratko, Yi Luan, Sébastien M. R. Arnold, Vincent Perot, Siddharth Dalmia, Hexiang Hu, Xudong Lin, Panupong Pasupat, Aida Amini, Jeremy R. Cole, Sebastian Riedel, Iftekhar Naim, Ming-Wei Chang, and Kelvin Guu. 2024. Can long-context language models subsume retrieval, rag, sql, and more? *Preprint*, arXiv:2406.13121.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Annual Meeting of the Association for Computational Linguistics*.

Leibo Liu, Oscar Perez-Concha, Anthony Nguyen, Vicki Bennett, and Louisa Jorm. 2022. Hierarchical label-wise attention transformer model for explainable icd coding. *Journal of Biomedical Informatics*, 133:104161.

Shengzhong Liu, Franck Le, Supriyo Chakraborty, and Tarek Abdelzaher. 2021. On exploring attention-based explanation for transformer models in text classification. In *2021 IEEE International Conference on Big Data (Big Data)*, pages 1193–1203.

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

Scott M. Lundberg and Su-In Lee. 2017. A unified approach to interpreting model predictions. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 4768–4777, Red Hook, NY, USA. Curran Associates Inc.

MLOps Community. 2024. Explainable ai: Visualizing attention in transformers. MLOps Community Article. Accessed: 2024-12-13.

Preslav Nakov, David P. A. Corney, Maram Hasanain, Firoj Alam, Tamer Elsayed, Alberto Barrón-Cedeño, Paolo Papotti, Shaden Shaar, and Giovanni Da San Martino. 2021. Automated fact-checking for assisting human fact-checkers. In *IJCAI*, pages 4551–4558. ijcai.org.

Yixin Nie, Adina Williams, Emily Dinan, Mohit Bansal, Jason Weston, and Douwe Kiela. 2020. Adversarial NLI: A new benchmark for natural language understanding. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, and al. 2024. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Danish Pruthi, Mansi Gupta, Bhuwan Dhingra, Graham Neubig, and Zachary C. Lipton. 2020. Learning to deceive with attention-based explanations. *Preprint*, arXiv:1909.07913.

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.

Marco Túlio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. "why should I trust you?": Explaining the predictions of any classifier. In *HLT-NAACL Demos*, pages 97–101. The Association for Computational Linguistics.

Yao Rong, David Scheerer, and Enkelejda Kasneci. 2024. Faithful attention explainer: Verbalizing decisions based on discriminative features. *Preprint*, arXiv:2405.13032.

Mohammed Saeed, Giulio Alfarano, Khai Nguyen, Duc Pham, Raphael Troncy, and Paolo Papotti. 2021. Neural re-rankers for evidence retrieval in the FEVEROUS task. In *Proceedings of the Fourth Workshop on Fact Extraction and VERification (FEVER)*, pages 108–112, Dominican Republic. Association for Computational Linguistics.

Michael Schlichtkrull, Zhijiang Guo, and Andreas Vlachos. 2023. Averitec: A dataset for real-world claim verification with evidence from the web. *Preprint*, arXiv:2305.13117.

Cansu Sen, Thomas Hartvigsen, Biao Yin, Xiangnan Kong, and Elke Rundensteiner. 2020. Human attention maps for text classification: Do humans and neural networks focus on the same words? In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4596–4608, Online. Association for Computational Linguistics.

Jiasheng Si, Yingjie Zhu, and Deyu Zhou. 2023a. Consistent multi-granular rationale extraction for explainable multi-hop fact verification. *CoRR*, abs/2305.09400.

Jiasheng Si, Yingjie Zhu, and Deyu Zhou. 2023b. Exploring faithful rationale for multi-hop fact verification via salience-aware graph learning. In *Proceedings of the Thirty-Seventh AAAI Conference on Artificial Intelligence and Thirty-Fifth Conference on Innovative Applications of Artificial Intelligence and Thirteenth Symposium on Educational Advances in Artificial Intelligence*, AAAI'23/IAAI'23/EAAI'23. AAAI Press.

Xiaobing Sun and Wei Lu. 2020. Understanding attention for text classification. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3418–3428, Online. Association for Computational Linguistics.

Comet Team. 2024. Explainable ai for transformers: How attention mechanisms aid interpretability. Accessed: 2024-12-13.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018. FEVER: a large-scale dataset for fact extraction and verification. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)*, pages 809–819. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Preprint*, arXiv:1706.03762.

Elena Voita, David Talbot, Fedor Moiseev, Rico Sennrich, and Ivan Titov. 2019. Analyzing multi-head self-attention: Specialized heads do the heavy lifting, the rest can be pruned. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5797–5808, Florence, Italy. Association for Computational Linguistics.

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

Sarah Wiegreffe and Yuval Pinter. 2019. Attention is not not explanation. *Preprint*, arXiv:1908.04626.

Jungsun Yoo, Tae Joon Jun, and Young-Hak Kim. 2021. xecgnet: Fine-tuning attention map within convolutional neural network to improve detection and explainability of concurrent cardiac arrhythmias. *Computer Methods and Programs in Biomedicine*, 208:106281.

Ori Yoran, Tomer Wolfson, Ori Ram, and Jonathan Berant. 2024. Making retrieval-augmented language models robust to irrelevant context. *Preprint*, arXiv:2310.01558.

A Appendix

A.1 Features of our system and competitors

We propose Table 2 to compare our system versus other existing ones. Our approach permits lightweight and faithful prediction.

A.2 Alternative configurations for f_e

We mention in the main content that values corresponding to each evidence span e_i are split per layer and per head before being passed to a linear classifier.

Before selecting this approach, we explored several variants:

We denote by `stratKey` the strategy used to operate on evidence e_i, and by `stratQuery` the strategy applied to the overall context.

For `stratKey`, we either compute the average of the attention values over the tokens in e_i (`average`), or we extract the attention value from the previous separator token (`sepToken`) or the following one (`sepTokenAfter`). The latter options may be more or less meaningful depending on the architecture of the model.

For overall context handling, `stratQuery`, we either compute the average over all input tokens, referred to as `averageEverything`, or we compute the average over the tokens corresponding to the claim (`averageClaim`). We also test using only the attention to the last token (`lastToken`).

Using ratio on `stratKey` and `stratQuery` defines our overall `operation` strategy. We also experimented with subtraction instead of ratio.

For example, when using the `average` (for `stratKey`) and `averageEverything` (for `stratQuery`) strategies, we compute, for each evidence span e_i, the average attention weight received by the tokens in e_i from all other tokens in the input. This computation is performed separately for each attention layer and head, resulting in a matrix of shape $[l, h]$ for each e_i. We then compute a reference matrix in the same manner for the claim span. Each evidence matrix is element-wise divided by this reference matrix. The resulting $[l, h]$ matrices are flattened and passed through a linear classifier to produce scalar scores indicating evidence usefulness. We introduce an additional parameter, an advanced evidence loss factor, which multiplies the loss attributed to useful evidence, thereby increasing the importance of correctly labeling them. This is applied exclusively to Phi-4, as other models perform well without it. More details can be found in the code released with the paper.

A.3 Selection of best parameters

We show in Table 3 the parameters we found optimals for ATTUN upon experiments. The results we report used those values for training.

We show in Tables 4 and 5 the impact using other parameters have on the final results on RoBERTa.

Similar behavior is observed on DeBERTa. However, for Phi-4, the only strategies that prove effective are `averageEverything` and `average`.

Due to cost and time constraints, we are unable to provide a comprehensive comparison of all parameter combinations.

A.4 Results on question answering

To extend our analysis on Question Answering, we use the MS MARCO dataset (Bajaj et al., 2018), which consists of over 1 million real search queries from Bing, paired with human-generated answers and supporting web passages, enabling tasks like answer generation and passage ranking. Our goal is to generate an answer from the inputs, and label at the same time the useful passages. The input of our task is constituted from the questions and passages. We choose to use the T5 model (Raffel et al., 2020) for the task. Since the inputs are too long for the model, we limit the context to 4 passages—retaining all useful ones and filling the remainder with randomly selected noisy passages from the same question. Future work should investigate the full 10-passage setting. We fine-tune both the original model and its ATTUN version on this dataset, setting the learning rate to $1e^{-05}$ and training for 3 epochs. We test models on the MS Marco test set, both on the quality of the answer generated with ROUGE (Lin, 2004), BLEU (Papineni et al., 2002) and Perplexity (Jelinek et al., 2005) metrics, and on evidence labeling with the standard classification metrics. Results are displayed in Table 6.

T5 serves as the baseline in this setting, while both T5 and T5+ATTUN represent our contributions. We do not provide results for ATTÆX in this context. Exploring other models for question answering, as well as conducting ablation studies, is left for future work.

Model	ChatGPT	SHAP/LIME	Joint-Model	ATTÆX	ATTUN
Scalable			✓	✓	✓
Low resource constraint		✓	✓	✓	✓
Instant prediction	✓		✓	✓	✓
Privacy / Run on your own		✓	✓	✓	✓
Flexible structure		✓		✓	✓
Context-label tuned	✓		✓		✓
Intrinsic explanation		✓		✓	✓
Impact label prediction	✓		✓		✓
Specific loss adjustment					✓

Table 2: Comparison of models based on various characteristics.

	Dataset	stratQ	stratK.	ope.	γ	mult.C.	adv.Ev.
RoBERTa	Scifact	av.Claim	av.	ratio	1.25	False	1
	Feverous	av.Claim	av.	ratio	0.75	False	1
	FM2	av.Every	av.	ratio	1.5	False	1
	AVeriTeC	av.Claim	av.	ratio	2.5	False	1
DeBERTa	Scifact	av.Claim	av.	ratio	2.5	False	1
	Feverous	av.Claim	av.	ratio	1.15	False	1
	FM2	av.Every	av.	ratio	1	False	1
	AVeriTeC	av.Every	av.	ratio	0.5	False	1
Phi-4	Scifact	av.Every	av.	ratio	0.5	False	1
	Feverous	av.Every	av.	ratio	0.5	False	1
	FM2	av.Every	av.	ratio	0.05	False	1.5
	AVeriTeC	av.Every	av.	ratio	0.25	False	1

Table 3: Configuration settings for the different datasets in the results reported. av. stands for average, stratQ and stratK refer to the strategy for query and key, respectively. adv.Ev. represents the advanced evidence loss. mult.C. corresponds to multiple classifiers for evidence labeling, one for each claim label. mult.C=True trains three binary evidence lassifiers (one per claim label) - we use the classifier corresponding to the predicted label at test time. ope. denotes the operation used, which can either be ratio or subtraction.

From the preliminary experiments, we can see similar results of T5 and T5 & ATTUN for most text evaluation metrics. We observe one notable difference, as ATTUN reduces perplexity by half. This indicates that the model trained with our module generates outputs with greater confidence. We assume that for the same rationale as the higher accuracy on the fact-checking task, the evidence labeling module helps to focus on the right inputs. We think that the noise here is too simple with only 4 evidence. Therefore, there is no impact on BLEU or Rouge.

Concerning evidence labeling, we see that ATTUN permits again to provide explainability, with a F1 Useful of 0.74 and a F1 Noise of 0.95. By contrast, T5 vanilla model is not able to provide explanation.

A.5 Details on datasets used

We select four fact-checking datasets. All examples consist of a claim written by a human, a label, and the golden evidence used to classify the claim. Statistics about the datasets after pre-processing are in Table 7. When the original test set is private, we use the original validation set as test set.

Feverous (Aly et al., 2021) is an extension of Fever (Thorne et al., 2018) with more complex claims. Claims are crafted by humans from textual and tabular evidence from Wikipedia. We linearize tabular evidence in the format: $Cell_{Value}$ <*context*>$Cell_{Header}$</*context*>.

SciFact (Wadden et al., 2020) contains expert-written claims paired with evidence from scientific papers. The evidence is from textual sources only.

FM2 (Eisenschlos et al., 2021) is obtained from an online multiplayer game where users write claims from a list of evidence from Wikipedia. To gain points, claims must be hard to fact-check by other players. Evidence is only textual and the *'Not Enough Information'* label is not present.

AVeriTeC (Schlichtkrull et al., 2023) contains real-world claims to verify with Web evidence. Each claim has evidence in the form of question-answer pairs supported by online content. We treat each pair as one textual evidence. This dataset has a fourth label, *'Conflicting Evidence/Cherry-picking'*, we do not report it in our results as no verifier returns it.

We exclude examples exceeding 512 tokens as they are too long for RoBERTa.

stratQuery			
Dataset	avEverything	avClaim	lastToken
Scifact	95.3	100.0	58.7
FM2	100.0	99.0	-
AVeriTeC	99.7	100	-
Feverous	98.4	100.0	97.3
stratKey			
Dataset	average	sepTok	sepTokAft
Scifact	100.0	98.2	94.7
FM2	100.0	99.5	98.6
AVeriTeC	100.0	99.7	98.9
Feverous	100.0	100.0	98.9
multipleClassifier			
Dataset	True	False	
Scifact	100.0	98.2	
FM2	100.0	99.6	
AVeriTeC	100.0	99.7	
Feverous	100.0	100.0	

Table 4: Performance comparison across different datasets on RoBERTa in terms of accuracy in claim verification when using different parameters for attention operations. Scores are min-max normalized acrossed the compared settings(best=100).

stratQuery			
Dataset	avEverything	avClaim	lastToken
Scifact	99.2	100.0	41.7
FM2	100.0	99.9	-
AVeriTeC	95.6	100.0	-
Feverous	100.0	99.6	50.1
stratKey			
Dataset	average	sepToken	sepTokAft
Scifact	97.8	100.0	62.7
FM2	100.0	99.7	97.7
AVeriTeC	100.0	95.6	96.6
Feverous	100.0	98.5	98.5
multipleClassifier			
Dataset	True	False	
Scifact	97.8	100.0	
FM2	99.9	100.0	
AVeriTeC	97.3	100.0	
Feverous	99.4	100.0	

Table 5: Performance comparison across different datasets on RoBERTa in terms of F1 Useful of evidence labeling when using different parameters for attention operations. Scores are min-max normalized acrossed the compared settings(best=100).

A.6 Retrieval for Noisy Evidence

Datasets come with golden evidence for every claim. However, we must include noisy evidence in the examples to enable our experiments. Whenever possible, we obtain the evidence with retrievers, as this is how they arise in practice. When no corpus is available, we mix the gold evidence with noise. We show an overview of retrieval performances in Table 8. We next detail how we add noise to every data set.

For Feverous, the corpus of Wikipedia pages and the retriever are provided in the pipeline, so we run it on the train and test datasets to obtain their noisy versions. We retrieve 5 documents per claim, and in each document 5 sentences and 3 tables. The retriever selects an arbitrary number of cells per table. Crucially, gold evidence is missing after the retrieval step (0.36 Recall) and a lot of noise is selected in the dataset (0.07 Precision). In the Feverous datasets, recall is lower than 1, i.e., some golden evidence are not picked by the retriever. This may lead the labels of the verifier to be less accurate, as it may lack sufficient information to verify the claim.

SciFact authors included for each example five "distractor abstracts" that cover topics mentioned in the original article. We append sentences from these abstracts to the original evidence, up to a total of 20 sentences.

To write a claim, FM2's players use one to two sentences out of ten sentences from Wikipedia, the remaining sentences are used as noise.

In AVeriTeC, gold evidence are human-created question-answer pairs. The authors provide a question-answer generator to obtain retrieved evidence. From the generator, we pick the least relevant pairs measured by BM25 against the claim. We add an average of 5.5 pairs per example.

A.7 Fact-checking models and their training

For the verifier models, the objective is to infer a label from a claim and the retrieved evidence. Before testing models on the test set, we fine-tune them on the corresponding train set.

GFCE (Atanasova et al., 2020) 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 every dataset. We train FM2, Feverous and SciFact on 3 epochs, and AVeriTeC on 2.

LLaMa 3.1 70B (Dubey et al., 2024) outputs a veracity label as well as a list of the evidence used to make the decision. The prompt used includes the description of the task and the claim with its evidence.

GPT-4o mini (OpenAI et al., 2024) We also report an LLM-prompting verifier based on GPT-4o mini with a similar task. To go further, we report a fine-tuned version of GPT-4o mini on our task. We use 100 examples from the train set of each dataset to fine-tune it.

Table 6: Comparison of model performance on Question Answering on MS Marco with and without attention tuning across various metrics. We put in bold the model obtaining the best results. Explanation here can only be provided with ATTUN.

Model	ROUGE-1	ROUGE-2	ROUGE-L	BLEU	BERTScore F1	Perplexity	F1 Useful	F1 Noise
T5	**0.88**	**0.81**	**0.88**	**0.75**	**0.98**	16	-	-
T5 & ATTUN	**0.88**	0.80	**0.88**	0.74	**0.98**	**7.3**	**0.74**	**0.95**

Dataset	#Claim	T.	|Claim|	#Ev.	#Noise
Feverous	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
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

Table 7: Statistics of datasets used for training and testing the fact-checking models. The number of claim details stand from left to right for *'Supports'*, *'Not Enough Information'*, and *'Refutes'*. AVTC stands for AVeriTeC.

Dataset	Precision	Recall	F1	Sufficient
Feverous	0.07	0.36	0.12	✗
SciFact	0.13	1	0.23	✓
AVeriTeC	0.32	1	0.48	✓
FM2	0.12	1	0.22	✓

Table 8: Performance metrics for the retrievers used to build every dataset (Precision, Recall, F1).

RoBERTa (Liu et al., 2019) is an Encoder only transformer model. We fine-tune the model on relevant NLI datasets (Nie et al., 2020) as in Feverous, with learning rate $1e^{-5}$ for the dataset with 3 labels, and $1e^{-7}$ for the datasets with 2 labels. We run the training for 1 epoch on Feverous and AVeriTeC, and 3 epochs on FM2 and Scifact.

DeBERTa-v3 (He et al., 2023) is a more recent Encoder only transformer model. We also fine-tune the model on relevant NLI datasets (Nie et al., 2020). We use a learning rate of $1e^{-5}$ for the dataset with 3 labels, and $1e^{-7}$ for the datasets with 2 labels. We run the training for 1 epoch on Feverous and AVeriTeC, and 3 epochs on FM2 and Scifact.

Phi-4 mini (Abdin et al., 2024) is a 4B decoder-only transformer model, originally designed for generation tasks rather than classification. To adapt this model for claim classification, we explored two strategies. First, we considered adding a linear classification layer on top of the transformer's hidden states. In this setup, we evaluated several pooling methods for selecting hidden states, including using only the last hidden state, taking the mean across all hidden states, or selecting hidden states corresponding specifically to the tokens that represent the claim. Alternatively, we explored using the model's generative nature directly by calculating classification logits from the probabilities of the tokens associated with each label. We used the second strategies for this paper, as it provides more solid results. A second critical adjustment addresses training stability issues, as fine-tuning Phi-4 mini can exhibit significant instability. To mitigate this, we implemented several enhancements. Specifically, we applied weighted loss functions separately tailored to both claim and evidence classification tasks to manage imbalanced data effectively. To further amplify the impact of minority classes during training, this variant incorporates an advanced evidence loss function designed to increase their contribution to the overall loss. Additionally, we employed careful training settings, including the introduction of warm-up steps, the use of weight decay for regularization, and a linear learning rate scheduler to progressively adjust the learning rate throughout the training process. These adjustments significantly improved the robustness and reliability of the training process. We trained the model using QLoRA with 4-bit quantization and a rank of 16. The training was conducted with a learning rate of $5e-04$. The training was 1 epoch long for Feverous and AVeriTeC, 2 epochs long for FM2, and 3 epochs long for SciFact.

A.8 Configurations of the Post-Hoc Explainers

We use LIME (Ribeiro et al., 2016) and SHAP (Lundberg and Lee, 2017) to analyze the Roberta outputs. We set 500 perturbation samples per explanation.

A.9 Experimental Setup

The experiments were conducted on a system running Ubuntu 20.04.6 LTS with a kernel version of 5.4.0-177-generic. The hardware configuration

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 2: Prompt for LLaMa and GPT-4o mini. `$LABELS_TAG$` is a place holder for the possible labels of the given dataset.

consisted 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 included Python 3.8.10, LIME 0.2.0.1, SHAP 0.45.0. To run 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 consisted of an AMD EPYC 7742 64-Core Processor with 512GB of RAM, and a NVIDIA A100-SXM with 80GB of memory. The software setup included Docker 23.0.3 where we ran in a container LLaMa 3.1 using Python 3.9.

For reproducibility, we run our experiments with a fixed seed of 1234.

A.10 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.

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.

B Controlled ablation on FM2

Appendix B: Controlled Ablation on FM2

We conduct a controlled ablation on FM2 by injecting $k\%$ random distractor sentences into the retriever list for each claim, with $k \in \{10, 25, 50, 100, 150, 200, 300, 500, 700\}$.

At $k = 100\%$, *LLaMA-prompt* slightly outperforms *DeBERTa+ATTUN* in terms of $F1_{\text{Useful}}$ (0.79 vs. 0.78, compared to 0.53 for *DeBERTa+SHAP*). However, as the amount of noise increases, ATTUN shows greater robustness: at $k = 500\%$, ATTUN maintains $F1_{\text{Useful}} \geq 0.69$, while LLaMA-prompt drops to 0.62 and DeBERTa+SHAP falls to 0.42. These relative trends confirm that refining attention makes the verifier less sensitive to noisy context.

The corresponding charts (Figures 3–5) display the evolution of $F1_{\text{Useful}}$, $F1_{\text{Noise}}$, and evidence accuracy across different k values for the three models.

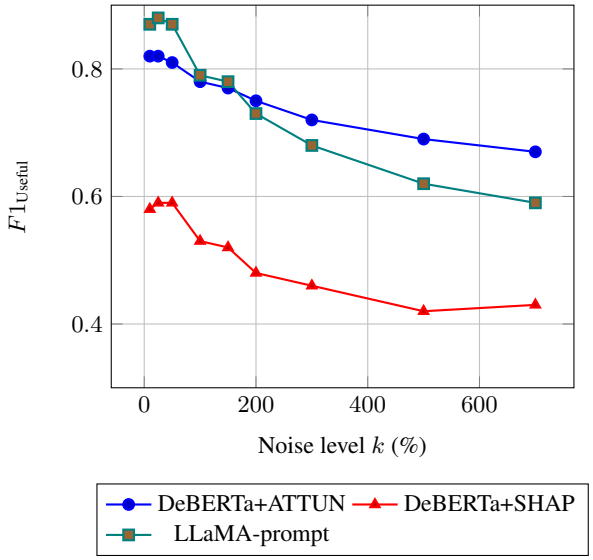

Figure 3: Evolution of $F1_{\text{Useful}}$ with increasing noise levels (k). k = % of additional random distractors.

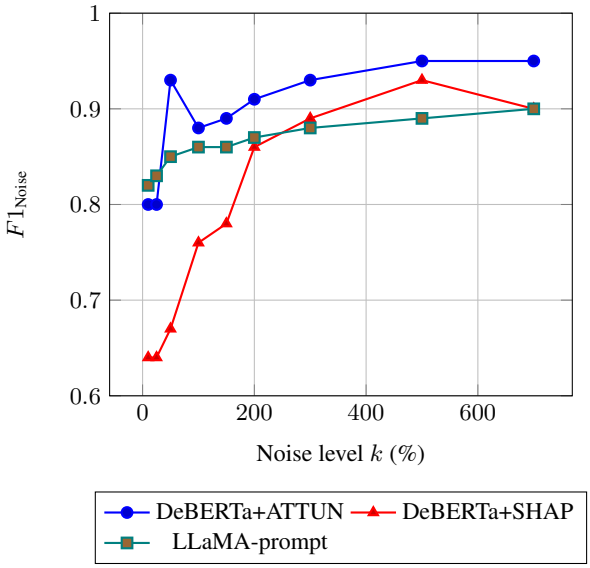

Figure 4: Evolution of $F1_{\text{Noise}}$ with increasing noise levels (k). k = % of additional random distractors.

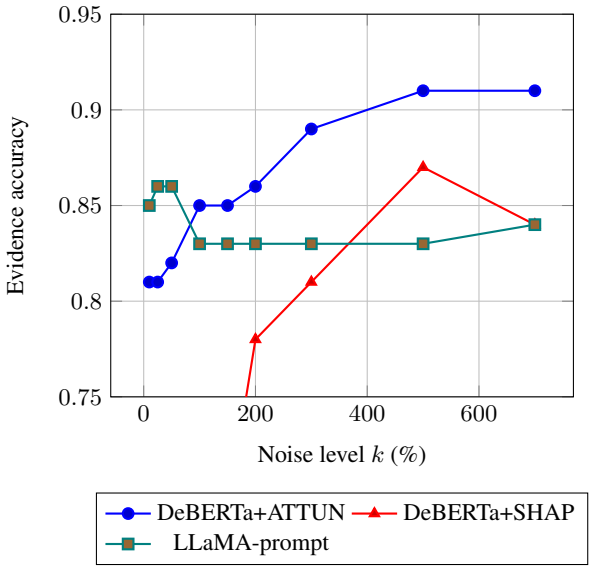

Figure 5: Evolution of evidence accuracy with increasing noise levels (k). k = % of additional random distractors.