Robust Claim Verification Through Fact Detection

Anonymous ACL submission

Abstract

Claim verification can be a challenging task. In this paper, we present a method to enhance the robustness and reasoning capabilities of automated claim verification through the extraction of short facts from evidence. Our novel approach, FactDetect, leverages Large Language Models (LLMs) to generate concise factual statements from evidence and label these facts based on their semantic relevance to the claim and evidence. The generated facts are then combined with the claim and evidence. To train a lightweight supervised model, we incorporate a fact-detection task into the claim verification process as a multitasking approach to improve both performance and explainability. We also show that augmenting FactDetect in the claim verification prompt enhances performance in zero-shot claim verification using LLMs.

Our method demonstrates competitive results in the supervised claim verification model by 15% on the F1 score when evaluated for challenging scientific claim verification datasets. We also demonstrate that FactDetect can be augmented with claim and evidence for zero-shot prompting (AugFactDetect) in LLMs for verdict prediction. We show that AugFactDetect outperforms the baseline with statistical significance on three challenging scientific claim verification datasets with an average of 17.3% performance gain compared to the best performing baselines.

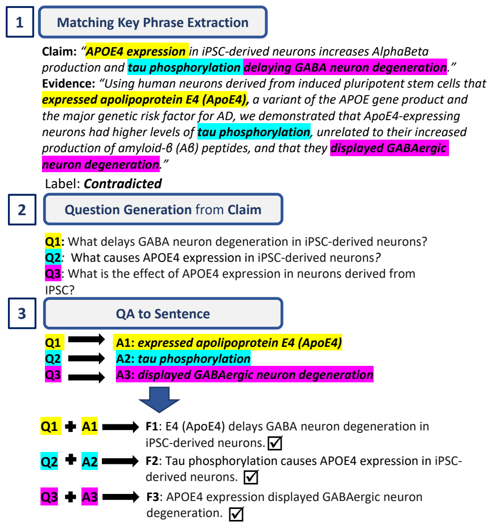

Figure 1: Three-step process of short fact generation from evidence. 1) First we use LLM to generate matching phrases between claim and evidence. 2) Using the extracted phrases from **claim** we design a question generation to generate questions from the claim and the given phrase. 3) The generated matching phrase from **evidence** is concatenated with the question generated from **claim** for short fact generation. Check marks suggest the importance of generated sentences.

1 Introduction

Due to the proliferation of disinformation in many online platforms such as social media, automated claim verification has become an important task in natural language processing (NLP). "Claim verification" refers to predicting the verdict for a claim – is it supported or contradicted by a piece of evidence that has been extracted from a corpus of documents (Thorne et al., 2018; Wadden et al., 2022a; Guo et al., 2022).

Claim verification can be challenging for several reasons. First, the available human-annotated data is limited, resulting in limited performance by current trained models. The task is even harder for scientific claim verification where the claim and the corresponding evidence belong to specific scientific domains, generally requiring specialized knowledge of scientific background, numerical reasoning, and statistics (Wadden et al., 2020). A key challenge in developing automated claim verification systems lies in accurately representing the subtleties of the task. This includes the capacity to change a verdict from 'supported' to change a verdict from 'supported' to 'contradicted' when new evidence in the test set contradicts what was in the training set.

Human-based reasoning for this task involves creating a meaningful link between the claim and the evidence and performing reasoning on such links. A few studies have proposed reasoning methods based on question answering (Liangming Pan, 2021; Dai et al., 2022; Lee et al., 2021), and more recent approaches leverage Large Language Models (LLMs) to generate reasoning programs (Pan et al., 2023) or decompose claims into first-order logic clauses (Wang and Shu, 2023). Question-answering, which involves asking questions about the claim or evidence, retrieving answers from each component, and using these answers for subsequent tasks, is one method used to improve reasoning and explanation in claim verification tasks (Liangming Pan, 2021; Dai et al., 2022). Intuitively, a question asked about a supported or contradicted claim should be *answerable* by the corresponding evidence. The evidence-provided answer can offer critical factual information for veracity prediction.

Motivated by these reasoning approaches, we introduce FactDetect. This short sentence generation framework enhances the state-of-the-art trained models and LLMs by simplifying the connection between claim and evidence pairs by identifying and distilling crucial facts from evidence and then transforming these facts into simpler and concise sentences. We hypothesize that these concise sentences will enhance reasoning abilities by including scientific understanding, simplifying the connection between a claim and its complex scientific evidence, and making a meaningful connection between the claim and the evidence. FactDetect comprises: a) short fact generation b) weakly labeling the short facts based on their importance given the claim; and, c) using these facts in either a multi-task learning-based training of a supervised claim verification model or as an extra step to improve the performance of zero-shot claim-verification using LLMs. An overview of the fact-generation process with an example is given in Figure 1.

We evaluate FactDetect in either multi-task-based finetuning of claim verification models or zero-shot claim verification through LLMs on three scientific claim-verification datasets: SciFact (Wadden et al., 2020), HealthVer (Sarrouti et al., 2021) and Scifact-Open (Wadden et al., 2022a).

In summary, our contributions are: 1) an effective approach for decomposing evidence sentences into shorter sentences. Our method prioritizes relevance to the claim and importance for the verdict, based on the connection between evidence and the claim. 2) FactDetect enhances the performance of supervised claim verification models in the proposed multi-task learning model. 3) augmenting FactDetect generated short sentences for relevant fact detection and claim verification demonstrates state-of-the-art performance in the majority of the LLMs in the few-shot prompting setting. The code and data are available at https://anonymous.4open.science/r/factdetect-0B82/.

2 Background

Automated claim verification means determining the veracity of a claim, typically by retrieving likely relevant documents and searching for evidence within them. The key objective is to ascertain if the evidence either *supports*, *contradicts* or does not have *enough information* to verify the claim. Various datasets have been proposed to facilitate research in this area in different domains: e.g., FEVER (Thorne et al., 2018) is a Wikipedia-based claim verification dataset. Claim verification in the scientific setting has also been proposed in recent years to facilitate research in this complex domain (Wadden et al., 2022a, 2020; Saakyan et al., 2021; Sarrouti et al., 2021; Kotonya and Toni, 2020; Diggelmann et al., 2020). The datasets used for these problems, despite their value, often have limited training data due to the high cost of creation, impacting the reasoning capabilities and robustness of claim verification methods.

In addressing these challenges, the literature shows significant advances in models for verifying scientific claims through reasoning. Prior studies have explored using attention mechanisms to identify key evidence segments (Popat et al., 2017; Cui et al., 2019; Yang et al., 2019; Jolly et al., 2022). Recently, the integration of LLMs in explanation generation has been investigated. For example, ProofVer (Krishna et al., 2022) generates proofs for the claim based on evidence using logic-based inference. ProgramFC (Pan et al., 2023) uses LLMs to generate reasoning programs that can be used to guide fact-checking, and FOLK (Wang and Shu, 2023) leverages the in-context learning ability of LLMs to generate First Order Logic-Guided reasoning over a set of knowledge-grounded question-and-answer pairs to make veracity predictions without using annotated evidence. Other sets of studies attempt to improve this problem through sentence simplification and evidence summarization using LLMs (e.g., (Mehta et al., 2022; Stammbach and Ash, 2020)).

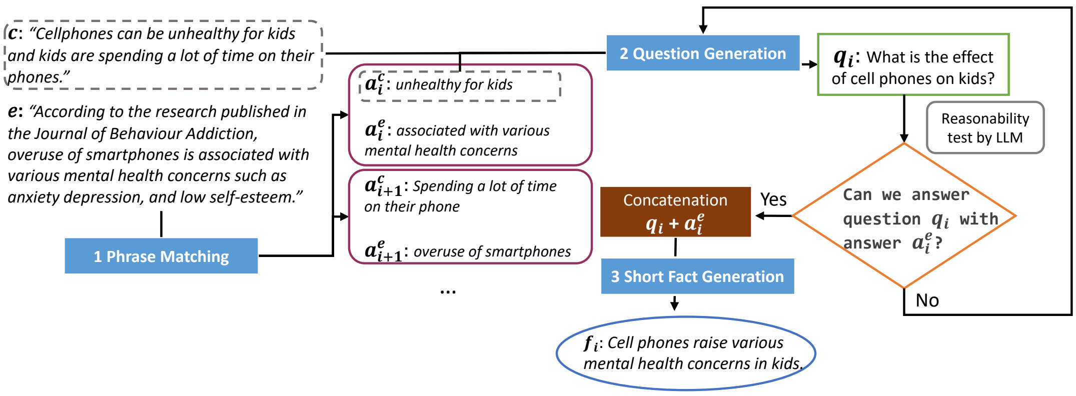

Figure 2: Overview of the proposed framework. FactDetect consists of three steps of 1) Phrase matching, 2) Question generation and finally 3) Short fact generation.

Our work diverges from these methods as we propose an add-on task to enhance the robustness and reasoning ability of existing models. This is achieved through a novel data augmentation strategy which improves the connection between claims and evidence by focusing on learning critical, relevant, and short facts essential for effective scientific claim verification.

3 Methodology

We introduce FactDetect, a novel approach designed to enhance the performance of claim verification solutions by leveraging automatically generated short facts extracted from the evidence. We will show that FactDetect is a versatile tool that can be integrated into various claim verification methods, improving the robustness and reasoning capabilities of existing models. The core of FactDetect relies on weakly-labeled short facts, which are categorized as either *important* for verifying a given claim or *not important* for that purpose, which are used to train a multi-task learning-based model (FactDetect) for importance detection and claim verification.

3.1 Definition

Here, we formally define the primary task of fact generation and labeling: given a claim statement c and corresponding evidence statement e, our objective is to generate concise "facts" from e. We denote this set of facts by $\mathcal{F}_e = \{f_1, \ldots, f_m\}$. Each fact is subsequently labeled as either "important" or "not important," denoted as $y_{f_i} \in \{\mathit{important}, \mathit{not\ important}\}$.

It is important to note that these facts are intentionally designed to be shorter in length compared to the original evidence (e). They serve as distilled pieces of information extracted from the broader context of the evidence. These succinct facts are intended to capture essential details or insights within the evidence, making them more manageable for claim verification tasks. An overview of FactDetect is given in Figure 2. We next elaborate on the processes of short fact generation and weak labeling.

3.2 Short Fact Generation

To generate short facts from the evidence e, we adopt a three-step approach. For these steps, we employ LLM Mistral-7B (Jiang et al., 2023)[1]. We have experimented with different LLMs such as Vicuna-13B (Chiang et al., 2023) and GPT-3.5 and based on our experiments we observed better performance with this open-source LLM. Details of the prompts for each phase of the short fact generation using this approach are given in Appendix A.

1) Phrase matching: Initially, we extract matching phrases from both the claim c and the evidence, treating seeing each phrase as a potential answer to a questions framed around the other ($\mathcal{A} = (a_1^c, a_1^e), \ldots, (a_n^c, a_n^e)$). Phrases "match" if they convey similar meanings and/or are semantically similar. We call these answer pairs. We use an LLM to extract the matching phrases. We do not restrict the LLM to follow specific phrase rules such as n-grams, extracting only entities or noun phrases. This way, we ensure the capture of diverse answer pairs that are more likely to be relevant.

2) Question Generation: After identifying the answer pairs, we formulate concise questions from them. For each answer a_i^c in the pair (a_i^c, a_i^e) with corresponding claim c, we generate a question q_i.

[1]Used following model checkpoint: mistralai/Mistral-7B-Instruct-v0.2

We use c as the context and a_i^c as a desired answer. The question does not use the evidence answer a_i^e to ensure the generated question is directly associated with the claim – because a_i^e is an answer paired with a_i^c, we know that the question drawn from the claim will also be aligned with the evidence answer. We create a question based on these inputs—namely, the *context* and the *answer* we only incorporate the answer from the claim (a_i^c) in this stage and not the answer from evidence (a_i^e). This is to 1) ensure the generation of a high-quality question that can be associated directly with the claim, achievable only by pairing the claim with an internal answer, and 2) incorporate the essential context from the claim into the question, which will later be aligned with the a_i^e for short sentence generations.

3) Short Fact Generation : Finally, We generate short fact sentences by pairing each question q_i with its corresponding evidence-based answer a_i^e which was extracted in the first step and matche3d a_i^c. These questions along with the answers are then converted into full sentences f_i. For example, the previous question and answer results in the sentence *Cellphones cause various mental health concerns for the kids.* We note that not all (q_i, a_i^e) pairs are *reasonable* – i.e., a generated q_i may not align semantically well with the a_i^e due to possible errors during generation or the structure of the context c. Therefore, to ensure a reasonable and useful fact sentence, we further refine these questions and answer pairs by querying the LLM to determine if the (q_i, a_i^e) pair is unreasonable. If the output is "not reasonable," we move forward with other candidates – i.e., (q_{i+1}, a_{i+1}^e) – otherwise, the sentence f_i is added to the candidate answers $\mathcal{A}_c$. This step is crucial because it serves to eliminate most unsuccessful question generations that can occur with LLMs (e.g., the failures can be due to the inconsistent and hallucinated generations) and helps the FactDetect to extract the most important question-answer pairs.

4) Weak labeling Labeling each generated fact as important or not is a crucial step in the FactDetect process. After extracting the candidates in the previous steps, we label a short fact sentence f_i as "important" if the cosine similarity between f_i and the claim c and f_i and evidence e combined to exceed a predefined threshold t and "not important" otherwise. More specifically:

$$sim(f_i, c, e) = \gamma(\cos(f_i, c) + \cos(f_i, e)) \quad (1)$$

$$y_{f_i} = \begin{cases} \text{"important"} & \text{if } sim(f_i, c, e) \geq t \\ \text{"not important"} & \text{otherwise} \end{cases}$$

Here γ is a hyperparameter and $\cos(.)$ is calculated using the Sentence Transformers (Reimers and Gurevych, 2019) embedding of f_i, c and e.

3.3 Joint Claim Verification and Fact Detection Framework

Because of the success of the full context training of claim verification tasks within state-of-the-art models such as MULTIVERS (Wadden et al., 2022b), PARAGRAPHJOINT (Li et al., 2021), and ARSJOINT (Zhang et al., 2021), we propose a similar enhancement approach. Our framework revolves around performing full context predictions by concatenating the claim (c), title of the document in the scientific claim verification datasets (t), gold evidence (e), and all the facts in $\mathcal{F}_e$ with a special separator token to separate each fact in $\mathcal{F}_e$.

The FactDetect approach employs a strategy based on multitasking where the model is jointly trained to minimize a multitask loss:

$$L = L_{cv} + \alpha L_{fact} \quad (2)$$

where L_{cv} represents the cross-entropy loss associated with predicting the overall claim verification task. Specifically, we predict $y(c, e) \in$ {*support*, *contradict*, *nei*} by adding a classification head on the $</s>$ token, where nei refers to Not Enough Info. In addition, L_{fact} denotes the binary cross-entropy loss for predicting whether each fact f_i is important to the claim c or not, and α is a hyperparameter. During inference, we only predict $y(c, e)$, setting aside the fact detection part.

3.4 Zero-shot Claim Verification with LLMs

In the zero-shot approach, without the need for human-annotated training dataset and finetuning a claim verification model, we leverage in-context learning ability of Large Language Models (LLMs) to extract the encoded knowledge in them using a prompting strategy aimed at eliciting the most accurate responses from them. This is done as follows. We augment FactDetect generated short fact sentences $\mathcal{F}_{\rceil}$ into the prompt for claim verification through fact-detection: given c, e and $\mathcal{F}_e$ we first ask an LLM to detect the most important facts and then, by providing an explanation, we ask it to predict the verdict $y(c, e)$.

This approach is similar to the popular Retrieval Augmented Generation (RAG, see e.g. Lewis et al., 2020) approach used in optimizing the output of the Large Language Models using external sources. A difference between our approach to the "retrieval" augmented approach is that we augment the candidate facts from the evidence into the input rather than retrieving any external knowledge.

The approach is formulated as follows: let $\mathcal{M}$ be a language model and $\mathcal{P}$ be the prompt. The $\mathcal{P}$ for the test inputs is generated by concatenating c, e and $\mathcal{F}_e$. We first extract *important facts* and then get the predicted verdict. i.e., $p(y(c,e)|\mathcal{M}(\mathcal{P}))$.

4 Experiments

We evaluate the effect of including FactDetect within different claim verification models and encoders. To evaluate this, we first explain the datasets used and introduce the baseline models we compared to our approach.

4.1 Datasets

SciFact (Wadden et al., 2020) consists of expert annotated scientific claims from biomedical literature with corresponding evidence sentences retrieved from abstracts. *Supported* claims are human-generated using abstract citation sentences, and *Contradicted* claims negate original claims.

SciFact-Open (Wadden et al., 2022a) constitutes a test collection specifically crafted for the assessment of scientific claim verification systems. In addition to the task of verifying claims against evidence within the SciFact domain, this dataset contains evidence originating from a vast scientific corpus of 500,000 documents.

HealthVer (Sarrouti et al., 2021) is a compilation of COVID-19-related claims from real-world scenarios that have been subjected to fact-checking using scientific articles. Unlike most available datasets, where *contradict*ed claims are usually just the negation of the supported ones, in this dataset *contradicted* claims are themselves extracted from real-world claims. The claims in this dataset are more challenging compared to other datasets. More detailed statistics of the datasets are given in Appendix B.

4.2 Baselines

We evaluate FactDetect in supervised and zero-shot settings. In a supervised setting, we either fully or *few-shot* train the state-of-the-art models on the given datasets. For the zero-shot setting, we use several best-performing LLMs and prompt them to predict the verdict based on different baseline prompting strategies. For few-shot supervised training, we train on $k = 45$ training samples.

4.2.1 Supervised Baselines

We incorporate FactDetect as an add-on for a multi-task learning-based approach on two transformer-based encoders. We train the supervised models on NVIDIA RTX8000 GPU and overall model parameters do not exceed 1B. We set the learning rate to $2e-5$ and save the best model in 25 epochs. We choose 0.5 for the γ similarity parameter, in equation (1) and 10 [2] for the α hyperparameter of equation (2). The threshold t for the cosine similarity between fact sentences and claim and evidence is set to 0.6.

Longformer (Beltagy et al., 2020) With the self-attention mechanism incorporated into this model and its ability to process long sequences, we use this encoder to concatenate short sentences into the claim along with additional context provided in the title (if any).

MULTIVERS (Wadden et al., 2022b) is a state-of-the-art supervised scientific claim verification approach which uses Longformer as a base encoder for long-context end-to-end claim verification in a multi-task learning based approach where in addition to the claim and title it incorporates the whole document (abstract) for both claim verification and rationale (evidence) selection. We augment the short sentences extracted by FactDetect into the model as an input and train FactDetect on top of MULTIVERS in a multitasking-based approach.

4.2.2 Zero-shot baselines

LLMs serve as a robust source of knowledge and demonstrate impressive outcomes in various downstream tasks, especially in contexts where zero-shot and few-shot learning are employed. However, the effectiveness of these models heavily depends on the methods used to prompt their responses. Consequently, we evaluate state-of-the-art prompting methods both specific to the claim verification task and general task approaches, and compare them to our novel prompting method based on adding the FactDetect-generated short sentences into the prompt and requiring the LLM to detect the most important sentences for verdict as well as predicting the verdict. We name this prompting strategy

[2] We performed experiments with 5, 10 and 15 and the best performing value was 15.

Setting	Model	HealthVer			SciFact			SciFact-Open		
		F1	P	R	F1	P	R	F1	P	R
Few shot	Longformer	27.8	25.3	30.7	42.4	43.0	41.8	36.2	36.4	36.0
	Longformer + FactDetect	36.9	35.2	38.7	38.3	35.8	42.5	34.3	28.2	43.6
Full	Longformer	53.1	58.1	49.1	54.7	63.5	49.0	40.4	50.2	33.7
	Longformer + FactDetect	53.6	58.2	49.6	56.3	67.2	48.5	43.1	49.7	38.1
	MULTIVERS	60.6	59.1	**62.0**	70.4	**70.8**	70.0	**65.0**	**65.3**	**64.8**
	MULTIVERS + FactDetect	**61.2**	**64.5**	58.2	70.4	70.3	**70.3**	61.1	62.6	59.7

Table 1: Overall performance comparison between different baselines without and with (+FactDetect) multi-task learning incorporating FactDetect. SciFact-Open results are reported in a zero-shot setting. The best results for each dataset are highlighted in bold and the best results within each pair (with and without FactDetect) are underlined.

AugFactDetect. More details of this strategy are given in Appendix C.1. Below are the baseline prompting strategies used to compare with AugFactDetect in the experiments.

Vanilla: We engage LLMs to assess the truthfulness of claims based on provided evidence and to offer justifications for their verdicts. This process is carried out without integrating any extra knowledge or employing a specific strategy.

Chain of Thought (CoT) (Wei et al., 2022) This popular approach involves breaking down the task into a series of logical steps presented to LLMs via prompts for the given context. We use this approach by providing the claim and evidence as input and instructing it to think step by step and provide an explanation before predicting the verdict. We consequently add the *let's think step by step* instruction into the prompt and provide a few shot examples where the verdict is given followed by a step-by-step reasoning explanations. We compare these baseline strategies in FlanT5-XXL (Chung et al., 2022), GPT-3.5 (gpt-3.5-turbo checkpoint),, Llama2-13B (Llama-2-13b-chat-hf checkpoint) (Touvron et al., 2023), Vicuna-13B (Chiang et al., 2023) (vicuna-13b-v1.5 checkpoint), and Mistral-7B Instruct (Mistral-7B-Instruct-v0.2 checkpoint). We perform experiments in few-shot prompting ($k = 5$) for all the strategies. Details of the prompts for Vanilla and CoT are given in Appendix C.

ProgramFC (Pan et al., 2023) is a newly introduced approach that converts complex claims into sub-claims which are then used to generate reasoning programs using LLMs that are executed and used for guiding the verification. We utilize the closed-book setting of this method with N=1. This approach is built for only two-label datasets where claims are either *supported* or *contradicted* by evidence. We used GPT-3.5 to generate programs for ProgramFC and extracted the verification with FlanT5-XL. We experimented with this model in two-label settings (*supported* and *contradicted*) because the original model is designed in binary verification mode. For a fair comparison, we report binary classification results (by excluding the *not enough info* labeled dataset) in all our experiments as well.

4.3 Main Results

4.3.1 Supervised Setup

We first report the results of *supervised* baselines with and without FactDetect incorporated in their training process in Table 1. We experiment with few-shot and full training setups. We observe that incorporating FactDetect into the Longformer encoder achieves the best performance in all three datasets (in bold) in the Full training setup. The average performance gain in F1 when adding FactDetect to Longformer is 3.0% for SciFact. Longformer + FactDetect in the few-shot setting also improves the F1 score for HealthVer by 32.7%. However, we do not see a performance improvement in the few-shot setting for SciFact and SciFact-Open datasets. As mentioned earlier, the results of SciFact-Open dataset are reported in a zero-shot setting (with model trained on SciFact training dataset), resulting in lower performance. Additionally, SciFact-Open receives less benefit from FactDetect than other datasets even in the cases where it does improve results. We suspect that this is due to the more complex nature of the dataset, because it contains claims that are both *supported* and *contradicted* by different evidence sentences. The outcomes are consistent with the top-performing baseline, MULTIVERS. By integrating FactDetect into MULTIVERS, we achieve similar performance, de-

Datasets		SciFact		SciFact-Open		HealthVer	
Metrics		F1	F1 /wo NEI	F1	F1 /wo NEI	F1	F1 /wo NEI
FlanT5-XXL*	Vanilla	75.4	**84.4***	68.5	84.3	50.5	69.1
	CoT	67.9	82.6	68.5	83.2	53.6	62.4
	AugFactDetect	74.5	82.4	73.6	83.4	56.5	69.1
Llama2-13B*	Vanilla	47.7	63.1	47.4	61.0	48.9	67.3
	CoT	55.4	65.7	55.1	71.5	51.5	65.5
	AugFactDetect	75.1	71.7	70.5	76.7	**62.3***	**75.8***
Vicuna-13B*	Vanilla	38.4	67.2	53.5	68.2	51.0	58.7
	CoT	45.3	61.5	52.7	70.9	50.4	62.0
	AugFactDetect	49.1	75.8	50.3	79.5	51.3	71.8
Mistral-7B*	Vanilla	67.3	79.0	62.5	81.8	51.0	73.0
	CoT	70.8	80.3	65.0	83.3	54.2	73.8
	AugFactDetect	**76.0***	82.3	**76.0***	82.4	61.8	73.6
GPT-3.5	Vanilla	64.5	72.5	63.0	80.4	50.9	68.0
	CoT	69.8	81.8	62.9	**84.5***	52.1	67.9
	AugFactDetect	75.4	70.2	71.6	73.1	58.6	64.9
ProgramFC		–	45.0	–	78.0	–	62.9

Table 2: We evaluate the effectiveness of different prompting strategies in 5 LLMs. We report results both with *not enough info* data samples and without them (/wo NEI). For open source LLMs, we ran experiments 5 times and report the average scores (indicated with $*$). The best-performing strategy for each LLM is underlined and overall the best results are highlighted in bold for each dataset. Statistically significant ($p < 0.05$) results compared to the best-performing ones are highlighted with *.

spite the advantage of complete context encoding within this framework.

4.3.2 Zero-shot Setup

The results corresponding to the performance evaluation for the zero-shot prompting with different strategies are reported in Table 2.

We observe that AugFactDetect significantly improves the performance of Llama2-13B, Mistral-7B, and GPT-3.5 in all three datasets compared to the best-performing baseline with an average performance gain of 28.1%, 12.7% and 11.3% in the F1 score for SciFact, Scifact-Open, and Healthver test sets respectively. Similarly, AugFactDetect shows significant improvements for Vicuna-13B in SciFact and HealthVer and FlanT5-XXL with AugFactDetect outperforms other prompting strategies in Scifact-Open and HealthVer test sets. Comparison between ProgramFC and baselines also shows the limited advantage in predicting verdicts in scientific claim verification datasets compared to the general claim verification datasets.

Overall AugFactDetect demonstrates better performance compared to other prompting strategies which suggests the effectiveness of the short fact generation strategy based on the connection between claim and evidence and its performance is comparable to the best-performing baseline in the binary setting.

4.4 Effectiveness of FactDetect

To further understand the impact of the FactDetect, we compare FactDetect based short fact generation approach with the Direct approach where we directly generate short sentences from evidence e (we give 5 examples as few-shot prompting). The details of the promoting strategy and the examples are given in Appendix C.4. We collect the short sentences for each piece of evidence in a claim-evidence (CE) pair, for the SciFact dataset (dev set) and run experiments in the zero-shot setup for 5 LLMS. Macro F1 score comparisons between Direct and AugFactDetect are given in Figure 4. We report results in an average of 5 runs.

Overall, AugFactDetect performs better compared to the Direct approach across 4 out of 5 LLMs with a significant difference in FlanT5-XXL and Mistral-7B. These results suggest the usefulness of the three-step approach compared to the baseline direct sentence generation approach. We hypothesize that one key reason for this is in the Direct approach, the generated sentences are based on the evidence only without making a meaningful connection between the claim and the evidence.

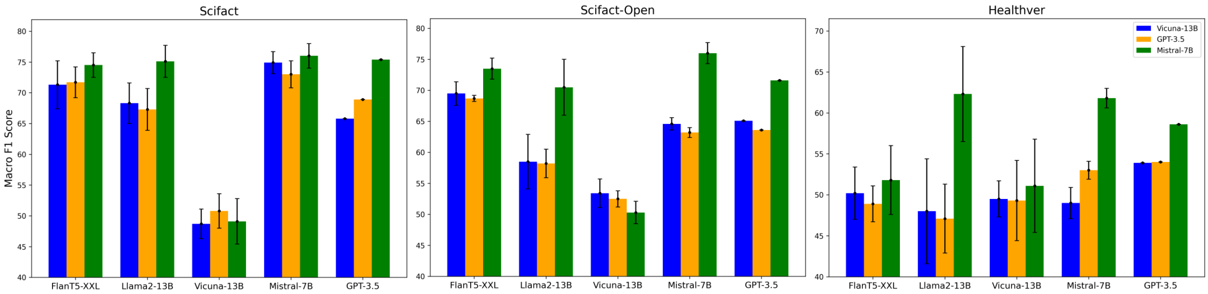

Figure 3: Comparing the F1 Score of zero-shot claim verification task on three test sets when FactDetect is generated with three different LLMs (Vicuna-13B, GPT-3.5 and Mistral-7B).

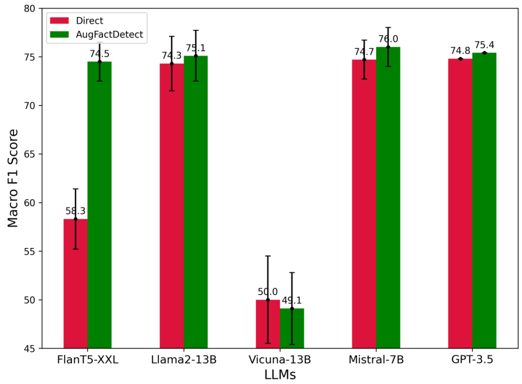

Figure 4: Comparison in Macro F1 score for SciFact between AugFactDetect and Direct.

Therefore, effective short sentences based on the keyphrases linking claim and evidence provide an advantage in predicting the verdict.

4.5 Assessing Generation Quality for FactDetect

Here, we explore the impact of various underlying large language models (LLMs) on the quality of FactDetect generated short sentences. We evaluate this by regenerating short fact sentences using three different LLMs: Mistral-7B[3], GPT-3.5[4], and Vicuna-13B [5] and assess their effect in the performance of AugFactDetect for the claim verification task. The findings are depicted in Figure 3.

The results indicate that choosing Vicuna-13B and GPT-3.5 as the base models for short fact generation demonstrates approximately similar performance across 5 LLMs for all the test sets whereas, Mistral-7B exhibits more pronounced performance. Even though Mistral-7B is a relatively smaller model, shows sufficient and consistent performance gains for the claim verification task whereas, the performance drops with using Vicuna-13B and GPT-3.5 as base models for short fact-generation. This result is independent of the LLM parameter and quality and based on our manual analysis we observed that GPT-3.5 and Vicuna-13B show higher sensitivity to the "reasonability filter" and many question-answer pairs generated in the question generation phase (see 3.2) are marked as not reasonable and do not make it to the next phase of sentence generation resulting in an average low number of generated sentences compared to generated sentences using Mistral-7B with 0.47 and 2.31 for GPT-3.5 and Vicuna-13B compared to 3.64 average number of short sentences per CE pair for Mistral-7B. We additionally perform a human analysis for the overall quality of generated sentences which we detail in Appendix D.

5 Conclusion and Future Work

In this work, we propose FactDetect, an effective short fact generation technique, for comprehensive and high-quality condensed small sentences derived from evidence. With the relevance-based weak-labeling approach this dataset can be augmented to any state-of-the-art claim verification model as a multi-task learning to train fact detection and claim verification. The effectiveness of this model has been demonstrated in both fine-tuned and prompt-based models. Our results suggest that FactDetect incorporated claim-verification task in a zero-shot setting consistently improves performance on average by 17.3% across three challenging scientific claim verification test sets.

FactDetect can have broader applications in different fact-checking and factual consistency evaluation tasks. As a future work, we plan to incorporate FactDetect in the factual consistency evaluation of LLMs. Our preliminary results (see Appendix E) showed promising performance for factuality evaluation in FIB (Tam et al., 2022) dataset.

[3] checkpoint: Mistral-7B-Instruct-v0.2
[4] checkpoint: gpt-3.5-turbo-1106
[5] checkpoint: vicuna-13b-v1.5

6 Limitations

A drawback of our method is the reliance on a generative language model for producing short fact sentences throughout the entire process. Despite employing Mistral-7B, which is among the top open-source LLMs available, the factual accuracy and overall quality of the generated content are bounded by the capabilities of this particular model. Consequently, any inaccuracies from the model could impact the effectiveness of the end-to-end claim verification system.

Furthermore, a limitation of zero-shot FactDetect in real-world claim-verification systems is the need to augment the short sentences into the prompt, which is an additional step and can be time-consuming in the claim verification task. However, this problem is mitigated when we fine-tune a claim-verification system with FactDetect in the training phase, and during inference, we just use the claim and evidence as input.

7 Ethics Statement

Biases. We acknowledge the possibility of bias in generated outputs from the trained LLM. However, this is beyond our control.

Potential Risks. Our approach can be used for automated fact-checking. However, they could also be used by malicious actors to manipulate and attack fact-checking models. A possible future direction is to detect such malicious actions before deployment.

Environmental Impact. Training and using LLMs involves considerable computational resources, including the necessity for GPUs or TPUs during training or inference which can have an impact on the environment. However, we trained our datasets on relatively smaller language models with less than 1B parameters and we used LLMs for inference only which has negligible negative effect on the environment.

References

Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv:2004.05150*.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. 2022. Scaling instruction-finetuned language models. *arXiv preprint arXiv:2210.11416*.

Limeng Cui, Kai Shu, Suhang Wang, Dongwon Lee, and Huan Liu. 2019. Defend: A system for explainable fake news detection. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, CIKM '19, page 2961–2964, New York, NY, USA. Association for Computing Machinery.

Shih-Chieh Dai, Yi-Li Hsu, Aiping Xiong, and Lun-Wei Ku. 2022. Ask to know more: Generating counterfactual explanations for fake claims. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 2800–2810.

Thomas Diggelmann, Jordan Boyd-Graber, Jannis Bulian, Massimiliano Ciaramita, and Markus Leippold. 2020. Climate-fever: A dataset for verification of real-world climate claims.

Zhijiang Guo, Michael Schlichtkrull, and Andreas Vlachos. 2022. A survey on automated fact-checking. *Transactions of the Association for Computational Linguistics*, 10:178–206.

Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. *Advances in neural information processing systems*, 28.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

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

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.

Amrith Krishna, Sebastian Riedel, and Andreas Vlachos. 2022. Proofver: Natural logic theorem proving for fact verification. *Transactions of the Association for Computational Linguistics*, 10:1013–1030.

Minwoo Lee, Seungpil Won, Juae Kim, Hwanhee Lee, Cheoneum Park, and Kyomin Jung. 2021. Crossaug: A contrastive data augmentation method for debiasing fact verification models. In *Proceedings of the 30th ACM International Conference on Information Knowledge Management*, CIKM '21. Association for Computing Machinery.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474.

Xiangci Li, Gully A Burns, and Nanyun Peng. 2021. A paragraph-level multi-task learning model for scientific fact-verification. In *SDU@ AAAI*.

Wenhan Xiong Min-Yen Kan William Yang Wang Liangming Pan, Wenhu Chen. 2021. Zero-shot fact verification by claim generation. In *The Joint Conference of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (ACL-IJCNLP 2021)*, Online.

Sneha Mehta, Huzefa Rangwala, and Naren Ramakrishnan. 2022. Improving zero-shot event extraction via sentence simplification. *arXiv preprint arXiv:2204.02531*.

Shashi Narayan, Shay B Cohen, and Mirella Lapata. 2018. Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. *arXiv preprint arXiv:1808.08745*.

Liangming Pan, Xiaobao Wu, Xinyuan Lu, Anh Tuan Luu, William Yang Wang, Min-Yen Kan, and Preslav Nakov. 2023. Fact-checking complex claims with program-guided reasoning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6981–7004, Toronto, Canada. Association for Computational Linguistics.

Kashyap Popat, Subhabrata Mukherjee, Jannik Strötgen, and Gerhard Weikum. 2017. Where the truth lies: Explaining the credibility of emerging claims on the web and social media. In *Proceedings of the 26th International Conference on World Wide Web Companion*, pages 1003–1012.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Arkadiy Saakyan, Tuhin Chakrabarty, and Smaranda Muresan. 2021. Covid-fact: Fact extraction and verification of real-world claims on covid-19 pandemic. *arXiv preprint arXiv:2106.03794*.

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

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

Derek Tam, Anisha Mascarenhas, Shiyue Zhang, Sarah Kwan, Mohit Bansal, and Colin Raffel. 2022. Evaluating the factual consistency of large language models through summarization. *arXiv preprint arXiv:2211.08412*.

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

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

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.

David Wadden, Kyle Lo, Bailey Kuehl, Arman Cohan, Iz Beltagy, Lucy Lu Wang, and Hannaneh Hajishirzi. 2022a. SciFact-open: Towards open-domain scientific claim verification. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 4719–4734, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

David Wadden, Kyle Lo, Lucy Lu Wang, Arman Cohan, Iz Beltagy, and Hannaneh Hajishirzi. 2022b. MultiVerS: Improving scientific claim verification with weak supervision and full-document context. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 61–76, Seattle, United States. Association for Computational Linguistics.

Haoran Wang and Kai Shu. 2023. Explainable claim verification via knowledge-grounded reasoning with large language models. *arXiv preprint arXiv:2310.05253*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35:24824–24837.

Fan Yang, Shiva K. Pentyala, Sina Mohseni, Mengnan Du, Hao Yuan, Rhema Linder, Eric D. Ragan, Shuiwang Ji, and Xia (Ben) Hu. 2019. Xfake: Explainable fake news detector with visualizations. In *The World Wide Web Conference*, WWW '19, page 3600–3604, New York, NY, USA. Association for Computing Machinery.

Zhiwei Zhang, Jiyi Li, Fumiyo Fukumoto, and Yanming Ye. 2021. Abstract, rationale, stance: a joint model for scientific claim verification. *arXiv preprint arXiv:2110.15116*.

Dataset	Corpus	Train		Dev		Test	
		Claims	CE pairs	Claims	CE pairs	Claims	CE pairs
SciFact-Open	500K	–	–	–	–	279	460
Scifact	14K	809	564	300	209	300	–
HealthVer	322	1393	3340	230	508	230	599

Table 3: Statistics of datasets used in our experiments. Claim Evidence pairs (CE pairs) for each dataset are provided. Scifact test set is not included with gold-labeled evidence sentences therefore the CE pairs are not reported for this dataset.

A Details in Short Fact Generation

A.1 Prompt for Matching Key Phrase Extraction

Figure 5 provides an example of a prompt used for key-phrase extraction.

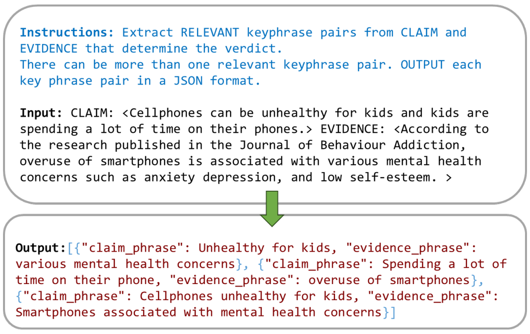

Figure 5: Example of the prompting method used to extract matching key phrases between claim c and evidence e.

A.2 Prompt Strategy for Question Generation

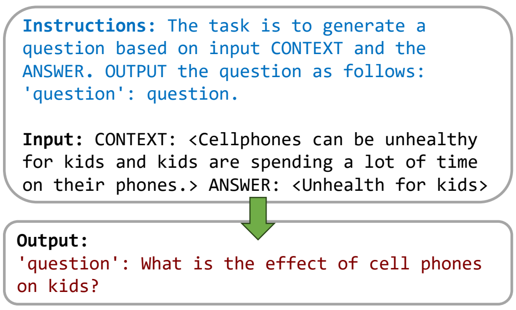

Figure 6: Example of the prompting method used to extract question from a claim c as context and a_i^c as answer.

Figure 6 provides an example of the prompt strategy used to generate a question from extracted phrases from claim and an answer extracted from the previous step. We use a standard question generation prompting method in this step.

A.3 Prompt for Short Fact Generation from Question and Answer

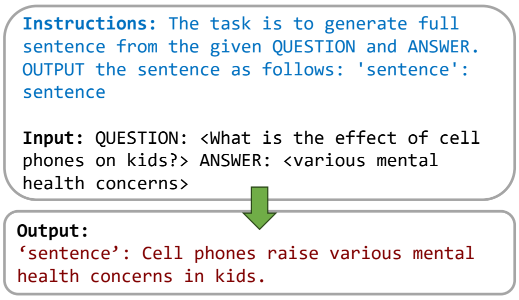

Figure 7: Example of the prompting method used to extract short sentence from a question q_i and a_i^e.

Figure 7 provides an example of the prompting method used to extract the short sentence, final step in short fact generation, from the generated question and matching evidence phrase.

B Dataset statistics

Statistics of the scientific claim verification dataset are given in Table 3.

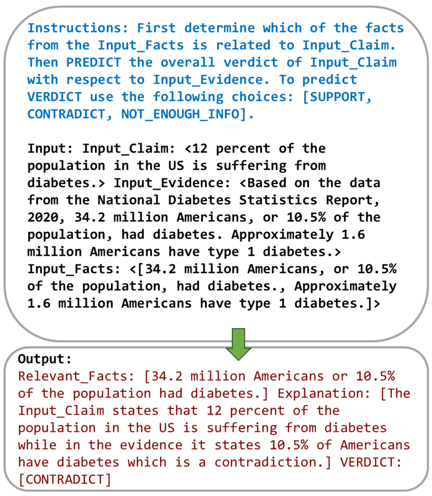

Figure 8: Example of AugFactDetect prompting strategy.

C Details of all the Prompting Strategies used in the experiments

```
Instructions: Predict the verdict
based on the given input Claim and
the Evidence. ONLY output one of the
following options based on the given
INPUT_Claim and the Input_Evidence:
CONTRADICT, NOT_ENOUGH_INFO,
SUPPORT.
Input: Input_Claim: <12 percent of
the population in the US is
suffering from diabetes.>
Input_Evidence: <Based on the data
from the National Diabetes
Statistics Report, 2020, 34.2
million Americans, or 10.5% of the
population, had diabetes.
Approximately 1.6 million Americans
have type 1 diabetes.>
```

```
Output:
VERDICT: [CONTRADICT]
```

Figure 9: Example of Vanilla prompting strategy.

```
Instructions: First think step by step. Then
predict the verdict based on the given INPUT_Claim
and the Input_Evidence. Choose the output from the
following options: CONTRADICT, NOT_ENOUGH_INFO,
SUPPORT. The output format: EXPLANATION:
[rationale] VERDICT: [verdict]

Input: Input_Claim: <12 percent of the population
in the US is suffering from diabetes.>
Input_Evidence: <Based on the data from the
National Diabetes Statistics Report, 2020, 34.2
million Americans, or 10.5% of the population, had
diabetes. Approximately 1.6 million Americans have
type 1 diabetes.>
```

```
Output:
Explanation: [The evidence states that 10.5% of the
population in the US is suffering from diabetes.
This is less than 12 percent of the population in
the US. Therefore, the claim is not supported by
the evidence.] VERDICT: [CONTRADICT]
```

Figure 10: Example of CoT prompting strategy.

C.1 AugFactDetect Prompting Strategy

Figure 8 demonstrates the prompt instructions used in this strategy with an example of input and output. First LLMs are prompted to extract the relevant facts from the input facts and then predict the verdict.

C.2 Vanilla Prompting Strategy

Figure 9 provides an example of the Vanilla prompting method.

C.3 CoT Prompting Strategy

Figure 10 provides an example of the CoT prompting method.

C.4 Direct Prompting Strategy

Figure 11 provides an example of the prompting method used to directly extract the short sentences along with 5 few shot examples concatenated to the prompt.

```
Instructions: The task is to generate a few
shorter more concise sentences from the input
Long Sentence.

Input: Long_Sentence: <Analysis was by
intention-to-treat. Educational level was
chosen as a proxy for social position. The
likelihood of intact perineum, use of upright
position for birth and water birth was also
higher.>
```

```
Output:
'shorter_sentences': [Analysis was by
intention-to-treat., Educational level was
chosen as a proxy for social position., The
likelihood of intact perineum, use of upright
position for birth, and water birth was also
higher.]
```

Figure 11: Example of the prompting method used to directly extract short sentences from evidence.

Base LLM	Support			Contradict		
	F	E	C	F	E	C
Vicuna-13B	73.3	80.0	73.3	**90.0**	73.3	70.0
GPT-3.5	86.3	86.3	**81.8**	70.2	59.0	**86.0**
Mistral-7B	**83.3**	**91.0**	78.1	85.2	**75.8**	84.9

Table 4: Human Evaluation results for 3 different LLM FactDetect generated short facts.

D Human Evaluation of the generated short facts using FactDetect

We conducted an experiment to assess the quality of generated short sentences using a manual human evaluation. we manually evaluated three criteria: 1) **faithfulness** (F), determining if the short sentence is entailed by the evidence, 2) **essentiality** (E), assessing if the generated sentence is crucial for determining the verdict, and 3) **conciseness** (C), evaluating if the sentence is sufficiently brief given the evidence. Each sentence was labeled as yes or no. We randomly sampled 15 supported claim-evidence pairs and 15 contradicted ones, evaluating only the originally labeled "important" short sentences. Each pair could have multiple short sentences, and we reported the average percentage of yes-labeled sentences per pair. The results of this experiment are presented in Table 4. These results show that Mistral-7B generates less concise sentences compared to GPT3.5 whereas it generates more essential sentences. We also see that all the LLMs are at least 70% faithful to the evidence sentences. Overall Mistral-7B generates higher quality short sentences compared to the other LLMs for this task.

E LLM Factuality Evaluation for Document Summarization Through FactDetect

We show that FactDetect is versatile and can be applied to tasks beyond claim verification, such as evaluating the factual consistency of LLM-generated document summaries. To conduct this experiment, we transform the task of evaluating factuality in LLM outputs for document summarization into a claim verification problem. In this setup, the original document serves as evidence, and the summary statement is treated as a claim. We then determine if the statement can be inferred from the document. We then generate short related sentences for the document(evidence) given the statement (claim) using FactDetect and perform experiments similar to the claim verification task. In this setup, the only difference is in the output verdict. Instead of prompting LLM to output one of the *Supported, Contradicted and NEI* verdicts, we prompt it if the statement can be inferred from the given document. The output should be either *Yes* or *No*.

E.1 Factuality Evaluation Dataset

We conduct experiments using the Factual Inconsistency Benchmark (FIB (Tam et al., 2022)) dataset, which includes data from the XSum (Narayan et al., 2018) and CNN/DM (Hermann et al., 2015) document summarization datasets. Each instance in the FIB dataset contains two summaries, one of which is factually consistent. For our experiments on the CNN/DM dataset, we use 457 documents, each paired with two statements, one factually consistent and the other not. We label these pairs as "Yes" for factually consistent and "No" for factually inconsistent, resulting in a total of 914 document-statement pairs.

E.2 Baselines

We compare AugFactDetect with Vanilla, CoT, and Direct prompting methods and report the results for 3 open source LLMs of Flan-T5-XXL, Llama2-13B, and Mistral-7B.

E.3 Metrics

We report results for Macro F1 score, Accuracy, and AUC for this binary classification approach.

Metrics	Prompt	F1	Acc	Auc
FlanT5-XXL	Vanilla	44.1	44.2	57.9
	COT	<u>44.9</u>	<u>44.9</u>	<u>58.7</u>
	Direct	31.6	33.0	53.0
	AugFactDetect	42.6	42.7	53.4
Llama2-13B	Vanilla	19.5	22.9	45.1
	COT	48.6	56.8	49.2
	Direct	54.8	63.8	54.2
	AugFactDetect	**58.9**	**75.0**	<u>58.3</u>
Mistral-7B	Vanilla	31.4	32.2	49.7
	COT	34.3	34.9	52.9
	Direct	38.6	38.6	47.4
	AugFactDetect	<u>46.7</u>	<u>48.7</u>	**63.4**

Table 5: Experimental results for factual consistency evaluation using different prompt strategies. Best performance is highlighted in bold. Underlined values represent best performance for the given LLM.

E.4 Results

The results are reported in Table 5. We observe that best results are achieved when AugFactDetect is used as prompting method for factual consistency evaluation. Overall decomposing the document into smaller sentences seems to be useful for factual consistency detection and using FactDetect for this task shows superior performance which suggest the effectiveness of FactDetect and its applications beyond the claim verification task.

Strategy	Input	Generated Output
Vanilla	**Claim:** Blocking the interaction between TDP-43 and respiratory complex I proteins ND3 and ND6 leads to increased TDP-43-induced neuronal loss. **Evidence:** The suppression of TDP-43 mitochondrial localization abolishes WT and mutant TDP-43-induced mitochondrial dysfunction and neuronal loss, and improves phenotypes of transgenic mutant TDP-43 mice	**Verdict**: Supported **Ground Truth:** Contradicted
CoT	**Claim:** 1/2000 in UK have abnormal PrP positivity. **Evidence:** RESULTS Of the 32,441 appendix samples 16 were positive for abnormal PrP, indicating an overall prevalence of 493 per million population (95% confidence interval 282 to 801 per million) *Let's Think Step by Step*	**Explanation:** The evidence states that an overall prevalence of 493 per million population. This is a counterfactual statement that contradicts with the claim that 1/2000 in UK have abnormal PrP positivity. **Verdict:** Contradicted **Ground Truth:** Supported
FactDetect	**Claim:** Angiotensin converting enzyme inhibitors are associated with increased risk for functional renal insufficiency. **Evidence:** Although ACE inhibitor therapy usually improves renal blood flow (RBF) and sodium excretion rates in CHF and reduces the rate of progressive renal injury in chronic renal disease, its use can also be associated with a syndrome of "functional renal insufficiency" and/or hyperkalemia. Understanding the pathophysiological mechanisms and the common risk factors for ACE inhibitor–induced functional ARF is critical, because preventive strategies for ARF exist, and if effectively used, they may permit use of these compounds in a less restricted fashion. **Facts:** 1-angiotensin converting enzyme inhibitors can be associated with functional renal insufficiency and or hyperkalemia. 2- angiotensin converting enzyme inhibitors are linked to functional renal insufficiency through various pathophysiological mechanisms and common risk factors. preventive strategies are crucial in understanding these connections to mitigate the risk.	**Relevant Facts:** [angiotensin converting enzyme inhibitors can be associated with functional renal insufficiency and or hyperkalemia., angiotensin converting enzyme inhibitors are linked to functional renal insufficiency through various pathophysiological mechanisms and common risk factors. preventive strategies are crucial in understanding these connections to mitigate the risk.] **Explanation:** he input claim states that angiotensin converting enzyme inhibitors are associated with increased risk for functional renal insufficiency, and the evidence provides relevant supporting facts that ACE inhibitor therapy can be associated with a syndrome of "functional renal insufficiency" and/or hyperkalemia.. **Verdict**: Supported **Ground Truth:** Supported

Table 6: Example prompts used for extracting predictions from GPT-3.5 and their corresponding outputs. The examples are drawn from SciFact dev set.

Example	Direct	FactDetect
Claim: Antiretroviral therapy reduces rates of tuberculosis across a broad range of CD4 strata. **Evidence:** Antiretroviral therapy is strongly associated with a reduction in the incidence of tuberculosis in all baseline CD4 count categories: (1) less than 200 cells/µl (hazard ratio [HR] 0.35, 95% CI 0.28 to 0.44). CONCLUSIONS Antiretroviral therapy is strongly associated with a reduction in the incidence of tuberculosis across all CD4 count strata.	• antiretroviral therapy is strongly associated with a reduction in the incidence of tuberculosis. • 95% confidence interval ci 0.07 to 0.36) • 200 to 350 cells/µl (hr 0.34 • antiretroviral therapy is strongly associated with a reduction in the incidence of tuberculosis in all baseline cd4 count categories: less than 200 cells/µl (hazard ratio hr 0.16" • conclusions: antiretroviral therapy is strongly associated with a reduction in the incidence of tuberculosis across all cd4 count strata.	• antiretroviral therapy reduces the incidence of tuberculosis in all baseline cd4 count categories less than 200 cells/µl, with a hazard ratio of 0.16 (95% ci: 0.07-0.36)
Claim: Autologous transplantation of mesenchymal stem cells causes a higher rate of opportunistic infections than induction therapy with anti-interleukin-2 receptor antibodies. **Evidence:** Also, during the 1-year follow-up, combined analysis of MSC-treated groups revealed a significantly decreased risk of opportunistic infections than the control group (hazard ratio, 0.42; 95% CI, 0.20-0.85, P=.02) ... and better estimated renal function at 1 year.	• during the 1-year follow-up combined analysis of msc-treated groups revealed significantly decreased risk of opportunistic infections than the control group. • conclusion among patients undergoing renal transplant • the use of autologous mscs compared with anti-il-2 receptor antibody induction therapy resulted in lower incidence of acute rejection • decreased risk of opportunistic infection and better estimated renal function at 1 year.	• autologous transplantation of mesenchymal stem cells increases the risk of opportunistic infections compared to induction therapy with anti-interleukin-2 receptor antibodies, as shown by a significantly decreased risk of opportunistic infections in the control group compared to the msct-treated groups. • autologous transplantation of mesenchymal stem cells increases the risk of opportunistic infections compared to induction therapy with anti-interleukin-2 receptor antibodies by a hazard ratio of 0.42 (95% ci 0.20-0.85), with a p-value of 0.02.

Table 7: Example of the FactDetect generated short facts and Direct approach generated short facts for 2 examples from SciFact Dev set.