information

MDPI

Article

Generating Fluent Fact Checking Explanations with Unsupervised Post-Editing

Shailza Jolly [1,*,†], Pepa Atanasova [2] and Isabelle Augenstein [2]

[1] Amazon Alexa AI, 10117 Berlin, Germany
[2] Department of Computer Science, University of Copenhagen, 1050 Copenhagen, Denmark
* Correspondene: shailzj@amazon.com
† Work was done prior to joining Amazon.

Abstract: Fact-checking systems have become important tools to verify fake and misguiding news. These systems become more trustworthy when human-readable explanations accompany the veracity labels. However, manual collection of these explanations is expensive and time-consuming. Recent work has used extractive summarization to select a sufficient subset of the most important facts from the ruling comments (RCs) of a professional journalist to obtain fact-checking explanations. However, these explanations lack fluency and sentence coherence. In this work, we present an iterative edit-based algorithm that uses only phrase-level edits to perform unsupervised post-editing of disconnected RCs. To regulate our editing algorithm, we use a scoring function with components including fluency and semantic preservation. In addition, we show the applicability of our approach in a completely unsupervised setting. We experiment with two benchmark datasets, namely LIAR-PLUS and PubHealth. We show that our model generates explanations that are fluent, readable, non-redundant, and cover important information for the fact check.

Keywords: natural language generation; fact-checking; explainable AI

Citation: Jolly, S.; Atanasova, P.; Augenstein, I. Generating Fluent Fact Checking Explanations with Unsupervised Post-Editing. *Information* **2022**, *13*, 500. https://doi.org/10.3390/info13100500

Academic Editors: Gabriele Gianini and Pierre-Edouard Portier

Received: 29 August 2022
Accepted: 12 October 2022
Published: 17 October 2022

1. Introduction

In today's era of social media, the spread of news is a click away, regardless of if it is fake or real. However, the quick propagation of fake news has repercussions on peoples' lives. To alleviate these consequences, independent teams of professional fact checkers manually verify the veracity and credibility of news, which is time and labor-intensive, making the process expensive and less scalable. Therefore, the need for accurate, scalable, and explainable automatic fact-checking systems is inevitable [1].

Current automatic fact-checking systems perform veracity prediction for given claims based on evidence documents (Thorne et al. [2], Augenstein et al. [3], inter alia), or based on long lists of supporting ruling comments (RCs, Wang [4], Alhindi et al. [5]). RCs are in-depth explanations for predicted veracity labels, but they are challenging to read and not useful as explanations for human readers due to their sizable content.

Recent work [6,7] has thus proposed to use automatic summarization to select a subset of sentences from long RCs and used them as short layman explanations. However, using a purely extractive approach [6] means sentences are cherry-picked from different parts of the corresponding RCs, and as a result, explanations are often disjoint and non-fluent.

While a Seq2Seq model trained on parallel data can partially alleviate these problems, as Kotonya and Toni [7] propose, it is an expensive affair in terms of the large amount of data and compute required to train these models. Therefore, in this work, we focus on unsupervised post-editing of explanations extracted from RCs. In recent studies, researchers have addressed unsupervised post-editing to generate paraphrases [8] and sentence simplifications [9]. However, they use small single sentences and perform exhaustive word-level or a combination of word and phrase-level edits, which has limited

applicability for longer text inputs with multiple sentences, e.g., veracity explanations, due to prohibitive convergence times.

Hence, we present a *novel iterative edit-based algorithm* that performs three edit operations (insertion, deletion, reorder), all at the phrase level. Figure 1 presents a qualitative example from the PubHealth dataset [7], which illustrates how each post-editing step contributes to creating explanations that are more readable, fluent, and creates a coherent story, while also preserving the information important for the fact check.

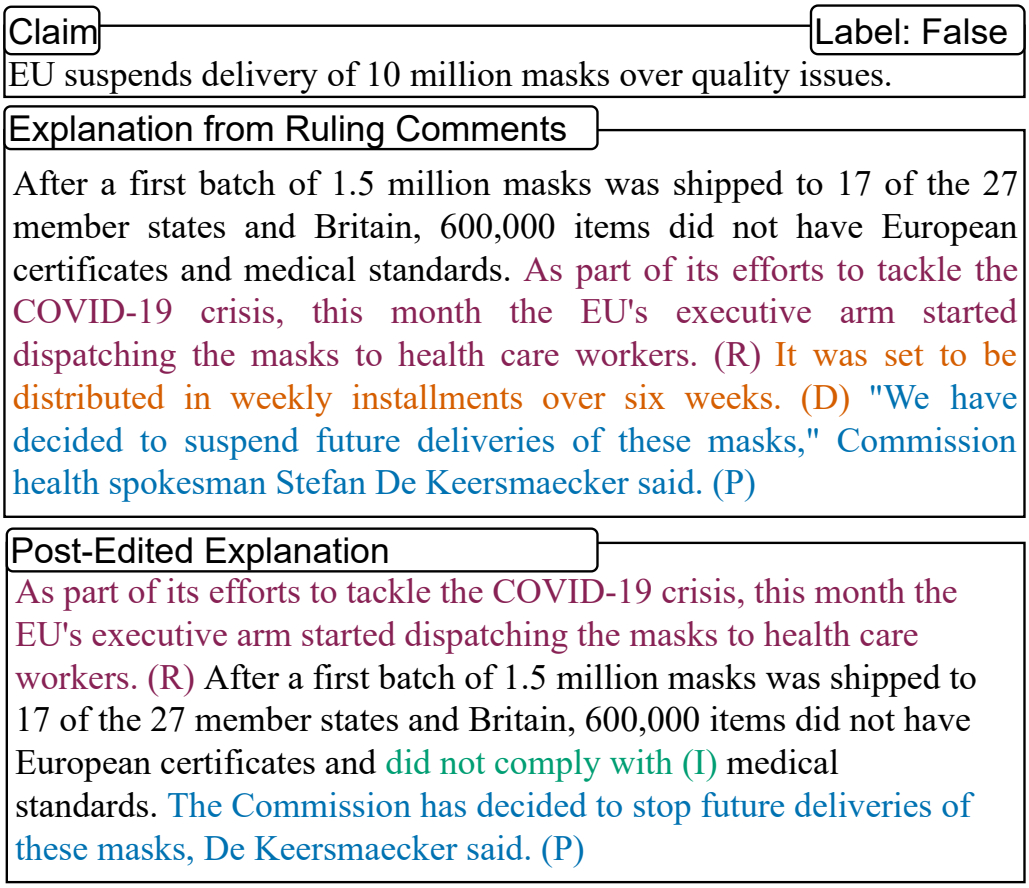

Figure 1. Example of a post-edited explanation from PubHealth that was initially extracted from ruling comments. We illustrate four post-editing steps: insertion (I), reordering (R), deletion (D), and paraphrasing (P).

Our proposed method finds the best post-edited explanation candidate according to a scoring function, ensuring the quality of explanations in fluency, semantic similarity, and semantic preservation. To ensure that the sentences are grammatically correct, we also perform grammar checking of the candidate explanations. As a second step, we apply paraphrasing to further improve the conciseness and human readability of the explanations.

In summary, our main contributions include:

- To the best of our knowledge, this work is the first to explore an iterative unsupervised edit-based algorithm using only phrase-level edits. The proposed algorithm also leads to the first computationally feasible solutions for unsupervised post-editing of long text inputs, such as veracity ruling comments.
- We show how combining an iterative algorithm with grammatical corrections, and paraphrasing-based post-processing leads to fluent and easy-to-read explanations.
- We conduct extensive experiments on the LIAR-PLUS [4] and PubHealth [7] fact-checking datasets. Our manual evaluation confirms that our approach improves the fluency and conciseness of explanations.

2. Related Work

The most closely related streams of approaches to our work are explainable fact checking, generative approaches to explainability and post-editing for language generation.

2.1. Explainable Fact Checking

Recent work has produced fact-checking explanations by highlighting words in tweets using neural attention [10]. However, their explanations are used only to evaluate and compare the proposed model with other baselines without neural attention. Wu et al. [11] propose to model evidence documents with decision trees, which are inherently interpretable ML models. In a recent study, Atanasova et al. [6] present a multi-task approach to generate free-text explanations for political claims jointly with predicting the veracity of claims. They formulate an extractive summarization task to select a few important sentences from a long fact-checking report. Atanasova et al. [12] also perform extractive explanation generation guided by a set of diagnostic properties of explanations and evaluate on the FEVER [2] fact-checking dataset, where explanation sentences have to be extracted from Wikipedia documents.

In the domain of public health claims, Kotonya and Toni [7] propose to generate explanations separately from the task of veracity prediction. Mishra et al. [13] generate summaries of evidence documents from the Web using an attention-based mechanism. They show that their summaries perform better than using the original evidence documents directly. Similarly to Atanasova et al. [6], Kotonya and Toni [7], we present a generative approach for creating fact-checking explanations. In contrast to related work, we propose an unsupervised post-editing approach to improve the fluency and readability of previously extracted fact-checking explanations.

2.2. Generative Approaches to Explainability

Explainable AI [14] is important to encourage trust of blackbox model's decisions and increase their acceptability among users. While most work on explanation generation propose methods to highlight portions of inputs (Camburu et al. [15], DeYoung et al. [16], inter alia), some work focuses on generative approaches to explainability. Ref Camburu et al. [15] propose combining an explanation generation and a target prediction model in a pipeline or a joint model for Natural Language Inference with abstractive explanations about the entailment of two sentences. They find that first explaining and then predicting based on the explanation achieves better trust as the prediction is based on the right reasons. Stammbach and Ash [17] propose few-shot training for the GPT-3 [18] model to explain a fact check from retrieved evidence snippets. GPT-3, however, is a limited-access model with high computational costs. As in our work, Kotonya and Toni [7] first extract evidence sentences, which are then summarised by an abstractive summarisation model. The latter is trained on the PubHealth dataset. In contrast, we are the first to focus on unsupervised post-editing of explanations produced using automatic summarization.

2.3. Post-Editing for Language Generation

Previous work has addressed unsupervised post-editing for multiple tasks such as paraphrase generation [8], sentence simplification [9] or sentence summarization [19]. However, all these tasks handle shorter inputs in comparison to the long multi-sentence extractive explanations that we have. Furthermore, they perform exhaustive edit operations at the word level and sometimes additionally at the phrase level, both of which increase computation and inference complexity. Therefore, we present a novel approach that performs a fixed number of edits only at the phrase level followed by grammar correction and paraphrasing.

3. Method

Our method is comprised of two steps. First, we select sentences from RCs that serve as extractive explanations for verifying claims (Section 3.1). We then apply a post-editing

algorithm on the extractive explanations in order to improve their fluency and coherence (Section 3.2).

3.1. Selecting Sentences for Post-Editing

Supervised Selection. To produce supervised extractive explanations, we use the method implemented by Atanasova et al. [20] for the LIAR-PLUS dataset. We then adapt the supervised method for the PubHealth dataset using the same pre-trained model as used by Kotonya and Toni [7] for the dataset. The models used for the extractive explanations are based on DistilBERT [21] for LIAR-PLUS, and SciBERT [22] for PubHealth, which allows for direct comparison with Kotonya and Toni [7], Atanasova et al. [20].

We supervise explanation generation by k greedily selected sentences from each claim's RCs that achieve the highest ROUGE-2 F1 score when compared to the gold justification. We choose $k = 4$ for LIAR-PLUS and $k = 3$ for PubHealth, the average number of sentences in the veracity justifications in the corresponding datasets. The selected sentences are positive gold labels, $\mathbf{y}^E \in \{0,1\}^N$, where N is the number of sentences in the RCs. We also use the veracity labels $\mathbf{y}^F \in Y_F$ for supervision.

Following Atanasova et al. [20], we then learn a multi-task model $g(X) = (\mathbf{p}^E, \mathbf{p}^F)$. Given the input X, comprised of a claim and the RCs, it predicts jointly the veracity explanation $\mathbf{p}^E$ and the veracity label $\mathbf{p}^F$, where $\mathbf{p}^E \in \mathbb{R}^{1,N}$ selects sentences for explanation, i.e., {0,1}, and $\mathbf{p}^F \in \mathbb{R}^m$, with $m = 6$ for LIAR-PLUS, and $m = 4$ for PubHealth. Finally, we optimise the joint cross-entropy loss function $\mathcal{L}_{MT} = \mathcal{H}(\mathbf{p}^E, \mathbf{y}^E) + \mathcal{H}(\mathbf{p}^F, \mathbf{y}^F)$.

Unsupervised selection. We also experiment with unsupervised selection of sentences to test the possibility to construct fluent fact-checking explanations in an entirely unsupervised way. We use a Longformer [23] model, which was introduced for tasks with longer input, instead of the sliding-window approach also used in Atanasova et al. [20], which is without cross-window attention. We train a model $h(X) = \mathbf{p}^F$ to predict the veracity of a claim. We optimise a cross-entropy loss function $\mathcal{L}_F = \mathcal{H}(\mathbf{p}^F, \mathbf{y}^F)$ and select k sentences $\mathbf{p}^{E'} \in \mathbb{R}^{1,N}$, {0, 1}, with the highest saliency scores. The saliency score of a sentence is the sum of the saliency scores of its tokens. The saliency of a token is the gradient of the input token w.r.t. the output [24]. We selected sentences using the raw gradients as Atanasova et al. [25] show that different gradient-based methods yield similar results. As the selection could be noisy [26], we consider these experiments as only complementary to the main supervised results.

3.2. Post-Editing

Our post-editing is completely unsupervised and operates on sentences obtained in Section 3.1. It is a search algorithm that evaluates the candidate sequence $\mathbf{p}^C$ for a given input sequence, where the input sequence is either $\mathbf{p}^E$ for supervised selection or $\mathbf{p}^{E'}$ for unsupervised selection. Below, we use $\mathbf{p}^E$ as a representative of both $\mathbf{p}^E$ and $\mathbf{p}^{E'}$.

Given $\mathbf{p}^E$, we iteratively generate multiple candidates by performing phrase-level edits as defined in Section 3.2.1. To evaluate a candidate explanation, we define a scoring function, which is a product of multiple scorers, also known as a product-of-experts model [27]. Our scoring function includes fluency and semantic preservation, and controls the length of the candidate explanation (Section 3.2.2). We repeat the process for n steps and select the last best-scoring candidate as our final output. We then use grammar correction (Section 3.2.4) and paraphrasing (Section 3.2.5) to further ensure conciseness and human readability.

3.2.1. Candidate Sequence Generation

We generate candidate sequences by phrase-level edits. We use the off-the-shelf syntactic parser from CoreNLP [28] to obtain the constituency tree of a candidate sequence $\mathbf{p}^C$. As $\mathbf{p}^C$ is long, we perform all operations at the phrase level. At each step t, our algorithm first randomly picks one operation—insertion, deletion, or reordering, and then randomly selects a phrase. For **insertion**, our algorithm inserts a <MASK> token before the randomly selected phrase, and use RoBERTa to evaluate the posterior probability of a

candidate word [29]. This functionality allows us to leverage the pre-training capabilities of RoBERTa and inserts high-quality words that support the context of the overall explanation. Furthermore, inserting a <MASK> token before a phrase prevents breaking other phrases within the explanation, thus preserving their fluency.

The **deletion** operation deletes the randomly selected phrase, For the **reorder** operation we randomly select one phrase, which we call *reorder phrase*, and randomly select m phrases, which we call *anchor phrases*. We **reorder** each *anchor phrase* with a *reorder phrase* and obtain m candidate sequences. We feed these candidates to GPT2 and select the most fluent candidate based on the fluency score given by Equation (1).

3.2.2. Scoring Functions

The scoring functions employed for our post-editing algorithm rely on pre-trained models, such as RoBERTa [30] for semantic preservation, and GPT-2 [31] for fluency preservation. Similar to our approach, most contemporary natural language processing methods rely on pre-trained models. Related work also uses pre-trained models to improve fluency and semantic similarity [9,32,33].

The **fluency score** (f_{flu}) measures the language fluency of a candidate sequence. We use pre-trained GPT2 model [31]. We use the joint likelihood of candidate $\mathbf{p}^C$:

$$f_{flu}(\mathbf{p}^C) = \prod_{i=1}^{n} P(\mathbf{p}_i^C | \mathbf{p}_1^C, ..., \mathbf{p}_{i-1}^C) \tag{1}$$

For **semantic preservation**, we compute similarities at both word and explanation level between our source explanation ($\mathbf{p}^E$) and candidate sequence ($\mathbf{p}^C$) at time-step t. The word-level semantic scorer evaluates the preserved amount of keyword information in the candidate sequence. Similarly to Li et al. [29], we use RoBERTa (R) [30], a pre-trained masked language model, to compute a contextual representation of the ith word in an explanation as $\mathrm{R}(\mathbf{p}_i^E, \mathbf{p}^E)$. Here, $\mathbf{p}^E = (\mathbf{p}_1^E \dots \mathbf{p}_m^E)$ is an input sequence of words. We then extract keywords from $\mathbf{p}^E$ using Rake [34] and compute a **word-level semantic similarity score**:

$$f_w(\mathbf{p}^E, \mathbf{p}^C) = \min_{k \in kw(\mathbf{p}^E)} \max_{\mathbf{p}_i^C \in \mathbf{p}^C} R(k, \mathbf{p}^E)^\intercal R(\mathbf{p}_i^C, \mathbf{p}^C) \tag{2}$$

which is the lowest cosine similarity among all keywords i.e., the least matched keyword of $\mathbf{p}^E$.

The **explanation-level semantic preservation scorer** evaluates the cosine similarity of two explanation vectors:

$$f_e(\mathbf{p}^E, \mathbf{p}^C) = \frac{(\mathbf{p}^C)^\intercal \mathbf{p}^E}{||\mathbf{p}^C|| ||\mathbf{p}^E||} \tag{3}$$

We use SBERT [35] for obtaining embeddings for both $\mathbf{p}^E$, $\mathbf{p}^C$. Our overall semantic score is the product of the word level and the explanation-level semantics scores:

$$f_{sem}(\mathbf{p}^E, \mathbf{p}^C) = f_w(\mathbf{p}^E, \mathbf{p}^C)^\beta . f_e(\mathbf{p}^E, \mathbf{p}^C)^\eta \tag{4}$$

where β, and η are hyperparameter weights for the separate scores.

Length score (f_{len}) This score encourages the generation of shorter sentences. It is proportional to the inverse of the sequence length, i.e., the higher the length of a candidate sentence, the lower its score. To control over-shortening, we reject explanations with fewer than 40 tokens.

Named entity (NE) score (f_{ent}) This score is a proxy for meaning preservation, since NEs hold the key information within a sentence. We first identify NEs using an off-the-shelf entity tagger (https://spacy.io/, accessed on 3 February 2021) and then count their number in a given explanation.

Overall scoring Our overall scoring function is the product of individual scores:

$$f_(\mathbf{p}^C) = f_{flu}(\mathbf{p}^C)^\alpha . f_{sem}(\mathbf{p}^E, \mathbf{p}^C) . f_{len}(\mathbf{p}^C)^\gamma . f_{ent}(\mathbf{p}^C)^\delta \tag{5}$$

where α, γ, and δ are hyperparameter weights for the different scores.

3.2.3. Iterative Edit-Based Algorithm

Given input explanations, our algorithm iteratively performs edit operations for n steps to search for a highly scored candidate ($\mathbf{p}^C$). At each search step, it computes scores for the previous sequence ($\mathbf{p}^{C-1}$) and candidate sequence using Equation (5). It selects a candidate sequence if its score is larger than the previous one by a multiplicative factor r_{op}:

$$f_{\mathbf{p}^C}/f_{\mathbf{p}^{C-1}} > r_{op} \tag{6}$$

For each edit operation, we use a separate threshold value r_{op}. r_{op} allows controlling specific operations, as for the reorder operation, if $\mathbf{p}^C$ gets a lower score than $\mathbf{p}^{C-1}$ then a lower value of r_{op} will enable selection of $\mathbf{p}^C$. In particular, it controls the exploration vs. the overall score of the selected candidates stemming from the particular operation. In other words, having a higher value for r_{op} would lead to selecting candidates with higher overall scores, but might lead to none or only a few operations of that type being selected. We pick values of r_{op} that result in selecting candidates with high scores, while also leading to a similar number of selected candidates per operation type. We tune all hyperparameters, including r_{op}, n, etc., using the validation split of the LIAR-PLUS dataset.

3.2.4. Grammatical Correction

Once the best candidate explanation is selected, we apply a language toolkit over the candidate explanation (https://github.com/jxmorris12/language_tool_python, accessed on 2 April 2021), which detects grammatical errors such as capitalization and irrelevant punctuation, and returns a corrected version of the explanation. Furthermore, to ensure that we have no incomplete sentences, we remove sentences without verbs in the explanation.

3.2.5. Paraphrasing

Finally, to improve fluency and readability further, we use Pegasus [36], a model pre-trained with an abstractive text summarization objective. It focuses on relevant input parts to summarize the input semantics in a concise and more readable way. Since we want our explanations to be both fluent and human-readable, we leverage this pre-trained model without fine-tuning on downstream tasks. This way, after applying our iterative edit-based algorithm with grammatical error correction and paraphrasing, we obtain explanations that are fluent, coherent, and non-redundant.

4. Experiments

4.1. Datasets

We use two fact-checking datasets, LIAR-PLUS [4] and PubHealth [7]. These are the only two available real-world fact-checking datasets that provide short veracity justifications along with claims, ruling comments, and veracity labels. LIAR-PLUS contains 10,146 training, 1278 validation, and 1255 test data points from the political domain. PubHealth contains 9817 training, 1227 validation, and 1235 test data points from the health domain, including 447 claims about COVID-19. The labels used in LIAR-PLUS are {true, false, half-true, barely-true, mostly-true, pants-on-fire}, and in PubHealth, {true, false, mixture, unproven}.

While claims in LIAR-PLUS are only from PolitiFact, PubHealth contains claims from eight fact-checking sources. PubHealth has also been manually curated, e.g., to exclude poorly defined claims. Finally, the claims in PubHealth are more challenging to read than those in LIAR-PLUS and other real-world fact-checking datasets.

4.2. Models

Our experiments include the following models; their hyperparameters are given in Appendix F.

(Un)Supervised Top-N extracts sentences from the RCs, which are later used as input to our algorithm. The sentences are extracted in either a supervised or unsupervised way (see Section 3.1).
(Un)Supervised Top-N+Edits-N generates explanations with the iterative edit-based algorithm (Section 3.2.3) and grammar correction (Section 3.2.4). The model is fed with sentences extracted from RCs in an (un)supervised way.
(Un)Supervised Top-N+Edits-N+Para generates explanations by paraphrasing the explanations produced by Edits-N - (Un)Supervised (see Section 3.2.5).
Atanasova et al. [20] is a reference model that trains a multi-task system to predict veracity labels and extract explanation sentences. The model extracts N sentences, where N is the average number of the sentences in the justifications of each dataset. Kotonya and Toni [7] is a baseline model that generates abstractive explanations with an average sentence length of 3.
Lead-K [37] is a common lower-bound baseline for summarisation models, which selects the first K sentences from the RCs.

4.3. Iterative Edit-Based Algorithm

The proposed scoring functions (Section 3.2.2) and the iterative edit-based algorithm (Section 3.2.3) introduce hyper-parameters for controlling the importance of the individual post-editing scores as well as the efficiency and effectiveness trade-off of the iterative post-editing algorithm. We choose the hyper-parameter values with a standard hyper-parameter search over several values over a held-out validation set (Appendix B). The hyper-parameters enhance the proposed algorithm by making it adaptable to the specifics of the downstream application task. For example, one can easily select the hyper-parameter values depending on the required length, fluency, and semantic preservation of the produced explanations.

We select the editing target and the editing operation at random as the space of the possible operations and targets is computationally prohibitive, especially given long textual inputs, such as veracity explanations. While we follow related work [8] by selecting these at random, the scoring functions, as well as the threshold (r_{op}) used in the interactive edit-based algorithm, ensure that only fluent and semantically coherent sentences are selected at each step.

4.4. Evaluation Overview

We perform both automatic and manual evaluations of the models above. We include automatic ROUGE F1 scores (overlap of the generated explanations with the gold ones, Section 5.1) for compatibility with prior work. We further include automatic measures for assessing readability (see Section 5.2). While the latter was not included in prior work, we consider readability an essential quality of an explanation, and thus report it. We note, however, that the employed automatic measures are limited as they are based on word-level statistics. Especially ROUGE F1 scores should be taken with a grain of salt, as only exact matches of words are rewarded with higher scores, where paraphrases or synonyms of words in the gold summary are not scored. Hence, we also conduct a manual evaluation following Atanasova et al. [20] to further assess the quality of the generated explanations with a user study. As manual evaluation is expensive to obtain, the latter is, however, usually estimated based on small samples.

5. Automatic Evaluation and Results

As mentioned above, we use ROUGE F1 scores to compute overlap between the generated explanations and the gold ones, and compute readability scores to assess how challenging the produced explanations are to read.

5.1. Automatic ROUGE Scores

Metrics. To evaluate the generated explanations w.r.t. the gold justifications, we follow Kotonya and Toni [7], Atanasova et al. [20] and use measures from automatic text summarisation – ROUGE-1, ROUGE-2, and ROUGE-L F1 scores. These account for n-gram (1/2) and longest (L) overlap between generated and gold justification. The scores are recall-oriented, i.e., they calculate how many of the n-grams in the gold text appear in the generated one.

Caveats. Here, automatic evaluation with ROUGE scores is used to verify that the generated explanations preserve information important for the fact check, as opposed to generating completely unrelated text. Thus, we are interested in whether the ROUGE scores of the post-edited explanations are close but not necessarily higher than the ROUGE scores of the selected sentences from the RCs given as input. This work includes paraphrasing and insertion of new words to improve the readability of the explanation, which, while bearing the same meaning, necessarily results in lower ROUGE scores.

Results. In Table 1, we present the ROUGE score results. First, comparing the results for the input Top-N sentences with the intermediate and final explanations generated by our system, we see that, while very close, the ROUGE scores tend to decrease. For PubHealth, we also see that the intermediate explanations always have higher ROUGE scores compared to the final explanations from our system. These observations corroborate two main assumptions about our system. First, our system manages to preserve a large portion of the information important for explaining the veracity label, which is also present in the justification. This is further corroborated by observing that the decrease in the ROUGE scores is often not statistically significant ($p < 0.05$, except for some ROUGE-2 and one ROUGE-L score). Second, the operations in the iterative editing and the subsequent paraphrasing allow for the introduction of novel n-grams, which, while preserving the meaning of the text, are not explicitly present in the gold justification, thus, affecting the word-level ROUGE scores. We further discuss this in Section 7 and the Appendix E.

The ROUGE scores of the explanations generated by our post-editing algorithm when fed with sentences selected in an unsupervised way are considerably lower than with the supervised models. The latter illustrates that supervision for extracting the most important sentences is important to obtain explanations close to the gold ones. Finally, the systems' results are mostly above the LEAD-N scores, with a few exceptions for the unsupervised explanations for LIAR-PLUS.

Table 1. ROUGE-1/2/L F1 scores (see Section 5.1), and readability measures (see Section 5.2) over the test splits (for validation and ablations, see the Table A3 in appendix). Readability measures include sample variance. In *italics*, we report results reported from prior work, where we do not always have the outputs to compute readability. Underlined ROUGE scores of the Top-N+Edits-N and Top-N+Edits-N+Para are statistically significant ($p < 0.05$) compared to the input Top-N ROUGE scores, N = {5, 6}. Readability scores for Top-N+Edits-N and Top-N+Edits-N+Para are statistically significant ($p < 0.05$) compared to Top-N, and to Atanasova et al. [6]-3/4, except for the score in purple.

	Method	R-1↗	R-2↗	R-L↗	Flesch ↗	Dale–Chall ↘
			LIAR-PLUS			
Baselines	Lead-4	*28.11*	*6.96*	*24.38*	51.70 ± 14.85	8.72 ± 0.95
	Lead-6	29.15	8.28	25.84	53.24 ± 12.18	8.42 ± 0.78
Supervised	Top-6 (Supervised)	34.42	12.36	30.58	58.39 ± 12.11	7.88 ± 0.80
	Top-6+Edits-6	33.92	11.73	30.01	60.20 ± 12.08	7.74 ± 0.86
	Top-6+Edits-6+Para	33.94	11.25	30.08	66.33 ± 11.09	7.41 ± 0.91
Unsupervised	Top-6 (Unsupervised)	29.63	7.58	25.86	53.32 ± 10.86	8.50 ± 0.73
	Top-6+Edits-6	28.93	7.06	25.14	55.25 ± 12.03	8.46 ± 0.85
	Top-6+Edits-6+Para	28.98	6.84	25.39	62.13 ± 11.16	8.10 ± 0.89
	Atanasova et al. [6]-4	*35.70*	*13.51*	*31.58*	58.55 ± 13.70	7.97 ± 1.05
	Justification	-	-	-	58.81 ± 13.33	8.22 ± 1.07

Table 1. *Cont.*

	Method	R-1↗	R-2↗	R-L↗	Flesch ↗	Dale–Chall ↘
			PubHealth			
	Lead-3	*29.01*	*10.24*	*24.18*	-	-
Baselines	Lead-3	23.05	6.28	19.27	44.43 ± 22.97	9.10 ± 1.32
	Lead-5	23.73	6.86	20.67	45.95 ± 18.77	8.85 ± 1.03
	Top-5 (Supervised)	29.93	12.42	26.24	48.63 ± 14.14	8.67 ± 0.89
Supervised	Top-5+Edits-5	29.38	11.16	25.41	53.79 ± 14.56	8.36 ± 0.97
	Top-5+Edits-5+Para	28.40	9.56	24.37	61.38 ± 12.69	7.96 ± 0.98
	Top-5 (Unsupervised)	23.52	6.12	19.93	45.20 ± 14.36	8.94 ± 0.88
Unsupervised	Top-5+Edits-5	23.09	5.56	19.44	50.74 ± 14.92	8.62 ± 0.99
	Top-5+Edits-5+Para	23.35	5.38	19.56	60.06 ± 12.97	8.14 ± 0.95
	Kotonya and Toni [7]-3	*32.30*	*13.46*	*26.99*	-	-
	Atanasova et al. [6]-3	33.55	13.12	29.41	48.72 ± 16.38	8.87 ± 1.09
	Justification	-	-	-	49.28 ± 19.08	9.15 ± 1.61

Overall observations. We note that while automatic measures can serve as sanity checks and point to major discrepancies between generated explanations and gold ones, related work in generating fact-checking explanations [20] has shown that the automatic scores to some extent disagree with human evaluation studies, as they only capture word-level overlap and cannot reflect improvements of explanation quality. Human evaluations are therefore conducted for most summarisation models [38,39], which we include in Section 6.

5.2. Readability Results

Metrics. Readability is a desirable property for fact-checking explanations, as explanations that are challenging to read would fail to convey the reasons for the chosen veracity label and would not improve the trust of end-users. To evaluate readability, we compute Flesch Reading Ease [40] and Dale–Chall Readability Score [41]. The Flesch Reading Ease metric gives a text a score between 1 and 100, where a score between 50 and 30 requires college education and is difficult to read, a score between 50 and 60 requires a 10th to 12th school grade and is still fairly difficult to read, a score between 60 and 70 is regarded as plain English, which is easily understood by 13- to 15-year-old students. The Dale–Chall Readability Score gives a text a score between 9.0 and 9.9 when it is easily understood by a 13th to 15th-grade (college) student, a score between 8.0 and 8.9 when it is easily understood by an 11th or 12th-grade student, a score between 7.0 and 7.9 when it is easily understood by a 9th or 10th-grade student.

Results. Table 1 presents the readability results. We find that our iterative edit-based algorithm consistently improves the reading ease of the explanations by up to 5.16 points, and reduces the grade requirement by up to 0.32 points. Conducting paraphrasing further improves the reading ease of the text by up to 9.32 points, and reduces the grade requirement by up to 0.48 points. It is also worth noting that the explanations produced by Atanasova et al. [20] as well as the gold justifications are fairly difficult to read and can require even college education for grasping the explanation, while the explanations generated by our algorithm can be easily understood by 13- to 15-year-old students according to the Flesch Reading Ease score.

Overall observations. Our results show that our method makes fact-checking explanations less challenging to read and makes them accessible to a broader audience of up to 10th-grade students.

6. Manual Evaluation and Results

As automated ROUGE scores only account for word-level similarity between the generated and the gold explanation, and the readability scores account only for surface-

level characteristics of the explanation, we further conduct a manual evaluation of the quality of the produced explanations.

6.1. Explanation Quality

We manually evaluate two explanations: the input Top-N sentences, and the final explanations produced after paraphrasing (Edits-N+Para). We perform a manual evaluation of the test explanations obtained from supervised selection for both datasets with two annotators for each. Both annotators have a university-level education in English.

Metrics. We show a claim, veracity label, and two explanations to each annotator and ask them to rank the explanations according to the following criteria. **Coverage** means the explanation contains important and salient information for the fact check. **Non-redundancy** implies the explanation does not contain any redundant/repeated/not relevant information to the claim and the fact check. **Non-contradiction** checks if there is information contradictory to the fact check. **Fluency** measures the grammatical correctness of the explanation and if there is a coherent story. **Overall** measures the overall explanation quality. Following Atanasova et al. [20], we allow annotators to give the same rank to both explanations. We randomly sample 40 instances and do not provide the annotators with information about the explanation type. We choose 40 instances following related work [20] and work in the domain of automated summarisation [42], which use this low number of annotators/annotations due to the incurring annotation costs.

Results. Table 2 presents the human evaluation results for the first task. Each row indicates the annotator number and the number of times they ranked an explanation higher for one criterion. *Both* refers to both explanations being equal. Our system's explanations achieve higher acceptance for non-redundancy and fluency for LIAR-PLUS. The results are more pronounced for the PubHealth dataset, where our system's explanations were preferred in almost all metrics by both annotators. We hypothesise that PubHealth being a manually curated dataset leads to overall cleaner post-editing explanations, which annotators prefer.

Table 2. Manual annotation results of explanation quality with two annotators for both datasets. Each value indicates the relative proportion of when an annotator preferred a justification for a criterion. The preferred method, out of the input Top-N and the output of our method, Top-N+Edits-N+Para, is emboldened, Both indicates no preference.

	LIAR-PLUS			PubHealth		
#	Top-L	E-N+P	Both	Top-L	E-N+P	Both
			Coverage			
1	**42.5**	0.0	57.5	27.5	**60.0**	12.5
2	**40.0**	5.0	55.0	**22.5**	20.0	57.5
			Non-redundancy			
1	10.0	**87.5**	2.5	10.0	**82.5**	7.5
2	7.5	**10.0**	82.5	7.5	**75.0**	17.5
			Non-contradictory			
1	**32.5**	5.0	62.5	7.5	**10.0**	82.5
2	**10.0**	7.5	82.5	**20.0**	15.0	65.0
			Fluency			
1	40.0	**57.5**	2.5	35.0	**52.5**	12.5
2	**77.5**	15.0	7.5	20.0	**72.5**	7.5
			Overall Quality			
1	**57.5**	42.5	0.0	35.0	**62.5**	2.5
2	**62.5**	15.0	22.5	25.0	**67.5**	7.5

6.2. Explanation Informativeness

Metrics. We also perform a manual evaluation for veracity prediction. We ask annotators to provide a veracity label for a claim and an explanation where, same as for the evaluation of Explanation Quality, the explanations are either our system's input or

output. The annotators provide a veracity label for three-way classification; true, false, and insufficient (see map to original labels for both datasets in Appendix A. We use 30 instances of explanation type and perform evaluation for both datasets with two annotators for each dataset and instance.

Results. For the LIAR-PLUS dataset, one annotator gave the correct label 80% times for input and 67% times for the output explanations. The second annotator chose the correct label 56% times using output explanations and 44% times using input explanations. However, both annotators found at least 16% of explanations to be insufficient for the task of veracity prediction (Table A1 in Appendix A).

For PubHealth, both annotators found each explanation to be useful for the task. The first annotator chose the correct label 50% & 40% of the times for the given input & output explanations. The second annotator chose the correct label in 70% of the cases for both explanations. This corroborates that for a clean dataset such as PubHealth our explanations help for the task of veracity prediction.

7. Discussion

Results from our automatic and manual evaluation suggest two main implications of applying our post-editing algorithm over extracted RCs. First, with the automatic ROUGE evaluation, we confirmed that the post-editing preserves a large portion of important information that is contained in the gold explanation and is important for the fact check. This was further supported by our manual evaluation of veracity predictions, where the post-edited explanations have been most useful for predicting the correct label. We conjecture the above indicates that our post-editing can be applied more generally to summaries generated automatically for knowledge-intensive tasks, such as fact checking and question answering, where the information needed for prediction has to be preserved.

Second, with both the automatic and manual evaluation, we also corroborate that our proposed post-editing method improves several qualities of the generated explanations – fluency, conciseness, and readability. The latter are important prerequisites for building trust in automated fact-checking predictions as Thagard [43] find that people generally prefer simpler, more general explanations with fewer causes. They can also contribute to reaching a broader audience when conveying the veracity of the claim. Conciseness and readability are also the downsides of current professional long and in-depth ruling comments, which some leading fact-checking organisations, e.g., PolitiFact, (https://www.politifact.com/, accessed on 1 April 2021) have slowly started addressing by including short overview sections for the RCs.

8. Conclusions

In this work, we present an unsupervised post-editing approach to improve extractive explanations for fact-checking. Our novel approach is based on an iterative edit-based algorithm and rephrasing-based post-processing. In our experiments on two fact-checking benchmarking datasets, we observe, in both the manual & automatic evaluation, that our approaches generate fluent, coherent, and semantically preserved explanations.

For future work, an obvious next step is to investigate the applicability of our approach for other downstream tasks, such as machine summarisation, where the requirements for length and readability could vary depending on the end-user specifics. Furthermore, future work could explore additional improvements regarding the computational complexity of the proposed approach. For example, generative models trained with few-shot learning from a few post-editing examples could be employed to perform efficiently and effectively different editing operations. This would reduce the space of possible target positions and editing operations, especially for long input texts, such as veracity ruling comments. Finally, future work could explore other editing scores, e.g., scores optimising properties of natural language explanations, such as whether the explanation can be used to simulate the veracity prediction of the model.

Author Contributions: Conceptualization, S.J., P.A. and I.A.; Data curation, S.J. and P.A., Formal analysis, S.J.; Methodology, S.J. and P.A.; Software, S.J. and P.A.; Writing—original draft, S.J. and P.A.; Writing—review and editing, S.J., P.A. and I.A.; Supervision, I.A. All authors have read and agreed to the published version of the manuscript.

Funding: Shailza Jolly was supported by the TU Kaiserslautern CS Ph.D. scholarship program, the BMBF project XAINES (Grant 01IW20005), a STSM grant from the COST project Multi3Generation (CA18231), and the NVIDIA AI Lab (NVAIL) program. Pepa Atanasova has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 801199. Isabelle Augenstein's research is further partially funded by a DFF Sapere Aude research leader grant.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: We use open-source datasets that can be accessed from the referenced papers introducing the corresponding datasets.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A. Manual Evaluation

As explained in the Section 6 of the main paper, we mapped user inputs (TRUE/FALSE) for task two to the original labels for each dataset. For Liar, we map "true", "mostly-true", "half-true" to TRUE and "false", "pants-on-fire", and "barely-true" to FALSE. In the PubHealth dataset, we map "true" to TRUE, "false" to FALSE. The "insufficient" label is mapped to UNPROVEN. This way, once the mapping is done, we then compute the number of matches and non-matches to obtain an overall accuracy for this subset.

We appointed annotators with a university-level education in English.

Table A1. Results of manual evaluation for second task, i.e., predicting veracity label. DT refers to data type, # refers to annotator number, M/NM refers to number of matches/non-matches between annotator labels and original labels and I refers to number of times annotators found an explanation not sufficient to predict a label.

#	DT	LIAR-PLUS			PubHealth		
		M	NM	I	M	NM	I
1	Top-L	20	5	5	15	15	0
1	Edits-N+Para	14	7	9	12	18	0
2	Top-L	11	14	5	21	9	0
2	Edits-N+Para	13	10	7	21	9	0

Appendix B. Iterative Edit-Based Algorithm

We used the validation split of LIAR-PLUS to select the best hyperparameters for both datasets. We use the weight of 1.5, 1.2, 1.4, 0.95 for α, η, γ, δ and 1.0 for β in our scoring function. We set the thresholds as 0.94 for reordering, 0.97 for deletion, and 1.10 for insertion. We keep all models – GPT-2, RoBERTa, and Pegasus, fixed and do not finetune them on any in-house dataset. We run our search algorithm on a single V100-32 GB GPU for 220 steps, which takes around 13 h for each split for both datasets.

Appendix C. Automatic Evaluation

In Tables A2 and A3, we provide results over both dev and test splits of the dataset for the ROUGE and readability automatic evaluation. We additionally provide ablation results for components of our approach. First, applying Pegasus directly on the extracted sentences preserves a slightly larger amount of information when compared to applying Pegasus on top of the iterative editing approach—up to 0.96 ROUGE-L scores, but the readability scores are still lower—up to 4.28 Flesch Reading Ease points. We also show

results of the two parts included in the Edits step—the iterative editing and the grammar correction. We find that the grammar correction improves the ROUGE scores with up to 8 ROUGE-L score points and up to 8 Flesch Reading Ease points.

Table A2. Flesch Reading Ease (Flesch) and Dale–Chall Readability Score (Dale–Chall) for Validation (V) and Test (T) sets. Ablations are provided for the method as well – input selected sentences of Top-6, iterative-editing (Edits-IE), grammatical corrections (Edits-Gram), paraphrasing (Para).

	Method	Flesch-V ↗	Flesch-T ↗	Dale-Chall-V ↘	Dale-Chall-T ↘
		LIAR-PLUS			
	Justification	58.90 ± 13.38	58.81 ± 13.33	8.26 ± 1.08	8.22 ± 1.07
	Atanasova et al. [6]-4	54.76 ± 11.53	58.55 ± 13.70	8.38 ± 0.76	7.97 ± 1.05
Sup.	Top-6	57.77 ± 11.54	58.39 ± 12.11	7.90 ± 0.81	7.88 ± 0.80
	Top 6+Para	63.87 ± 10.60	64.44 ± 10.78	7.55 ± 0.76	7.52 ± 0.78
	Top 6+Edits	55.70 ± 12.40	56.26 ± 14.12	6.50 ± 0.69	6.46 ± 0.80
	Top 6+Edits+Gram	59.52 ± 11.98	60.20 ± 12.08	7.77 ± 0.88	7.74 ± 0.86
	Top 6+Edits+Gram+Para	66.04 ± 10.74	66.33 ± 11.09	7.44 ± 0.85	7.41 ± 0.91
Unsup.	Top-6	52.84 ± 10.37	53.32 ± 10.86	8.51 ± 0.69	8.50 ± 0.73
	Top 6+Para	59.33 ± 10.43	59.82 ± 10.58	8.13 ± 0.70	8.20 ± 0.80
	Top 6+Edits	50.70 ± 11.09	50.92 ± 12.54	6.91 ± 0.50	6.96 ± 0.62
	Top 6+Edits+Gram	54.76 ± 11.53	55.25 ± 12.03	8.38 ± 0.76	8.46 ± 0.85
	Top 6+Edits+Gram+Para	61.80 ± 11.11	62.13 ± 11.16	8.01 ± 0.77	8.10 ± 0.89
		PubHealth			
	Justification	48.19 ± 17.77	49.28 ± 19.08	9.21 ± 1.53	9.15 ± 1.61
	Atanasova et al. [6]-3	49.68 ± 15.96	48.72 ± 16.38	8.81 ± 1.09	8.87 ± 1.09
Sup.	Top-5	49.56 ± 13.48	48.63 ± 14.14	8.63 ± 0.88	8.67 ± 0.89
	Top 5+Para	57.52 ± 12.07	57.28 ± 12.35	8.18 ± 0.87	8.20 ± 0.88
	Top 5+Edits	47.38 ± 14.61	46.22 ± 15.95	7.06 ± 0.67	7.10 ± 0.75
	Top 5+Edits+Gram	54.30 ± 13.01	53.79 ± 14.56	8.32 ± 0.92	8.36 ± 0.97
	Top 5+Edits+Gram+Para	61.51 ± 11.28	61.38 ± 12.69	7.95 ± 0.92	7.96 ± 0.98
Unsup.	Top-5	43.54 ± 17.96	45.20 ± 14.36	9.25 ± 1.13	8.94 ± 0.88
	Top 5+Para	56.32 ± 11.41	55.78 ± 11.91	8.35 ± 0.83	8.39 ± 0.84
	Top 5+Edits	42.70 ± 17.01	42.29 ± 17.34	7.34 ± 0.79	7.36 ± 0.80
	Top 5+Edits+Gram	50.45 ± 14.45	50.74 ± 14.92	8.64 ± 0.95	8.62 ± 0.99
	Top 5+Edits+Gram+Para	60.24 ± 11.77	60.06 ± 12.97	8.12 ± 0.93	8.14 ± 0.95

Table A3. ROUGE-1/2/L F1 scores (see Section 5.1) for the edited justifications, higher results are better. Results in *italics* are those reported in the corresponding related work. Ablations are provided for the method as well – input selected sentences of Top-6, iterative-editing (Edits-IE), grammatical corrections (Edits-Gram), paraphrasing (Para).

	Method	Validation			Test		
		R-1	R-2	R-L	R-1	R-2	R-L
		LIAR-PLUS					
Baseline	Lead-4	*27.92*	*6.94*	*24.26*	*28.11*	*6.96*	*24.38*
	Lead-6	28.92	8.33	25.69	29.15	8.28	25.84
Supervised	Top-6	34.30	12.20	30.51	34.42	12.36	30.58
	Top 6 + Para	34.49	11.51	30.72	34.60	11.79	30.79
	Top 6 + Edits-IE	25.17	8.60	22.07	25.49	8.76	22.28
	Top 6 + Edits-IE + Edits-Gram	34.07	11.59	30.14	33.92	11.73	30.01
	Top-6 + Edits-IE + Edits-Gram + Para	34.20	11.05	30.29	33.94	11.25	30.08
Unsupervised	Top-6	29.24	7.99	25.83	29.63	7.58	25.86
	Top 6 + Para	29.94	7.72	26.40	29.92	7.35	26.24
	Top-6 + Edits-IE	21.49	5.67	18.77	22.73	5.56	19.51
	Top 6 + Edits-IE + Edits-Gram	29.00	7.46	25.51	28.93	7.06	25.14
	Top 6 + Edits-IE + Edits-Gram + Para	29.40	7.25	25.90	28.98	6.84	25.39
SOTA	Atanasova et al. [6]-4	*35.64*	*13.50*	*31.44*	*35.70*	*13.51*	*31.58*

Table A3. *Cont.*

	Method	Validation			Test		
		R-1	R-2	R-L	R-1	R-2	R-L
		PubHealth					
Baseline	Lead-3				*29.01*	*10.24*	*24.18*
	Lead-3	23.11	5.93	19.04	23.05	6.28	19.27
	Lead-5	24.20	6.83	20.89	23.73	6.86	20.67
Supervised	Top-6	30.35	12.63	26.43	29.93	12.42	26.24
	Top 5 + Para	29.76	10.75	25.47	29.43	10.69	25.51
	Top 5 + Edits-IE	22.49	8.94	19.70	22.11	8.72	19.49
	Top 5 + Edits-IE + Edits-Gram	29.58	11.18	25.54	29.38	11.16	25.41
	Top 5 + Edits-IE + Edits-Gram + Para	28.82	9.68	24.51	28.40	9.56	24.37
Unsupervised	Top-5	23.94	6.13	20.04	23.52	6.12	19.93
	Top 5 + Para	24.45	5.96	20.53	24.10	6.01	20.43
	Top-5 + Edits-IE	18.26	4.49	15.50	18.09	4.41	15.48
	Top 5 + Edits-IE + Edits-Gram	23.75	5.71	19.77	23.09	5.56	19.44
	Top-5 + Edits-IE + Edits-Gram + Para	23.97	5.46	19.98	23.35	5.38	19.56
SOTA	Kotonya and Toni [7]-3				*32.30*	*13.46*	*26.99*

Appendix D. Examples

Table A4 shows a qualitative example from the PubHealth dataset. We find that the final post-processed explanation is more readable, fluent, and concise in comparison to the originally selected explanation from RCs.

Table A4. Example explanations from Ruling Comments, Our iterative edit-based algorithm, and a combination of our algorithm and post-processing from test split of Pubhealth dataset. Each color presents an edit operation with **Reordering**, **deletion**, **insertion**, and **paraphrasing**.

Explanation from Ruling comments: Heavily-armed Muslims shouting "Allahu Akbar" open fire **campers and hikers** in a park. A heavily armed group of Middle Eastern looking Muslim men was arrested **outside Los Angeles** after opening fire upon hikers and campers in a large State Park in the area. There was no evidence found that a crime had been committed by any of the subjects **who were detained and they were released**. Moreover, the police report described the men only as "males," not "Middle Eastern males" or "Muslim males." The web site that started this rumor was Superstation95, which is not a "superstation" at all but rather a repository of misinformation from Hal Turner, who in 2010 was sentenced to 33 months in prison for making death threats against three federal judges. No credible news reports made any mention of the "Allahu Akbar" claim, and no witnesses stated they had been "shot at" by the men while hiking or camping.
Explanation from iterative algorithm: Heavily-armed Muslims males shouting "Allahu Akbar" open fire in a park. A heavily armed group of Middle Eastern looking Muslim men was arrested after opening fire upon hikers and campers in a large State Park **outside Los Angeles**. There was no evidence found that a crime had been committed by any of the subjects on **campers and hikers**. Furthermore, the police report described the men only as "," not "Middle Eastern" or "Muslim." **The website that started this rumor was Superstation95, which is not a "superstation" at all but rather a repository of misinformation from Hal Turner, who in 2010 was sentenced to 33 months in prison for making death threats against three federal judges.** No credible news reports made any mention of the "Allahu Akbar" claim, and no witnesses stated they had been "shot at".
Explanation from iterative algorithm + Post-processing: Muslims shout "Allahu Akbar" open fire in a park. A heavily armed group of Middle Eastern looking Muslim men was arrested after opening fire on hikers and campers in a large State Park outside Los Angeles. There was no evidence that a crime had been committed by any of the campers or hikers. **The website that started this rumor was Superstation95, which is not a "superstation" at all but rather a repository of misinformation from Hal Turner, who in 2010 was sentenced to 33 months in prison**. There were no credible news reports that mentioned the Allahu Akbar claim, and no witnesses that said they had been shot at.
Claim: The media covered up an incident in San Bernardino during which several Muslim men fired upon a number of Californian hikers.
Label False

Appendix E. Novelty and Copy Rate

Table A5 presents additional statistics for the generated explanations from the test sets of both datasets. First, we compute how many of the words from the input Top-N Ruling Comments are preserved in the final explanation. We find that with the final step of the post-editing process, up to 8% of the tokens from the Ruling comments are not found in the final explanation. On the other hand, our post-editing approach generates up to 10% novel words that are not previously found in the RCs. This could explain the lower results for the ROUGE scores, which account only for exact token overlaps. Finally, while ROUGE scores are recall-oriented, i.e., they compute how many of the words in the gold explanation can be found in the candidate one, we compute a precision-oriented statistic of the words in the candidate that can be found in the gold explanation. Surprisingly, while ROUGE scores of our generated explanations decrease after post-processing, the reverse score increases, pointing to improvements in the precision-oriented overlap with our method.

Table A5. Copy rate from the Ruling Comments, Novelty w.r.t the Ruling comments, and Coverage % of words in the explanation that are found in the justification.

Method	Copy Rate	Novelty	Gold Coverage
LIAR-PLUS			
Top-6 Sup.	100	0	29.2 ± 11.4
Justification	41.4 ± 13.0	58.6 ± 13.0	100
Top-6+Edits-6 Sup.	98.5 ± 1.8	1.5 ± 1.8	30.7 ± 12.1
Top-6+Edits-6+Para Sup.	90.8 ± 4.8	9.2 ± 4.8	32.5 ± 12.6
PubHealth			
Top-5 Sup.	100	0	26.3 ± 21.2
Justification	47.1 ± 21.0	52.9 ± 21.0	100
Top-5+Edits-6 Sup.	98.1 ± 3.4	1.8 ± 2.0	27.8 ± 21.3
Top-5+Edits-6+Para Sup.	90.4 ± 5.8	9.5 ± 5.2	28.5 ± 20.2

In addition, in LIAR/PubHealth, the average summary length is 136/142 tokens for the extracted RCs, 89/86 for the gold justifications, 118.7/117.3 after iterative editing, and 98.5/94.7 after paraphrasing.

Appendix F. Experimental Setup

Selection of Ruling Comments

For the supervised selection of Ruling Comments, as described in Section 3.1, we follow the implementation of the multi-task model of Atanasova et al. [20]. For LIAR-PLUS, we do not conduct fine-tuning as the model is already optimised for the dataset. For PubHealth, we change the base model to SciBERT, as the claims in PubHealth are from the health domain and previous work [7] has shown that SciBERT outperforms BERTs for the domain. In Table A6, we show the results for the fine-tuning we performed over the multi-task architecture with a grid-search over the maximum length limit of the text and the weight for the positive sentences in the explanation extraction training objective. We finally select and use explanations generated with the multi-task model with a maximum text length of 1700, and a positive sentence weight of 5.

For the unsupervised selection of explanation sentences, we employ a Longformer model. We construct the Longformer model with BERT as a base architecture and conduct 2000 additional fine-tuning steps for the newly added cross-attention weights to be optimised. We then train models for both datasets supervised by veracity prediction. The most salient sentences are selected as the sentences that have the highest sum of token saliencies.

Table A6. Fine-tuning for PubHealth supervised multi-task model over positive sentence loss weight, base model and maximum length.

Method	Validation			Test		
	R-1	R-2	R-L	R-1	R-2	R-L
SciBERT, w-1, l-1200	26.00	7.29	21.41	25.78	7.71	21.42
SciBERT, w-1, l-1500	27.78	9.81	23.32	27.37	9.62	23.07
SciBERT, w-1, l-1700	28.73	11.27	24.42	28.45	11.32	24.21
SciBERT, w-2, l-1700	30.15	12.32	25.66	29.71	12.04	25.35
SciBERT, w-5, l-1700	30.96	12.59	26.54	30.79	12.31	26.38

Finally, we remove long sentences and questions from the Ruling Comments, where the ROUGE score changes after filtering are illustrated in Table A7, which results in the Top-N sentences, that are used as input for the post-editing method.

Table A7. Sentence clean-up of long sentences for LIAR-PLUS and PubHealth.

Method	Validation			Test		
	R-1	R-2	R-L	R-1	R-2	R-L
LIAR-PLUS Unsup						
Top-6	29.26	7.98	25.83	29.62	7.94	26.04
Filtered Top-6	29.52	7.90	25.98	29.60	7.96	25.94
LIAR-PLUS SUP						
Top-6	34.42	12.35	30.64	34.49	12.54	30.67
Filtered Top-6	34.30	12.20	30.51	34.42	12.36	30.58
PubHealth Unsup						
Top-5	23.78	6.23	19.95	23.13	6.08	19.63
Filtered Top-5	23.94	6.13	20.04	23.52	6.12	19.93
PubHealth SUP						
Top-5	30.24	12.61	26.36	29.78	12.50	26.18
Filtered Top-5	30.35	12.63	26.43	29.93	12.42	26.24

These experiments were run on a single NVIDIA TitanRTX GPU with 24 GB memory and 4 Intel Xeon Silver 4110 CPUs. Model training took ∼3 h.

References

1. Kotonya, N.; Toni, F. Explainable Automated Fact-Checking: A Survey. In Proceedings of the 28th International Conference on Computational Linguistics, Barcelona, Spain, 8–13 December 2020; pp. 5430–5443. [CrossRef]
2. Thorne, J.; Vlachos, A.; Christodoulopoulos, C.; Mittal, A. 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, Volume 1 (Long Papers), New Orleans, LA, USA, 1–6 June 2018; pp. 809–819. [CrossRef]
3. Augenstein, I.; Lioma, C.; Wang, D.; Lima, L.C.; Hansen, C.; Hansen, C.; Simonsen, J.G. MultiFC: A Real-World Multi-Domain Dataset for Evidence-Based Fact Checking of Claims. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Hong Kong, China, 3–7 November 2019; pp. 4677–4691.
4. Wang, W.Y. "Liar, Liar Pants on Fire": A New Benchmark Dataset for Fake News Detection. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), Vancouver, BC, Canada, 30 July–4 August 2017; pp. 422–426.
5. Alhindi, T.; Petridis, S.; Muresan, S. Where is Your Evidence: Improving Fact-checking by Justification Modeling. In Proceedings of the First Workshop on Fact Extraction and VERification (FEVER), Brussels, Belgium, 1 November 2018; pp. 85–90. [CrossRef]
6. Atanasova, P.; Wright, D.; Augenstein, I. Generating Label Cohesive and Well-Formed Adversarial Claims. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), Online, 16–18 November 2020; pp. 3168–3177. [CrossRef]
7. Kotonya, N.; Toni, F. Explainable Automated Fact-Checking for Public Health Claims. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), Online, 16–18 November 2020; pp. 7740–7754. [CrossRef]
8. Liu, X.; Mou, L.; Meng, F.; Zhou, H.; Zhou, J.; Song, S. Unsupervised Paraphrasing by Simulated Annealing. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Online, 5–10 July 2020; pp. 302–312.

9. Kumar, D.; Mou, L.; Golab, L.; Vechtomova, O. Iterative Edit-Based Unsupervised Sentence Simplification. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Online, 5–10 July 2020; pp. 7918–7928.
10. Lu, Y.J.; Li, C.T. GCAN: Graph-aware Co-Attention Networks for Explainable Fake News Detection on Social Media. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Online, 5–10 July 2020; pp. 505–514. [CrossRef]
11. Wu, L.; Rao, Y.; Zhao, Y.; Liang, H.; Nazir, A. DTCA: Decision Tree-based Co-Attention Networks for Explainable Claim Verification. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Online, 5–10 July 2020; pp. 1024–1035. [CrossRef]
12. Atanasova, P.; Simonsen, J.G.; Lioma, C.; Augenstein, I. Diagnostics-Guided Explanation Generation. In Proceedings of the Thirty-Sixth AAAI Conference on Artificial Intelligence (AAAI'21), Virtually, 2–9 February 2021.
13. Mishra, R.; Gupta, D.; Leippold, M. Generating Fact Checking Summaries for Web Claims. In Proceedings of the Sixth Workshop on Noisy User-generated Text (W-NUT 2020), Online, 19 November 2020; pp. 81–90. [CrossRef]
14. Gunning, D. *Explainable Artificial Intelligence (xai)*; Defense Advanced Research Projects Agency (DARPA): Arlington County, VA, USA, 2017; Volume 2.
15. Camburu, O.M.; Rocktäschel, T.; Lukasiewicz, T.; Blunsom, P. e-SNLI: Natural Language Inference with Natural Language Explanations. In *Advances in Neural Information Processing Systems 31*; Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., Garnett, R., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2018; pp. 9539–9549.
16. DeYoung, J.; Jain, S.; Rajani, N.F.; Lehman, E.; Xiong, C.; Socher, R.; Wallace, B.C. ERASER: A Benchmark to Evaluate Rationalized NLP Models. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Online, 5–10 July 2020; pp. 4443–4458. [CrossRef]
17. Stammbach, D.; Ash, E. e-FEVER: Explanations and Summaries for Automated Fact Checking. In Proceedings of the 2020 Truth and Trust Online Conference (TTO 2020), Virtual, 16–17 October 2020, p. 32.
18. Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. Language Models are Few-Shot Learners. In *Proceedings of the Advances in Neural Information Processing Systems*; Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M.F., Lin, H., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2020; Volume 33, pp. 1877–1901.
19. Schumann, R.; Mou, L.; Lu, Y.; Vechtomova, O.; Markert, K. Discrete Optimization for Unsupervised Sentence Summarization with Word-Level Extraction. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Online, 5–10 July 2020; pp. 5032–5042.
20. Atanasova, P.; Simonsen, J.G.; Lioma, C.; Augenstein, I. Generating Fact Checking Explanations. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Online, 5–10 July 2020; pp. 7352–7364. [CrossRef]
21. Sanh, V.; Debut, L.; Chaumond, J.; Wolf, T. DistilBERT, a distilled version of BERT: Smaller, faster, cheaper and lighter. *arXiv* **2020**, arXiv:1910.01108.
22. Beltagy, I.; Lo, K.; Cohan, A. SciBERT: A Pretrained Language Model for Scientific Text. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Hong Kong, China, 3–7 November 2019; pp. 3615–3620. [CrossRef]
23. Beltagy, I.; Peters, M.E.; Cohan, A. Longformer: The Long-Document Transformer. *arXiv* **2020**, arXiv:2004.05150.
24. Simonyan, K.; Vedaldi, A.; Zisserman, A. Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps. *arXiv* **2014**, arXiv:1312.6034.
25. Atanasova, P.; Simonsen, J.G.; Lioma, C.; Augenstein, I. A Diagnostic Study of Explainability Techniques for Text Classification. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), Online, 5–10 July 2020; pp. 3256–3274. [CrossRef]
26. Kindermans, P.J.; Hooker, S.; Adebayo, J.; Alber, M.; Schütt, K.T.; Dähne, S.; Erhan, D.; Kim, B. The (un) reliability of saliency methods. In *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning*; Springer: Berlin, Germany, 2019; pp. 267–280.
27. Hinton, G.E. Training products of experts by minimizing contrastive divergence. *Neural Comput.* **2002**, *14*, 1771–1800. [CrossRef] [PubMed]
28. Manning, C.D.; Surdeanu, M.; Bauer, J.; Finkel, J.R.; Bethard, S.; McClosky, D. The Stanford CoreNLP natural language processing toolkit. In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations, Baltimore, MD, USA, 22–27 June 2014; pp. 55–60.
29. Li, J.; Li, Z.; Mou, L.; Jiang, X.; Lyu, M.; King, I. Unsupervised Text Generation by Learning from Search. In *Proceedings of the Advances in Neural Information Processing Systems*; Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M.F., Lin, H., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2020; Volume 33, pp. 10820–10831.
30. Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; Stoyanov, V. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *arXiv* **2019**, arXiv:1907.11692.
31. Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; Sutskever, I. Language models are unsupervised multitask learners. *OpenAI Blog* **2019**, *1*, 9.
32. Zhang, X.; Lapata, M. Sentence Simplification with Deep Reinforcement Learning. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, Copenhagen, Denmark, 7–11 September 2017; pp. 584–594. [CrossRef]

33. Kriz, R.; Sedoc, J.; Apidianaki, M.; Zheng, C.; Kumar, G.; Miltsakaki, E.; Callison-Burch, C. Complexity-Weighted Loss and Diverse Reranking for Sentence Simplification. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), Minneapolis, MN, USA, 2–7 June 2019; pp. 3137–3147. [CrossRef]
34. Rose, S.; Engel, D.; Cramer, N.; Cowley, W. Automatic keyword extraction from individual documents. *Text Min. Appl. Theory* **2010**, *1*, 1–20.
35. Reimers, N.; Gurevych, I.; Reimers, N.; Gurevych, I.; Thakur, N.; Reimers, N.; Daxenberger, J.; Gurevych, I. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing, Hong Kong, China, 3–7 November 2019.
36. Zhang, J.; Zhao, Y.; Saleh, M.; Liu, P. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. In Proceedings of the International Conference on Machine Learning. PMLR, Virtual Event, 13–18 July 2020; pp. 11328–11339.
37. Nallapati, R.; Zhai, F.; Zhou, B. SummaRuNNer: A Recurrent Neural Network Based Sequence Model for Extractive Summarization of Documents. In Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence (AAAI'17), San Francisco, CA, USA, 4–9 February 2017; pp. 3075–3081.
38. Chen, Y.C.; Bansal, M. Fast Abstractive Summarization with Reinforce-Selected Sentence Rewriting. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Melbourne, Australia, 15–20 July 2018; pp. 675–686. [CrossRef]
39. Tan, J.; Wan, X.; Xiao, J. Abstractive Document Summarization with a Graph-Based Attentional Neural Model. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Vancouver, BC, Canada, 30 July–4 August 2017; pp. 1171–1181. [CrossRef]
40. Kincaid, J.P.; Fishburne, R.P., Jr.; Rogers, R.L.; Chissom, B.S. *Derivation of New Readability Formulas (Automated Readability Index, Fog Count and Flesch Reading Ease Formula) for Navy Enlisted Personnel*; Technical report; Naval Technical Training Command Millington TN Research Branch: Millington, TN, USA, 1975.
41. Powers, R.D.; Sumner, W.A.; Kearl, B.E. A recalculation of four adult readability formulas. *J. Educ. Psychol.* **1958**, *49*, 99. [CrossRef]
42. Liu, Y.; Lapata, M. Text Summarization with Pretrained Encoders. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Hong Kong, China, 3–7 November 2019; pp. 3730–3740. [CrossRef]
43. Thagard, P. Explanatory coherence. *Behav. Brain Sci.* **1989**, *12*, 435–467. [CrossRef]