"Beware of deception": Detecting Half-Truth and Debunking it through Controlled Claim Editing

Anonymous ARR submission

Abstract

The prevalence of ***half-truths***, which are statements containing some truth but that are ultimately deceptive, has risen with the increasing use of the internet. To help combat this problem, we have created a comprehensive pipeline consisting of a half-truth detection model and a claim editing model. Our approach utilizes the T5 model for controlled claim editing; *"controlled"* here means precise adjustments to select parts of a claim. Our methodology achieves an average BLEU score of **0.88** (on a scale of 0-1) and a *disinfo-debunk* score of **85%** on edited claims. Significantly, our T5-based approach outperforms other Language Models such as GPT3, RoBERTa, ChatGPT, and Tailor, with average improvements of 82%, 57%, 12%, and 23% in *disinfo-debunk* scores, respectively. By extending the LIAR-PLUS dataset, we achieve an F1 score of **82%** for the half-truth detection model, setting a new benchmark in the field. While previous attempts have been made at half-truth detection, our approach is, to the best of our knowledge, the first to attempt to debunk half-truths.

1 Introduction

The dissemination of disinformation, especially in the form of half-truths, can have significant and negative implications as it has the potential to disrupt social and economic harmony (Allcott and Gentzkow, 2017; Su et al., 2020). A recent example of this was seen during the COVID-19 vaccination drive, where the spread of disinformation led to widespread fear and skepticism among the public regarding the efficacy and safety of the vaccine (He et al., 2021; Shahi and Nandini, 2020).

Our work tackles half-truths by utilizing the LIAR-PLUS dataset (Alhindi et al., 2018) for half-truth detection. There are many forms of half-truth such as deception, exaggeration, propaganda, and intentionally hidden facts, etc. In this work, we only deal with half-truths related to deception and intentionally hidden facts. To improve upon the LIAR-PLUS dataset, we added a new column to it, called ***shortened justification***, using the concept of textual entailment. This shortened justification is referred to as ***support*** when the label is true or mostly-true, and ***counter*** when the label is half-true, false, barely-true, or pants-on-fire. *Supports* or *counters*, in our context, are explanations for the label associated with each claim. We refer to these explanations as ***evidence*** in our work. Our approach not only detects half-truths but also aims to debunk the claim by editing and transforming it into a truthful statement. ***'Claim'***, as coined by (Toulmin, 2003), is *'an assertion that deserves our attention'*. In our study, a *claim* is defined as a textual statement that can be made by individuals, news websites, political parties, and other sources.

This research is a significant advancement in the field of natural language processing (NLP) and has the potential to contribute to fact-checking and computational journalism, ultimately helping to prevent people from falling prey to disinformation.

A ***half-truth*** is a statement that is partially true but intentionally omits important details that would significantly alter its meaning. This type of statement is deceptive as it can lead to misunderstandings or false impressions. Even if a statement is technically true, it cannot be considered entirely truthful if it excludes crucial information. Half-truths are lies of omission.

For instance, an example of a half-truth is the statement *"Electronic gadgets mandatory for e-census in 2023"*, which contains a hidden piece of information that people need not buy the gadgets since the government will provide them. The statement is partially accurate, but also misleading because it fails to disclose a crucial detail that could cause confusion for the reader.

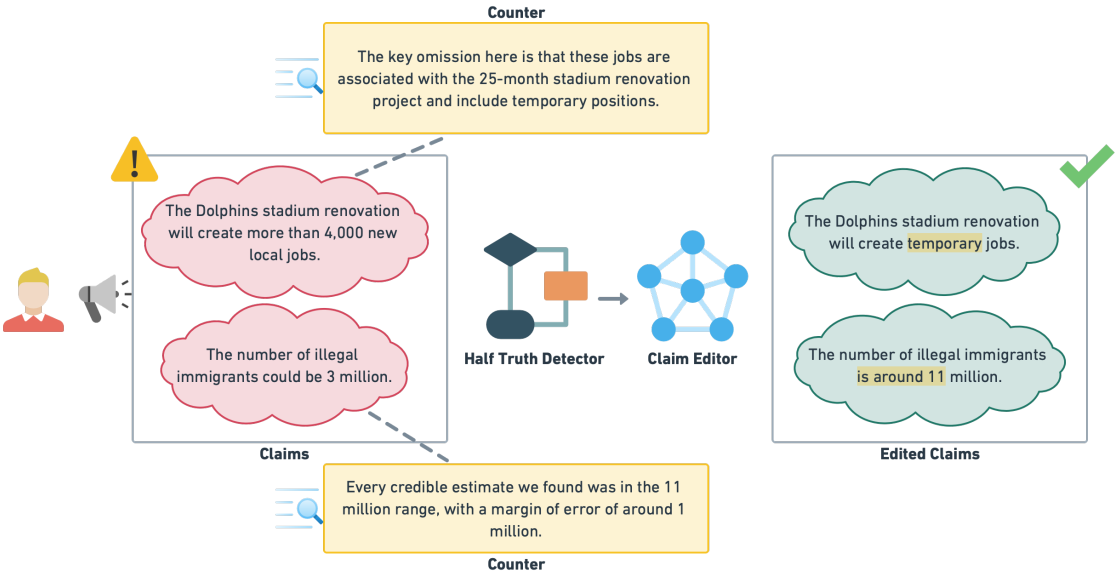
Counter
The key omission here is that these jobs are associated with the 25-month stadium renovation project and include temporary positions.
The Dolphins stadium renovation will create more than 4,000 new local jobs.
The number of illegal immigrants could be 3 million.
Claims
Half Truth Detector
Claim Editor
The Dolphins stadium renovation will create temporary jobs.
The number of illegal immigrants is around 11 million.
Edited Claims
Every credible estimate we found was in the 11 million range, with a margin of error of around 1 million.
Counter

Figure 1: Picture depicting the half-truth detection and debunking pipeline

1.1 Motivation

Generating sensational or misleading content can attract viewership, engagement, and advertising revenue. Some individuals and groups exploit this for personal gain, including political, financial, or ideological motivations. However, traditional fact-checking methods rely on human fact-checkers and can be time-consuming (Hassan et al., 2015), which limits their effectiveness in responding to the constant stream of disinformation. This is where automated fact-checking (Guo et al., 2022) and disinformation debunking systems become crucial, as they can quickly detect (Monti et al., 2019) and respond to disinformation in real-time, which can help limit its reach (Cohen et al., 2011). Our detection and debunking pipeline is a powerful tool that can help detect half-truths and false claims faster, thus saving valuable person-hours that would otherwise be spent on manual fact-checking. Additionally, our pipeline is equipped with an evidence extraction tool that allows us to collect evidence faster, which can then be used to debunk half-truths and false claims faster. Overall, our system is an important step toward helping combat the spread of disinformation on digital platforms.

1.2 Problem Definition

In this work, we have implemented a half-truth detection model to detect half-truths. Given a claim C and the corresponding evidence E as input, the half-truth detection model predicts whether the given claim is true, half-true, or false. It is a three-class classification problem.

In addition to that, we have implemented a claim editing model to edit *half-true* and *false* claims. Given a *half-true* or *false* claim C and the corresponding evidence E as input, our claim editing pipeline uses the evidence to edit the *half-true* or *false* claim and tries to generate an edited *true* claim $C*$ with control over editing the selected parts of input claim. The overall task is depicted in Figure 1.

1.3 Contributions

Our contributions are:

1. Extending the LIAR-PLUS dataset by adding an extra column called *shortened justification*, using textual entailment, and achieving a benchmark accuracy of **82%** for the half-truth detection model.

2. Devising a novel claim editing technique, which aided the T5 model, to outperform cutting-edge systems such as GPT by 82%, Roberta by 57%, ChatGPT[1] by 12%, and Tailor by 23% with a success rate of **85%** in the task of debunking claims and achieving an average content preservation score of **0.88** (on a scale of 0-1) on the edited claims.

1.4 Roadmap

The paper is structured as follows: In section 2, we have discussed the related works. Section 3

[1] ChatGPT

presents a model for detecting half-truths, while section 4 proposes a technique for editing half-true claims. The experiments and results are discussed in 5 and the conclusion and future work are outlined in section 6.

2 Related Work

Automated fact-checking is a process that involves the detection of the veracity of claims by extracting relevant evidence and validating the claim against the evidence. This has become a topic of immense interest and research in recent years, resulting in numerous works in the domain of fact-checking. Alhindi et al. (2018) introduced the LIAR-PLUS dataset and implemented a veracity prediction model using LSTM. The authors' work involved training the LSTM model to classify statements into six different categories based on their truthfulness, thereby demonstrating the effectiveness of deep learning approaches in this field. FEVER (Thorne et al., 2018) is a popular dataset in the domain of fact-checking.

The idea of detecting veracity using the emotion of the claim was proposed in Guo et al. (2019). The authors used sentiment analysis to detect the emotion conveyed by a claim and then used this to infer the claim's veracity. Estornell et al. (2019) discusses the computational complexity of deception by half-truth. The authors demonstrated that half-truths can be computationally more challenging to detect than other forms of deception, thus emphasizing the need for specialized approaches to identify and address this issue. Building on this idea, Monteiro et al. (2018) filtered out half-truths during fake news detection and expressed their idea of detecting half-truths in the future. Motivated by this idea, our work attempts to address the half-truth detection problem.

Along with half-truths, there are other forms of disinformation, such as fake news and exaggerated and sensationalized news. Wright and Augenstein (2021) focuses on detecting exaggeration in the claims made by press releases. The authors propose a supervised learning approach that utilizes sentence-level features to detect exaggerated claims. Li et al. (2017) conducted an analysis and inspection of exaggerated claims in the domain of scientific news. The authors proposed a framework that leverages natural language processing techniques to detect exaggerated claims in scientific news articles.

In addition to detecting half-truths, debunking them is necessary to keep a check on the spread of disinformation. Popat et al. (2018) discusses debunking fake news using external evidence. Atanasova et al. (2020) discusses generating explanations along with detection. Schuster et al. (2021) discusses about robust fact verification using evidence that changes with time. Counterfactuals called contrast sets (Gardner et al., 2020) can be created to debunk half-truths. Moreover, structural-level properties of a claim can be used to edit it by semantically controlling the text generation (Ross et al., 2021). Building on these ideas, we have developed a controlled claim editing technique that makes minimum edits to a half-true claim to make it true. Our approach leverages the structural-level properties of a claim to identify the necessary edits and then uses semantically controlled text generation to make the claim true.

Overall, our work attempts to address the challenging problem of half-truth detection and debunking by utilizing a combination of techniques, including contrast sets and controlled claim editing. By detecting and debunking half-truths, we can take a step toward combating disinformation and promoting the spread of accurate information.

3 Half-truth detection model

For half-truth detection, the task is to build a model that takes a claim and evidence and predicts one of the three labels: true, half-true, or false for the given input. Given a claim **C** and evidence **E**, and **y** is the generated label, we can mathematically formulate the task as follows:

$$y^* = \underset{y \in \{true, false, half-true\}}{argmax} \quad P(y|C;E) \qquad (1)$$

3.1 LIAR-PLUS-PLUS Dataset

The LIAR-PLUS dataset (Alhindi et al., 2018) is a benchmark dataset for fact-checking research, extended from the LIAR dataset (Wang, 2017). It has 12,000+ human-labeled statements classified into six categories: *true*, *mostly-true*, *half-true*, *barely-true*, *false*, or *pants-on-fire*. The dataset was collected from the Politifact[2]. We extended LIAR-PLUS by adding the ***shortened justification*** column and called it **LIAR-PLUS-PLUS** dataset. This column was created by using textual entailment technique on the extracted justification column of the LIAR-PLUS dataset. An excerpt from

[2] Politifact: Website

Statement: *"Says Rick Scott cut education to pay for even more tax breaks for big, powerful, well-connected corporations."*
Speaker: *Florida Democratic Party*
Context: *TV Ad*
Label: ***half-true***
Extracted Justification: *A TV ad by the Florida Democratic Party says Scott "cut education to pay for even more tax breaks for big, powerful, well-connected corporations." However, the ad exaggerates when it focuses attention on tax breaks for "big, powerful, well-connected corporations." Some such companies benefited, but so did many other types of businesses. And the question of whether the tax cuts and the education cuts had any causal relationship is murkier than the ad lets on.*
Shortened Justification: *However, the ad exaggerates when it focuses attention on tax breaks for "big, powerful, well-connected corporations." Some such companies benefited, but so did many other types of businesses.*

Figure 2: An excerpt from the LIAR-PLUS-PLUS dataset

the LIAR-PLUS-PLUS dataset is presented in Fig 2. The composition of LIAR-PLUS-PLUS data for training, validation, and test split is 10240, 1284, and 1283 instances respectively. The average number of sentences in the extracted justification is 6. The shortened justification is at max 2 sentences and a minimum of one sentence. We this technique, we there by reduce the number of sentences and try to obtain the relevant sentences.

3.1.1 Creation of shortened justification

To extract shortened justifications, we utilized a natural language inference (NLI) model that assigns entailment scores to pairs of sentences. Our algorithm employs this NLI model to generate supports and counters for each claim in the LIAR-PLUS dataset. A support is a statement that strengthens the claim, while a counter is a statement that challenges it. This is accomplished by calculating the entailment scores between each sentence in the claim and its corresponding justification. Each sentence is classified into one of three labels: entailment, contradiction, or neutral.

The rationale behind employing textual entailment is as follows. True and mostly-true claims typically have supporting text, which aligns with the *entailment* label. False and pants-fire claims, on the other hand, tend to have text that contradicts them, similar to a *contradiction* label. Half-true and barely-true claims often contain text that mentions hidden information or the deceptive aspect of the claim, which cannot be directly entailed or contradicted and thus corresponds to a *neutral* label. With this idea, we can have sufficient information in the evidence to detect these labels.

In our approach, we calculate the entailment scores on a sentence-by-sentence basis between the claim and each piece of evidence. Among all the sentences predicted as *entailment* or *contradiction*, we select the sentence with the highest confidence or probability score as the first part of the shortened justification. For sentences predicted as *neutral*, we choose the sentence with the highest confidence as the second part of the shortened justification. If no sentences are predicted as *neutral*, we have only one sentence in the shortened justification, and vice versa. As a result, the shortened-justification consists of a maximum of two sentences and a minimum of one sentence. We refer to this shortened justification as evidence, which serves as an explanation for the label associated with each claim. The details of our NLI model are discussed in section A.1 of the appendix.

We have performed a manual evaluation of the extracted shortened-justification using 2 evaluators. We selected 100 claims from each label and asked the annotators to check if the shortened justification contains the required information to predict the label of the claim. Out of 600 claims, for 568 claims, the annotators found sufficient information in the shortened-justification. This amounts to around **94.6%** of successful extraction of evidence.

We created a network with BERT ((Devlin et al., 2018) based classifier model. We call this model, the half-truth detection model. We used the LIAR-PLUS dataset and the LIAR-PLUS-PLUS dataset separately to train different versions of the model. The two versions are called **J(model trained on LIAR-PLUS dataset)** and **SJ (model trained on LIAR-PLUS-PLUS dataset)**. For version J, the input is claim and justification, and for version SJ, the input is the claim and shortened justification. We trained the models for the task of tri-class classification; true, false, and half-truth by grouping claims with half-true and barely-true labels into half-true, true, and mostly-true labels as true and false and pants-on-fire as false. This grouping has been done based on the annotation policy of the Politifact website and truth-o-meter[3] definitions. According to the definitions of truth-o-meter, barely-true statements are claims that have hidden information. Since, this falls into the category of half-truths, we have grouped barely-true and half-

[3] The Principles of Truth-O-Meter: Webpage

Label	Train	Test	Validation
true	3649	460	420
half-true	3780	481	485
false	2840	342	379

Table 1: The composition of train, test, and validation split of the LIAR-PLUS-PLUS dataset after grouping the labels.

true claims into a single label. The composition of the LIAR-PLUS-PLUS dataset after grouping the labels is shown in Table 1.

We tested both models on the test set of the LIAR-PLUS-PLUS dataset. We obtained an F1 score of **0.82**, which is considered a benchmark accuracy, for the SJ version of our model. On the other hand, for the J version of the model, we achieved an F1 score of **0.72**. The SJ version of the model outperformed the J version because it was presented with only the pertinent information in the justification. Therefore, we can confidently assert that the textual entailment task assisted in enhancing the accuracy of the half-truth detection model. By extracting solely the relevant information from the justification, the model is internally inferring if the statement is entailing or contradicting the shortened justification. As a result, the model can discern that half-truths are lies of omission, whereas truthful news entails and fake news contradicts the shortened justification. Logistic Regression and SVM models were used as baseline models, and they were trained on GloVe embeddings (Pennington et al., 2014) using both datasets (after grouping labels). The F1 scores of all these models are presented in Table 2. All these models are tested on the test set of the LIAR-PLUS-PLUS dataset. The confusion matrix and per-label macro averages of the best-performing model, BERT-based sequence classifier, trained using the LIAR-PLUS-PLUS dataset is presented in the Tabel 3. The macro average precision, recall, and F1 scores are **0.811**, **0.831**, and **0.82** respectively.

4 Claim Editing Model

The claim editing model is useful to edit a half-true or false claim. We can perform controlled claim editing using our technique. By control, we mean, we can make precise adjustments to the selected parts of the claim. We mask the tokens that need replacement and use the objective of masked language modeling to fill those masks. Thus, an edited claim is generated. Our objective is to convert false or half-true claims into true claims. If we find the contradicting part or the deceptive part in the claim and mask it, we can replace it with the correct part from the evidence. By making this adjustment to the false and half-true claims, they can be converted to true claims. So, firstly, we need a model to fill masked tokens using the context provided to the model.

Model	**Data**	**F1**	**P**	**R**
SVM	J	67.5	76	60.7
LR	J	66.8	74	60.8
BERT	J	72	74.8	69.2
SVM	SJ	72.6	71	74.1
LR	SJ	71.4	69	73.7
BERT	SJ	**82**	81.1	83.1

Table 2: Performance of Half-Truth detection module with various model-data combinations. Abbreviations F1:= F1-accuracy score %, P:= Precision %, R:= Recall %, J:= Model trained using the LIAR-PLUS dataset, SJ:= Model trained using the LIAR-PLUS-PLUS dataset, LR:= Logistic Regression Classifier.

4.1 TAPACO Dataset

We utilized the TAPACO paraphrase dataset (Scherrer, 2020) and augmented it. The TAPACO dataset is a paraphrase corpus consisting of 73 languages that were extracted from Tatoeba[4], which is a crowdsourcing project primarily for language learners. We have selected 60,000 instances from the TAPACO dataset to augment our dataset, where each instance consists of an original sentence and its corresponding paraphrased sentence.

Original sentence: *Many people respect you. Do not disappoint them.*

Paraphrased sentence: *A lot of people look up to you. Do not let them down.*

4.1.1 Dataset Augmentation:

We have obtained Semantic Role Labeling (SRL) tags for all paraphrased sentences and appended an additional column to the TAPACO dataset. Our approach involved utilizing the SRL generator provided by Allen AI [5] to extract SRL tags for each sentence. In some cases, the SRL generator produced multiple outputs for a single paraphrased sentence. We selected the output that contained the

[4] Tatoeba: Website

[5] Allen AI: Website

		Gold Labels			
		true	**false**	**half-true**	**per-label precision**
Model Outputs	**true**	364	0	96	0.791
	false	2	272	68	0.795
	half-true	31	42	408	**0.848**
	per-label recall	**0.916**	0.866	0.713	

Table 3: Confusion matrix and per-label precision and recall of the best-performing model, BERT-based sequence classifier, trained using the LIAR-PLUS-PLUS dataset

highest number of semantic roles (tags) to resolve this issue. The augmented dataset, which now contains three columns, as shown below, was utilized to train the claim editing model.

Original sentence: *Many people respect you. Do not disappoint them.*

Paraphrased sentence: *A lot of people look up to you. Do not let them down.*

SRL tagged Paraphrased sentence: *[ARG0: A lot of people] [V: look] [ARG1: up to you] . Don’t let them down.*

We use the augmented TAPACO dataset to train the T5 model for the task of claim editing. T5 is a text-text transformer model from Google. T5 is used for a variety of purposes, including machine translation, summarization, and masked language modeling. We took 50000 instances from the augmented dataset as a training split with a validation and test split of 5000 instances each. We have given the SRL tags of the paraphrased sentence and the original sentence with a few masked tokens as input to T5 and expect the original sentence as output.

Input: *[[ARG0: A lot of people] [V: look] [ARG1: up to you] . Don’t let them down .] Many people extra_id_0 you. Don’t extra_id_1 them.*

Output: *Many people respect you. Don’t disappoint them.*

During training, we have masked only nouns, adjectives, and verbs in the original claim to make sure that the model is not only learning the structural properties of the language but also semantics. That is the main motivation for us to use SRL-tagged paraphrased sentences in the input. This idea of using SRL tags to perturb and edit sentences have been used by Ross et al. (2022). With this mechanism of training, the model learns to fill the masked tokens by using the SRL-tagged paraphrased sentence. Hence, it maximizes the context (the paraphrased sentence) to fill only the masked tokens thereby minimizing the reconstruction loss. Now that our objective of filling masked tokens with the SRL-tagged context provided in the header (enclosed in []) is achieved by T5, we can adapt this to our task of claim editing.

For the task of claim editing, we can provide the SRL-tagged evidence in the header and the masked half-true or false claim as input to the T5 model. The model will fill the masked tokens using the SRL-tagged evidence. For editing half-true and false claims in the LIAR-PLUS-PLUS dataset, we use the ’shortened justification’ as evidence. We call it the counter. We used the Allen AI SRL tag generator to extract SRL tags of counters. If we have multiple outputs from the SRL generator, we take the one with the maximum number of tags as the SRL-tagged counter. In addition to that, we need to mask the deceptive and contradicting part of the claim. We use the below strategy to mask a false or half-true claim.

Masking false or half-true claim:

To ensure accurate editing of the claim, it is necessary to mask specific tokens rather than selecting them randomly. To identify the appropriate tokens for masking, we employ the concepts of textual entailment and cosine similarity. Using an AllenNLP constituency parser, we divide the claim into multiple segments. Only those segments that contradict the evidence or exhibit lower similarity are considered for replacement with a mask. By utilizing an NLI model, we calculate scores indicating the degree of contradiction. If no contradictory segments are found, we mask the segment that exhibits lower similarity with the counter. This strategy facilitates the T5 model in accurately filling the masked tokens using the counter, thereby ensuring precise editing of the relevant tokens.

The T5 model trained by us is capable of generating claims where the number of edits will be less and original content is preserved. We have given the SRL-tagged counter (as header) and the masked claim as input to the T5 model. The model generates a list of edited claims. In our case, we have limited the generations to five in number. We used Constrained Beam Search for text generation. Unlike ordinary beam search, constrained beam search allowed us to exert control over the output of text generation.

Example of claim editing:
Original claim- *The Dolphins stadium renovation will create more than 4,000 new local jobs.*
Masked claim- *The Dolphins stadium renovation will create extra_id_0.*
Counter- *The key omission here is that these are jobs associated with the 25-month stadium renovation project and include temporary positions.*
Input to T5- *[The key omission here is that [ARG2: these] are jobs associated with the 25-month stadium renovation project and [V: include] [ARG1: temporary positions] .] The Dolphins stadium renovation will create extra_id_0.*
Output from T5- List of edited claims (For example, one of the edited claims is *"The Dolphins stadium renovation will create temporary jobs."*)

Claim Filtering:
One edited claim among the list of edited claims will be filtered out as the best claim. We employed a filtering technique that filters out the best-edited claim using the reward mechanism of a claim being true, maximizing the word overlap between the original claim and the edited claim. For the reward mechanism, we have used the SJ version of the half-truth detection model (0.82 F1 accuracy). For a set of edited claims, we find the label predicted by the half-truth detection model. For this model, we give the edited claim and the counter as input. The edited claims that are predicted as true by the half-truth detection model are rewarded highly. Because, now after editing, we have a validation that the half-true or false claims have been converted to true. If none of the edited claims is true, we filter all the edited claims to the next stage. These filtered claims are checked for maximum word overlap with the original claim and the one with the highest overlap is considered the best-edited claim. We give importance to word overlap because we want to edit the claim minimally. The motivation behind this minimal edit was taken from Gardner et al. (2020).

5 Experiments and Results

In this section, we have presented the experiments performed on the task of claim editing task and our findings.

5.1 Baselines

We have used different language models for the task of claim editing to compare these results with our technique. We compare our claim editing technique with various language models such as GPT (Radford and Narasimhan, 2018), RoBERTa (Liu et al., 2019), ChatGPT[6] and Tailor (Ross et al., 2022). Please refer to section A.2 of the appendix for the usage of LLMs for claim editing.

5.2 Evaluation Metrics

We have evaluated the edited claims on two evaluation metrics to make sure the quality of the edits is not compromised in our technique. The two metrics are **content preservation** and **disinfo-debunk** (disinformation debunk). A minimally edited sentence must maintain its content. In this paper, *content preservation* was evaluated using the BLEU score (Papineni et al., 2002). Content preservation is not a similarity metric, it is a metric to measure overlap. The *content preservation* score is the BLEU score between the edited claim and the original claim. This metric measures how overlapped the edited claim and original claim are since we wanted to edit the claim minimally. The end goal of claim editing is to debunk false and half-true claims. Hence, we should evaluate how many claims have been converted to true after editing them using our technique and also the other language models. The *disinfo-debunk* metric measures the percentage of claims that have been converted to true after editing them. We have used the BERT(SJ) model to detect the label of the edited claim and compute the *disinfo-debunk* metric. The computation of the *disinfo-debunk* metric is limited by the accuracy of the BERT(SJ) model in predicting the labels.

5.3 Evaluation

We have used the LIAR-PLUS-PLUS dataset as our evaluation dataset. We have edited the half-true and false claims after grouping the six labels into three labels. We have 7433 half-true and false claims in total.

[6] ChatGPT

Model	CP	Disinfo-debunk
Tailor	0.76	4243 / 6844 (62%)
GPT	0.52	223 / 7433 (3%)
RoBERTa	0.82	2081 / 7433 (28%)
ChatGPT	0.78	5426 / 7433 (73%)
Our Technique (T5)	**0.88**	6318 / 7433 **(85%)**

Table 4: Evaluation of Language models vs. our technique on the LIAR-PLUS-PLUS dataset for the task of claim editing. Abbreviations CP:= Content Preservation

5.3.1 Quantitative

Our technique has the highest average **content preservation** score (BLEU score) of **0.88** (on a scale of 0-1). The average content preservation score is the average of BLEU scores between all the edited claims and the corresponding original claim. We found that our technique can debunk half-true and false claims with a **disinfo-debunk** score of **85%**. Clearly, our technique outscores cutting-edge systems such as GPT by 82%, Roberta by 57%, ChatGPT by 12%, and Tailor by 23%. Out of 7433 claims, we obtained edited claims for 6844 claims using Tailor. The reason is that Tailor has a perplexity cutoff threshold. If the perplexity of the generated claim is lesser than this cutoff, we don't get any output from Tailor. Hence, for Tailor, the metrics are computed on 6844 claims. The results can be found in Table 4.

5.3.2 Qualitative

In addition to the qualitative evaluation of edited claims, we performed the human evaluation of the edited claims on two metrics, fluency, and edit-correctness.
Fluency (On a scale of 1-3):
Less fluent and incorrect grammar- score of 1
Medium level fluency- score of 2
Fluent and grammatically correct- score of 3
Edit-correctness (On a scale of 1-3):
Incorrect edit- score of 1 (Edited the incorrect part)
Partially correct edit- score of 2 (Edited the right part but not correctly edited)
Correctly edited- score of 3
We have manually annotated 7433 claims edited by the T5 model. Two annotators annotated for fluency and edit-correctness. The inter-annotator agreement, Cohen's Kappa ((Artstein and Poesio, 2008), was found to be **74.86** for fluency and **66.28** for edit-correctness. The average fluency was **2.75** and the average edit-correctness is **2.3**.

We also wanted to compare our technique with Tailor. We randomly picked 5000 instances and the corresponding edited claims edited using Tailor and T5. We did not mention which claim is edited by which model. Hence, we avoid biased responses. We shared the annotation guideline with the **4 users** and asked them to rate the edited claims for two metrics, fluency and edit-correctness. For claims edited by the T5 model, we found the average fluency was **2.68** and the average edit-correctness was **2.42**. For claims edited by Tailor, we found the average fluency was **2.28** and the average edit-correctness was **1.36**. Our technique performs better than Tailor in human evaluation too.

6 Conclusion and Future work

Our study demonstrates that T5 surpasses sophisticated techniques like Tailor and ChatGPT in effectively debunking half-truths through controlled claim editing. T5's ability to accurately fill in the necessary information with minimal edits contributes to its superior performance. Accurate detection of half-truths can be achieved with good accuracy, aided by the creation of a shortened justification column. The utilization of textual entailment and SRL-tagged evidence highlights the significance of NLP models in understanding linguistic properties.

Moving forward, our future plans involve creating and annotating a dataset specifically for gathering a larger quantity of high-quality data on half-truths from news articles. We aim to develop a novel algorithm using reinforcement learning to select and rank phrases within a claim for editing purposes. Additionally, we aspire to build a reliable and trustworthy real-time evidence extraction module to facilitate the detection and debunking of disinformation. While we have successfully debunked half-true and false claims using evidence from the LIAR-PLUS-PLUS dataset, the same cannot be said for real-time half-truths. To address this, we have developed a real-time evidence extraction module that retrieves results from Google News and extracts article summaries. Summaries from reputable sources can serve as valuable evidence. However, further large-scale testing is ongoing.

7 Limitations

One of the major challenges that we have come across is the lack of adequate data in the LIAR-PLUS dataset. The NLI model used for creating the shortened justification column in the LIAR-PLUS-PLUS dataset is not 100% accurate. Hence, we performed manual evaluation of the newly created column. The masking algorithm needs improvement since we are using only the NLI model and similarity score to mask claims. We would like to train a model to mask half-true part of the claim instead of the masking algorithm.

Acknowledgements

We thank the Politifact website and its annotators for their hard work in annotating claims. We also thank the annotators who participated in the experiments and submitted their valuable responses.

Ethics Statement

All the annotators in the experimental study gave their consent on submitting their responses. The annotation guideline was provided to them. There was anonymity maintained in the collection of responses. No personal data was collected during our experiments.

References

Tariq Alhindi, Savvas Petridis, and Smaranda Muresan. 2018. Where is your evidence: Improving fact-checking by justification modeling. In *Proceedings of the First Workshop on Fact Extraction and VERification (FEVER)*, pages 85–90, Brussels, Belgium. Association for Computational Linguistics.

Hunt Allcott and Matthew Gentzkow. 2017. Social media and fake news in the 2016 election. Working Paper 23089, National Bureau of Economic Research.

Ron Artstein and Massimo Poesio. 2008. Inter-coder agreement for computational linguistics. *Comput. Linguist.*, 34(4):555–596.

Pepa Atanasova, Jakob Grue Simonsen, Christina Lioma, and Isabelle Augenstein. 2020. Generating fact checking explanations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7352–7364, Online. Association for Computational Linguistics.

Sarah Cohen, Chengkai Li, Jun Yang, and Cong Yu. 2011. Computational journalism: A call to arms to database researchers. In *CIDR*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Andrew Estornell, Sanmay Das, and Yevgeniy Vorobeychik. 2019. Deception through half-truths. *CoRR*, abs/1911.05885.

Matt Gardner, Yoav Artzi, Victoria Basmova, Jonathan Berant, Ben Bogin, Sihao Chen, Pradeep Dasigi, Dheeru Dua, Yanai Elazar, Ananth Gottumukkala, Nitish Gupta, Hanna Hajishirzi, Gabriel Ilharco, Daniel Khashabi, Kevin Lin, Jiangming Liu, Nelson F. Liu, Phoebe Mulcaire, Qiang Ning, Sameer Singh, Noah A. Smith, Sanjay Subramanian, Reut Tsarfaty, Eric Wallace, Ally Zhang, and Ben Zhou. 2020. Evaluating NLP models via contrast sets. *CoRR*, abs/2004.02709.

Max Glockner, Vered Shwartz, and Yoav Goldberg. 2018. Breaking NLI systems with sentences that require simple lexical inferences. *CoRR*, abs/1805.02266.

Chuan Guo, Juan Cao, Xueyao Zhang, Kai Shu, and Miao Yu. 2019. Exploiting emotions for fake news detection on social media. *CoRR*, abs/1903.01728.

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

Naeemul Hassan, Chengkai Li, and Mark Tremayne. 2015. Detecting check-worthy factual claims in presidential debates. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, CIKM '15, page 1835–1838, New York, NY, USA. Association for Computing Machinery.

Bing He, Caleb Ziems, Sandeep Soni, Naren Ramakrishnan, Diyi Yang, and Srijan Kumar. 2021. Racism is a virus: Anti-asian hate and counterspeech in social media during the covid-19 crisis. In *Proceedings of the 2021 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, pages 90–94.

Yingya Li, Jieke Zhang, and Bei Yu. 2017. An NLP analysis of exaggerated claims in science news. In *Proceedings of the 2017 EMNLP Workshop: Natural Language Processing meets Journalism*, pages 106–111, Copenhagen, Denmark. Association for Computational Linguistics.

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

Rafael Monteiro, Roney Santos, Thiago Pardo, Tiago Almeida, Evandro Ruiz, and Oto Vale. 2018. *Contributions to the Study of Fake News in Portuguese: New Corpus and Automatic Detection Results: 13th International Conference, PROPOR 2018, Canela, Brazil, September 24–26, 2018, Proceedings*, pages 324–334.

Federico Monti, Fabrizio Frasca, Davide Eynard, Damon Mannion, and Michael M. Bronstein. 2019. Fake news detection on social media using geometric deep learning. *CoRR*, abs/1902.06673.

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

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

Kashyap Popat, Subhabrata Mukherjee, Andrew Yates, and Gerhard Weikum. 2018. DeClarE: Debunking fake news and false claims using evidence-aware deep learning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 22–32, Brussels, Belgium. Association for Computational Linguistics.

Alec Radford and Karthik Narasimhan. 2018. Improving language understanding by generative pre-training.

Alexis Ross, Tongshuang Wu, Hao Peng, Matthew Peters, and Matt Gardner. 2022. Tailor: Generating and perturbing text with semantic controls. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3194–3213, Dublin, Ireland. Association for Computational Linguistics.

Alexis Ross, Tongshuang Wu, Hao Peng, Matthew E. Peters, and Matt Gardner. 2021. Tailor: Generating and perturbing text with semantic controls. *CoRR*, abs/2107.07150.

Yves Scherrer. 2020. TaPaCo: A corpus of sentential paraphrases for 73 languages. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 6868–6873, Marseille, France. European Language Resources Association.

Tal Schuster, Adam Fisch, and Regina Barzilay. 2021. Get your vitamin C! robust fact verification with contrastive evidence. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 624–643, Online. Association for Computational Linguistics.

Gautam Kishore Shahi and Durgesh Nandini. 2020. Fakecovid - A multilingual cross-domain fact check news dataset for COVID-19. *CoRR*, abs/2006.11343.

Qi Su, Mingyu Wan, Xiaoqian Liu, and Chu-Ren Huang. 2020. Motivations, methods and metrics of misinformation detection: An nlp perspective. *Natural Language Processing Research*, 1:1–13.

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

Stephen E. Toulmin. 2003. *The Uses of Argument*, 2 edition. Cambridge University Press.

William Yang Wang. 2017. “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)*, pages 422–426, Vancouver, Canada. Association for Computational Linguistics.

Adina Williams, Nikita Nangia, and Samuel R. Bowman. 2017. A broad-coverage challenge corpus for sentence understanding through inference. *CoRR*, abs/1704.05426.

Dustin Wright and Isabelle Augenstein. 2021. Semi-supervised exaggeration detection of health science press releases. *CoRR*, abs/2108.13493.

A Appendix

A.1 NLI model

We have used the NLI (Natural Language Inference) model for extending the LIAR-PLUS dataset. For the details related to the usage of this model, refer to section 3.1.1. We have trained a BERT-based NLI model using SNLI (Glockner et al., 2018) and MNLI (Williams et al., 2017) datasets. We took a combination of 300,000 instances from both datasets as a training split to train the NLI model for the task of producing entailment scores with validation and a test split of 5000 instances each. The average F1 score is 0.91. The model can predict the textual entailment label with 91% accuracy. The label-wise F1 scores for entailment, contradiction, and neutral are 93%, 92%, and 87% respectively. The accuracy of this model is one of the limitations. Since, it is not fully accurate, the shortened justification might miss out relevant information in few instances. Hence we did a manual evaluation and found that the effectiveness of the obtaining shortened justification is extremely promising.

A.2 Claim editing using multiple Language models

This section presents detailed information about using LLMs (Large Language Models) for the task of claim editing discussed in section 4. An example of claim editing using LLMs is presented in Figure 3.

Claim: The number of illegal immigrants could be 3 million.
Counter: Every credible estimate we found was in the 11 million range, with a margin of error of around 1 million.
Masked Claim: The number of illegal immigrants <extra_id_0> <extra_id_1>.
SRL-tagged counter: [ARG1: Every credible estimate we found] [V: was] [ARG2: in the 11 million range], [ARGM-ADV: with a margin of error of around 1 million].
Extracted Justification: Trump said the number of illegal immigrants "could be 3 million. It could be 30 million. "Both figures are not within the range of possibility. Every credible estimate we found was in the 11 million range, with a margin of error of around 1 million. The figure has "always" been 11 million, in recent years, because of the flow of undocumented immigrants in and out of the United States.
Tailor:
Input: SRL-tagged counter + Masked Claim
Output: The number of illegal immigrants estimated at 11 million.
GPT:
Input: The number of illegal immigrants
Output: The number of illegal immigrants could be even more alarming, given many of them came to the United States just to work legally in one of the most dangerous areas on earth.
RoBERTa:
Input: Counter + Masked Claim
Output: The number of illegal immigrants could be trumped.
ChatGPT:
Input: Claim + Counter + Instructions
Output: The number of illegal immigrants could be 11 Million.
Tailor:
We call a function ***perturb_with_context*** with Claim, masked part of the claim, and counter.
Output: The number of illegal immigrants could be in the 11 million range, with a margin of error of around 1 million.

Figure 3: Claim Editing using LLMs

A.2.1 GPT

We have provided a three-fourths length of the claim (masked the rest of the claim) and the complete evidence as a prompt to the GPT model. The generated output from GPT is considered the edited claim. The idea behind this strategy is to check how effectively GPT can use the evidence and fill the masked part of the claim.

A.2.2 RoBERTa

We have used the RoBERTa-based transformer model from Hugging Face[7] for the task of text infilling. We have provided the masked claim, provided to the T5 model, concatenated with the evidence as input. The masked positions of the claim will be filled by this model and the filled claim is considered the edited claim.

A.2.3 ChatGPT

We have used the ChatGPT model for generating edited claims. We provided the half-true and false claims and the corresponding evidence. We asked ChatGPT to edit the given claim using the provided evidence minimally to make the claim correct. The generated claim is considered as the edited claim.

A.2.4 Tailor

We have used Tailor (Ross et al., 2022) for perturbing the claim by maximizing the counter. We have used the perturb_with_context function from Tailor. This function is used to fill the masked part of the claim using the counter. The inputs for the function are, the claim, part of the claim which needs to be edited, and the counter. The perturbed claim is considered the edited claim.

[7] Hugging Face