Logical Satisfiability of Counterfactuals for Faithful Explanations in NLI

Anonymous NAACL submission

Abstract

Evaluating an explanation's faithfulness is desired for many reasons such as trust, interpretability and diagnosing the sources of model's errors. In this work, which focuses on the NLI task, we introduce the methodology of Faithfulness-through-Counterfactuals, which first generates a counterfactual hypothesis based on the logical predicates expressed in the explanations, and then evaluates if the model's prediction on the counterfactual is consistent with that expressed logic (i.e. if the new formula is *logically satisfiable*). In contrast to existing approaches, this does not require any explanations for training a separate verification model. We first validate the efficacy of automatic counterfactual hypothesis generation, leveraging on the few-shot priming paradigm. Next, we show that our proposed metric performs well compared to other metrics using simulatability studies as a proxy task for faithfulness. In addition, we conduct a sensitivity analysis to validate that our metric is sensitive to unfaithful explanations.

1 Introduction

How should we evaluate an explanation's *faithfulness* with respect to the task model? According to Jacovi and Goldberg (2020), faithful measures should focus on utility to the user and the idea that an explanation can be *sufficiently faithful*.[1] Fundamentally, the goal of interpretability research is to build user trust, identify the influence of certain variables and allow users to understand how a model will behave on given inputs (Doshi-Velez and Kim, 2017; Lipton, 2018).

In interpretable NLP, there is growing interest in tasks that require world and commonsense "knowledge" and "reasoning" (Danilevsky et al., 2020).

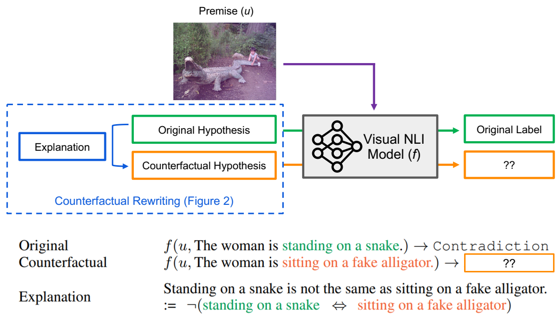

Figure 1: Overview of the proposed FTC approach, evaluating **f**aithfulness of explanations **t**hrough **c**ounterfactuals. If the explanation is faithful to the model, the NLI label on the new counterfactual hypothesis should change to `Entailment`. If the model still predicts `Contradiction`, this indicates that the explanation is not faithful to the model, i.e. the logic of the explanation and the model are not consistent.

We focus on natural language inference (SNLI; Bowman et al. (2015)), where extractive explanations also known as rationales (DeYoung et al., 2019) are limited as they take a subset of the existing input. Instead, we require *free-form natural language explanations* to fill in the reasoning or knowledge gap for such tasks (Camburu et al., 2018; Rajani et al., 2019). Our setting is thus characterised by the post-hoc interpretation of black-box classification models via generative explanations. Our work follows the standard "predict-and-explain" paradigm (Do et al., 2020). Here an explanation generator generates the explanations conditioned on the predicted task label.[2] Without faithfulness evaluations, the explanation approximately describes the internal process at best, and is generated from superficial similarities between the training data and the class label at worst.[3]

The central contribution of this paper is a

[1] Jacovi and Goldberg (2020) originally posit that faithful explanations should "accurately represents the reasoning process behind the model's prediction", however also acknowledge that this is "impossible to satisfy fully".

[2] Do et al. (2020) report little to no difference between jointly predict and explain compared to predict then explain. Note that the emphasis of this work is on evaluating the faithfulness of explanations rather than generating them.

[3] Preliminary experiments show that flipping the input class label can change the form of the explanation.

methodology grounded in first-order free logic (Lambert, 1967; Bencivenga, 2002),[4] to verify a given explanations' faithfulness. Our proposed approach generates a revised (counterfactual) hypothesis based on the logical propositions expressed in the explanation, and evaluates the *logical satisfiability* (Boolos et al., 2002) of the new hypothesis. Consider the following example:

Hypothesis:	The dog is barking at the girl.
Explanation:	The dog is an animal.
Counterfactual:	The animal is barking at the girl.

If the explanation is logically consistent (faithful) to the model, then the revised counterfactual hypothesis which replaces 'dog' with 'animal' in the original hypothesis, should be satisfiable, since 'dog is an animal'. However, if the explanation is inconsistent with the model, then the resulting hypothesis is unsatisfiable and the explanation is unfaithful. We describe this formally in Section 2.

Compared to previous automatic metrics (LAS; Hase et al. (2020), LRA; Wiegreffe et al. (2020)), our proposed method does not rely on an external verification model and therefore does not require explanation data for training.[5] Our method directly queries the task model in question while crucially avoiding the confound of "label leakage"[6] from the explanation (Hase et al., 2020). We expand on this discussion in Section 4. The contributions of this work are as follows:

- We present a methodology for evaluating faithfulness of free form explanations for NLI, grounded in first-order free logic (Subsection 2.1, Subsection 2.2). Our method evaluates the satisfiability of logical relations expressed through a counterfactual hypothesis.

- We introduce an automatic metric (Subsection 2.3) and show its viability with human studies, indicating that a practical solution exists for the proposed (theoretical) method. We leverage few-shot priming for generating counterfactual hypothesis, achieving $0.71-0.88$ METEOR score for human and generated explanations (Subsection 3.2).

- We show a strong effect size for simulatability of the counterfactual hypothesis as a proxy test of faithfulness, and achieve $0.69-0.78$ ρ-statistic on Wilcoxon rank-sum test (Subsection 3.3). A further sensitivity analysis indicates that our method is sensitive to pathological explanations that were generated by removing inputs to the explanation generator, as compared to other existing faithfulness metrics (Subsection 3.4).

[4]This is an extension of predicate logic and should not be confused with either predicate or propositional logic.

[5]"Faithfulness" measures which are tied to an external verification model are potentially problematic as given a fixed task model and explanation, one could in theory achieve two different faithfulness scores if the verification model changes.

[6]Label leakage occurs because of superficial similarities between the syntatic form of the explanation and the task label. For instance the explanations "A is a B" and "A is not a B" are highly associated with the Entailment and Contradiction label.

Label	Propositions	Description
E	$u \Rightarrow x$	hypothesis implied by premise
C	$u \Rightarrow \neg x$	hypothesis contradicts premise
N	$u \overset{(?)}{\Longrightarrow} x$	hypothesis neither contradicts or is entailed by premise

Table 1: Mapping NLI task labels to propositions. $\Rightarrow$ indicates logical implication, $\neg$ indicates logical negation, and $\overset{(?)}{\Longrightarrow}$ indicates *truth-valueless*.

2 Method

2.1 Problem Formulation

Natural Language Inference (NLI) is typically cast as a classification task; given a premise u and a hypothesis x, the classifier f predicts the label y, where $y \in \{\mathtt{E}, \mathtt{C}, \mathtt{N}\}$. Here E indicates entailment, C contradiction, and N neutral, for the relationship between u and x. f can therefore be viewed as a black-box function approximating the solution to a *logical satisfiability* problem.

Testing Predicate Relations in Explanations. An explanation z can express one or more logical predicate relations (R), which describes the relationship between two variables A and B that are expressed in x and u respectively (see Table 2 col 4 for examples). This is denoted as $R(A, B)$.

A "faithful" explanation with respect to a task model f, is one that expresses predicate relations $R(A, B)$ that are *consistent* with f's predictions. The central idea of this work, is to automatically verify this using a *counterfactual hypothesis*, x^{cf} and its derived associated *counterfactual label* (the expected satisfiability result). If $f(u, x^{\text{cf}})$ does not result in the associated counterfactual label, then the explanation is not faithful to f.

First-order Free Logic in NLI. In order to deduce the associated label for x^{cf}, we must address

Label (y)	Propositional Formula	Explanation (z)	$R(A,B)$	Propositional Formula (cf)	Label (y^{cf})
E	$u \Rightarrow x$	A is the same as B	$A \Leftrightarrow B$	$u \Rightarrow x^{\text{cf}}$	E
C	$u \Rightarrow \neg x$	A is not B	$\neg(A \Leftrightarrow B)$	$u \Rightarrow \neg(\neg x^{\text{cf}})$	E
N	$u \overset{(?)}{\Longrightarrow} x$	A does not imply B	$\neg(A \Rightarrow B)$	$u \Rightarrow x^{\text{cf}}_{[A]}$	E
				$u \overset{(?)}{\Longrightarrow} x^{\text{cf}}_{[B]}$	N

Table 2: From the original hypothesis x and logical predicate relations $R(A,B)$ expressed in the explanation, we generate the counterfactual hypothesis x^{cf} (2 cases $x^{\text{cf}}_{[A]}$ and $x^{\text{cf}}_{[B]}$ for $y = \text{N}$). The resulting counterfactual label y^{cf} is logically derived from the propositions Subsection 2.2. Table 3 shows examples of this process.

the issue of logical deduction for `Neutral`, which has no corresponding expression in classical predicate logic. We observe that the ternary label in NLI parallels first-order free logic, which has three distinct logical forms, positive, negative and neutral (Lambert, 1967; Nolt, 2021). In contrast to classical logic which requires each singular term to denote a Boolean variable in the domain, free logic may have formula which are *truth-valueless* (Nolt, 2021), i.e., it is not known whether they are True or False.[7] Table 1 shows the task label, propositions and their meaning in Free Logic.

2.2 Satisfiabilty of the Counterfactual

In this section, we derive what the counterfactual hypothesis and associated counterfactual label would be for each original label assuming logical formulas and discrete variables where exact substitution is possible. Subsection 2.4 describes our suggested approach to handle Natural Language where discrete substitution no longer holds.

Axiom 1. *Substitution for formulas (Fitting, 2012) For any variables A and B and any formula $x_{[A]}$ containing A, if $x_{[B\backslash A]}$ is obtained by replacing any number of free occurrences of A in x with B, then for $A \Leftrightarrow B$, $x_{[A]} \Leftrightarrow x_{[B\backslash A]}$.*

Assumption 1. *The hypothesis x and premise u contain n free variables, of which the variables A and B are members in u and x. We denote the membership of $A \subseteq x$ as $x_{[A]}$.*

Assumption 2. *Given $R(A,B)$, the counterfactual hypothesis can be constructed by applying Axiom 1 replacing A with B, denoted $x_{[B\backslash A]} = x^{cf}$.*[8]

Assumption 3. *The predicate relation expressed in $R(A,B)$ is a sufficient condition, to explain the model's predicted label.*

We formally derive the associated counterfactual label y^{cf} (expected satisfiability result) of the new counterfactual hypothesis x^{cf}. Figure 1 and Table 3 show examples for this process. The proofs follow the following high-level structure:

1. Substitution of variables A and B (Axiom 1 and Assumption 1) to construct x^{cf} (Assumption 2).

2. Examine the predicate relationship $R(A,B)$ and derive the logical relationship between the x^{cf} and the premise u.

Proposition 1. *If the original label is* E*, then the associated counterfactual label is* E*.*

Proof. By Assumption 1, the logical proposition represented by E is $u \Rightarrow x_{[A]}$. Since $A \Leftrightarrow B$, then $u \Rightarrow x_{[B\backslash A]}$, by Axiom 1, $u \Rightarrow x^{\text{cf}}$. Therefore we have the resulting propositional formula and associated counterfactual label $(u \Rightarrow x) \Leftrightarrow (u \Rightarrow x^{\text{cf}})$, i.e. $(y = \text{E}) \Leftrightarrow (y^{\text{cf}} = \text{E})$. □

Proposition 2. *If the original label is* C*, then the associated counterfactual label is* E*.*

Proof. By Assumption 1, the logical proposition represented by C is $(u \Rightarrow \neg x_{[A]})$. Since $\neg(A \Leftrightarrow B)$ is equivalent to $(A \Leftrightarrow \neg B)$, then by Axiom 1, $u \Rightarrow \neg(x_{[\neg B\backslash A]})$. However it is not possible to test for the "negation of" variables as negation is not seen in training data for NLI. Under Assumption 3,[9] if the explanation $\neg(A \Leftrightarrow B)$ sufficiently explains the label, then having $x_{[B\backslash A]}$ negates (or 'flips') the label to Entailment $u \Rightarrow \neg\neg(x_{[B\backslash A]})$. Therefore we have the result $(u \Rightarrow \neg x) \Leftrightarrow (u \Rightarrow \neg\neg(x^{\text{cf}}))$, i.e., $(y = \text{C}) \Leftrightarrow (y^{\text{cf}} = \text{E})$. □

Proposition 3. *If the original label is* N*, then there are **two** associated counterfactual labels,* E*, and* N*.*

[7] Truth-valueless formulas are often said to have "truth-value gaps". Informally, this can be interpreted as there being insufficient information on the truth-values of logical variables to conclude the relationship between u and x (Nolt, 2021).

[8] Equivalence formulas may be substituted for one another without changing that formula's truth value (Fitting, 2012).

[9] The violation of Assumption 3, can result in partially 'unfaithful' explanations that do not provide enough information to explain model prediction.

	Original	$f(u,$ There are people playing the piano$)$ $\rightarrow$ `Contradiction`
	Counterfactual	$f(u,$ There are people playing woodwind instruments.$)$ $\rightarrow$ `Entailment`
	Explanation	*"The people in the photo are not playing the piano. They are instead playing other woodwind instruments."* := $\neg$(playing the piano $\Leftrightarrow$ playing other woodwind instruments)
	Original	$f(u,$ Two skateboarders navigate a curve.$)$ $\rightarrow$ `Entailment`
	Counterfactual	$f(u,$ Two skateboarders round a curve.$)$ $\rightarrow$ `Entailment`
	Explanation	*"To navigate a curve is to round it."* := (navigate a curve $\Leftrightarrow$ round a curve)
	Original	$f(u,$ People are shopping in the city. $\rightarrow$ `Neutral`
	Counterfactual	$f(u,$ People are walking down a city sidewalk.$)$ $\rightarrow$ `Entailment`
	Counterfactual	$f(u,$ People are shopping.$)$ $\rightarrow$ `Neutral`
	Explanation	*"People can walk down a city sidewalk for reasons other than shopping."* := $\neg$(walk down a city sidewalk $\Rightarrow$ shopping)

Table 3: Examples of counterfactual hypothesis rewrites from the explanation in SNLI-VE dataset (Do et al., 2020). $f(u, x) \rightarrow y$ shows the expected model prediction given the premise (u) and either original or counterfactual hypothesis. Note that `Neutral` can still result in `Neutral` (Subsection 2.2: Proposition 3).

Two conditions arise because $\neg(A \Rightarrow B) \Leftrightarrow A$ AND $\neg B$. In the interest of space, we present the proof for Proposition 3 in the Appendix. Table 2 summarises the original propositional formula in the hypothesis x and premise u, the predicate relations $R(A, B)$ and the associated y^{cf}.

2.3 Proposed Metric: FTC

We introduce the metric Faithfulness-Through-Counterfactuals (FTC), to capture the difference in model predicted probabilities $p(\hat{y}^{\text{cf}})$ from the associated counterfactual label y^{cf}.

$$\text{FTC} = 1 - d(p(\hat{y}^{\text{cf}}), p(y^{\text{cf}})) \tag{1}$$

Choice of Distance Function (d) We consider three metrics for d, $\mathbb{1}[\text{argmax}(p(\hat{y}^{\text{cf}})) == y^{\text{cf}}]$ denoted FTC-δ, KL Divergence (FTC-$\mathcal{K}$) and Wasserstein distance (FTC-$\mathcal{W}$) with symmetrical distance of 1 between `E` and `C`, and $0 \geq \alpha \geq 1$ between `N` and the two other labels.

2.4 Generating Counterfactual Hypothesis

In the previous section, we had assumed that the logical variables A and B are substitutable in x directly. Indeed generating a counterfactual hypothesis would be trivial if A and B could be directly extracted from the explanation, and directly substituted in the hypothesis.[10] However, open domain semantic parsing is an unsolved problem (Lee et al., 2021) of which to our knowledge, there is no off-the-shelf solution which does not require fine-tuning on a train set.[11] Hence, we propose to leverage on advances in few-shot priming (Brown et al., 2020) which only requires several handwritten examples and no further fine-tuning. This is compared against the baseline of parsing via 'extract and transform' using regex (experiment described in Subsection 3.2).

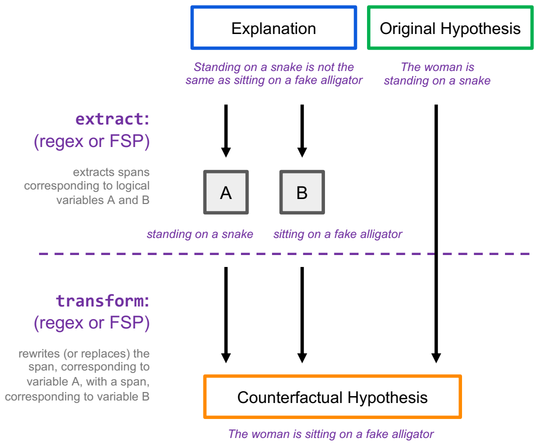

Figure 2: Two step counterfactual rewriting of the hypothesis, according to the explanation. We implement extract and transform steps using regular expressions (regex) and few-shot priming (FSP) models.

Extract and Transform with Regex We adopt and extend templates identified by Camburu et al. (2019) for explanations, who noted that these templates are a "natural consequence of the task and

[10] Consider the hypothesis: "the boy is outside" and the explanation: "A tire swing is usually installed outside". Naive substitution would result in "the boy is a tire swing".

[11] We experimented with dependency parses from Spacy and Stanford NLP which gave many irrelevant extractions.

dataset". Regex methods perform a rule-based span extraction and replacement (we allow for stemmed word replacement). Extraction rules are shown in Appendix (Tables 15 to 17).

Extract and Transform with Few-shot Priming We compare the brittle regex pipeline with the modern paradigm of in-context few-shot priming. This is an attractive option in our setting where we do not have prior training data for generating counterfactual hypothesis, and the task appears to be related to manipulation of text strings.

We thus consider a two step process of 1) extracting the logical spans, i.e., A and B in Table 2 from the explanation, and 2) modifying the hypothesis given these extracted spans.[12] Given a sequence of priming examples of how to extract spans from the explanation in the prefix, the model should perform the extraction given a test explanation.

Reducing Natural Language Artifacts with x^{cf} Previous attempts to test the explanation directly as input to a trained model, are subject to confounds of "label leakage" because of the close association between label and syntatic form of the explanation (Pruthi et al., 2020). Crucially, our proposed method sidesteps this confound by applying *logical satisfiability* checks via predicate relations $R(A, B)$. In theory, the construction of x^{cf} should preserve the syntatic structure of the original hypothesis, while only changing the semantics of A and B. Potential limitations of our method are discussed in Appendix: Subsection 6.7.

3 Experiments and Results

We validate our automatic method through three sets of experiments.

i) Evaluating the quality of generated counterfactual hypothesis from a few-shot generator $\mathcal{H}^{\text{model}}$ (Subsection 3.2).

ii) Evaluating the proposed metric (FTC) on simulatability of human generated x^{cf}, and comparing this to existing faithfulness metrics in the literature (Subsection 3.3).

iii) Studying the sensitivity of our proposed approach compared to other metrics given pathological explanations (Subsection 3.4).

[12] This two step-process is necessary as preliminary experiments show that even few-shot priming models with 175B parameters (academic access GPT3) are not able to construct counterfactual hypothesis in a single step.

3.1 Experimental Setup

Datasets We consider logical entailment datasets, e-SNLI (Camburu et al., 2018) and e-SNLI-VE (Do et al., 2020) which are the only explainable logical entailment datasets available at point of writing (Wiegreffe and Marasović, 2021). e-SNLI consists of crowdsourced explanations for SNLI. e-SNLI-VE replaces the textual premise u of SNLI with Flickr30k images (Young et al., 2014). To avoid trivial word overlap between u and x^{cf}, we adopt the image representation for the premise u in our experiments for Subsection 3.3 and Subsection 3.4. Note that Subsection 3.2 only requires x and z.[13]

Models For the task model f, we adopt a state-of-art multimodal model, `CLIP` (Radford et al., 2021), and fine-tune a 2-layer MLP to train a predictor $f(u, x) \to y$. For the explanation generator, g, we follow Do et al. (2020) and fine-tune a modified GPT2 (Radford et al., 2019). Training details are in Appendix: Subsection 6.3. For the counterfactual hypothesis generator, $\mathcal{H}^{\text{model}}$ we adopt a pretrained GPT2-XL and GPT-Neo1.3B and 2.7B (Black et al., 2021) without further fine-tuning, and apply only handwritten prompts. Prompt examples were randomly sampled from the training set and we used 20 prompts for each label.[14]

3.2 Quality of Counterfactual Hypothesis

As the feasibility of our automatic approach depends on the quality of counterfactual generation, we evaluate $\hat{x}^{\text{cf}} \leftarrow \mathcal{H}^{\text{model}}(x, z)$ against gold counterfactuals, $x^{\text{cf}*} \leftarrow \mathcal{H}^{\text{human}}(x, z)$, where $\mathcal{H}^{\text{human}}$ refers to the human annotator, $\mathcal{H}^{\text{model}}$ refers to our automated hypothesis generator, z refers to explanations and x refers to the original hypothesis. We randomly sample 300 examples from the validation set (100 each for `E`, `C`, `N`) and ask annotators to write counterfactual hypothesis for human generated explanations, z^* and model generated explanations $\hat{z}$. Annotators are asked to revise x such that the logic in z is expressed in the new counterfactual x^{cf} (Appendix:Figure 3). We show annotators the *same* set of examples that were used to prompt $\mathcal{H}^{\text{model}}$. We obtain three annotations per datapoint for multiple reference sentences.[15]

[13] All experiment code will be available at anonymous.io.

[14] Further details are available in Appendix:Subsection 6.3 and example of prompt templates(Appendix:Subsection 6.12)

[15] "A young boy wearing a white shirt on a beach" and "A young boy on a beach wearing a white shirt" are both valid.

Extract	Transform	Human Explanations $z^* \mapsto x^{\text{cf}}$ C	E	N$_{[A]}$	N$_{[B]}$	Generated Explanations $\hat{z} \mapsto x^{\text{cf}}$ C	E	N$_{[A]}$	N$_{[B]}$
regex	regex	0.690	0.638	0.010	0.012	0.744	0.701	0.473	0.418
regex	Neo2.7B	0.690	0.709	0.710	**0.752**	0.840	0.860	**0.805**	**0.714**
Neo2.7B	regex	0.801	0.691	0.509	0.565	**0.893**	0.781	0.669	0.584
Neo2.7B	Neo2.7B	**0.822**	**0.782**	**0.743**	**0.754**	0.870	**0.881**	**0.807**	0.709

Table 4: METEOR scores for Hypothesis Revision by either using regex, GPT-Neo, or a combination of both. Bold is applied column-wise. We show the breakdown of revised hypothesis by class label (Contradiction C, Entailment E and Neutral N), and also whether the explanations are human or model generated. Note that there are two counterfactual hypothesis generated for N, N$_{[A]}$ and N$_{[B]}$ which correspond to $x^{\text{cf}}_{[A]}$ and $x^{\text{cf}}_{[B]}$ as described in Subsection 2.2: Proposition 3.

Experiment Conditions As described in Subsection 2.4, we adopt a two-step process of 'extract' and 'transform' for generating $\hat{x}^{\text{cf}} \leftarrow \mathcal{H}^{\text{model}}(x, z)$. We compare different combinations of either extracting and transforming with regex or $\mathcal{H}^{\text{model}}$. Our main results for the largest and best performing model (GPT-Neo2.7B) is shown in Table 4, and additional results for GPT-Neo1.3B and GPT2-XL in Appendix: Table 11.

Metric The evaluation metric used is METEOR (Banerjee and Lavie, 2005), that was found to correlate with human judgement (Kayser et al., 2021). METEOR computes harmonic mean of unigram precision and recall and accounts for stemming. As the validity of any text generation metric is debatable (Deng et al., 2021), we further quantify the downstream effects of the automated process through Subsection 3.3.

Results (Presented in Table 4).

1. The combination of using models for both extract and transform steps (last row) performs best in all cases with human explanations z^*, and close to best with model explanations, $\hat{z}$.

2. The performance of most methods are better on $\hat{z}$ than z^* which might be explained by the more 'standardised' text format in $\hat{z}$. Brahman et al. (2020) reported that generator models tend to follow a similar format, supporting this interpretation. The row regex-regex can be seen as a direct comparison of how 'standardised' $\hat{z}$ is compared to z^* as it indicates the performance for a brittle rule-based approach.

3. For ease of rewriting each class (column-wise), C > E > N for z^* in most cases, which highlight the relative complexity (N tends to be expressed in a less 'straightforward' manner) of extracting and transforming hypothesis with different types of explanations.

3.3 Metric Validation via Simulatability of Counterfactual Hypothesis

As described in Section 1, faithfulness metrics should focus on utility to the user (Jacovi and Goldberg, 2020). One such practical utility of explanations is that humans should be able to simulate the model's predictions given the explanations (Doshi-Velez and Kim, 2017). However, instead of using the raw explanation which has reported issues of label leakage (Pruthi et al., 2020; Hase et al., 2020),[16] we consider simulatability on $x^{\text{cf}} \leftarrow \mathcal{H}^{\text{human}}(z^*, x)$ (obtained in Subsection 3.2).

There are two distinct outcome groups for the simulatability study; human annotators either agree or disagree with the model on x^{cf}. We use the Wilcoxon rank-sum test which is a nonparametric test for the null hypothesis that two groups are equal. If $\mathcal{H}^{\text{model}}$ had produced exactly the same x^{cf} as the human $\mathcal{H}^{\text{human}}$, then FTC (our metric) should be very effective at separating the two outcome groups, and the $\rho \in [0, 1]$ would be very close to 1. We compare with other faithfulness metrics namely Label Adjusted Simulation (LAS; Hase et al. (2020)) and Label Rationale Association (LRA; Wiegreffe et al. (2020)), reviewed in Section 4 and Appendix: Table 8.

We collect annotations for 100 data instances grouped by each original label-class, and obtain 3 annotations per instance.[17] Annotators are required to rate whether (x^{cf}, u) entails, contradicts, or is neutral (Appendix:Figure 4).

Results (Presented in Table 5)

1. FTC variants have the highest $\rho \in [0, 1]$ statis-

[16] Label leakage due to a nearly one-to-one correspondence with the linguistic form of explanations and the label.

[17] We measure the inter-annotator agreement (IAA) using Fleiss' kappa, achieving a "moderate" agreement (Landis and Koch, 1977) with $\kappa = 0.51$. More detailed IAA results are provided in Appendix: Subsection 6.6. We aggregate the final label using the majority vote.

	C	E	N$_{[A]}$	N$_{[B]}$
LAS	0.490	0.516	0.522	0.500
LRA	0.544	0.564	0.477	0.659*
FTC-δ	0.756*	0.757*	0.629*	0.731*
FTC-$\mathcal{K}$	0.788*	0.753*	0.687*	0.765*
FTC-$\mathcal{W}$	0.762*	0.751*	0.694*	0.765*

Table 5: ρ-statistic $\in [0, 1]$ for Wilcoxon rank-sum test on different faithfulness metrics. A larger ρ-statistic indicates a larger effect size. LRA: Label Adjusted Simulation, LRA: Label Rationale Agreement, and FTC: Faithfulness-by-Counterfactual (ours), and FTC-$\mathcal{W}$ and FTC-$\mathcal{K}$ ($\alpha = 0.7$) are the Wasserstein and Kl-divergence variants. * indicates significance (p-value< 0.05). The test is conducted between the two groups; data points where the human agrees vs disagrees with the model's prediction on counterfactual hypothesis.

tic (higher is better and indicates a larger effect size), which indicates that it is the most discriminative metric for whether the human can simulate the model's prediction given new counterfactual inputs. This is expected as the simulation procedure is similar to how FTC is calculated. However as reported in Subsection 3.2 discrepancies in the automated rewriting process affect the scoring of the metric in ways which are not easily captured by sentence generation metrics (Table 4). The results show that the best-automatic rewriting process of adopting GPT-Neo2.7B achieves 0.63-0.79 ρ-statistic across C, E, N, giving us an indication of the downstream impact of 0.74 to 0.82 METEOR score.

2. The ρ-statistic follows a similar trend to that previously observed in Subsection 3.2, C $>$ E $>$ N, which suggests that adopting METEOR Score to measure hypothesis rewrites is a reasonable measure of downstream performance.

3. KL-Divergence (FTC-$\mathcal{K}$) and Wasserstein Distance (FTC-$\mathcal{W}$) perform slightly better than the naive Identity function (FTC-δ). This indicates that a soft computation over probabilities can account for some of the errors in the x^{cf} rewriting process.

4. With the exception of LRA on N$_{[B]}$, we find that other metrics have ρ approximately 0.5 across C, E, N. This indicates that the metric is relatively poorer at distinguishing between the two outcome groups, i.e. poorer simulatability results.

Examining Inconsistent Explanations In cases where the human NLI label does *not* correspond to the logically derived NLI class in Table 2, our method suggests that the explanations are *not* faithful to the human's NLI model. We find that human NLI labels correspond to the derived counterfactual label 54% to 87% of the time (Appendix:Table 10).

x	u	y	BER	MET	LAS	LRA	FTC-$\mathcal{K}$
✓	✓	✓	0.888	0.245	0.047	0.788	0.126
✓			0.885	0.233	-0.035	0.561	-0.200
	✓		0.869	0.120	-0.063	0.298	-0.055
		✓	0.865	0.097	0.068	0.630	-0.245

Table 6: Raw scores for different metrics, by perturbing inputs to the explanation generator (sensitivity analysis). x, u, and y refers to hypothesis, premise, and label respectively. BER: BertScore, MET: METEOR, LAS: Label Adjusted Simulation, LRA: Label Rationale Association, FTC-$\mathcal{K}$ (ours).

Appendix:Subsection 6.9 shows examples of these cases, which we typically find to be due to explanations of low quality, supporting the central thesis of the paper. In the previous simulatability results, we filter out poor explanations.[18]

3.4 Sensitivity Analysis

As a sanity check, good faithfulness metrics should be sensitive towards unfaithful explanations, i.e. they should perform worse on unfaithful explanations compared to faithful ones. We perform a sensitivity analysis on various faithfulness metrics by examining their raw scores on unfaithful explanations *by construction*. These are constructed by leaving out all but one type of input to the explanation generator. The 'complete' set of inputs are the entailment label (y), hypothesis (x), and premise (u). Note that the 'complete' set of inputs to the explanation generator does not guarantee faithful explanations, but they are guaranteed to be less pathological than leaving out all but one type of input. We additionally consider BertScore (Zhang et al., 2019) and METEOR (Banerjee and Lavie, 2005) which are text similarity metrics evaluated against the human explanation, and use FTC-$\mathcal{K}$ variant which has an upperbound of 1 and a high ρ-statistic in the previous experiment.[19]

Results (presented in Table 6)
1. FTC-$\mathcal{K}$ performs consistently better for the non-pathological (first row 0.126) vs pathological explanations (-0.200, -0.055, -0.245). The same 'correct' trend is observed for LRA 0.788 vs (0.561, 0.298, 0.630).

[18] Quantifying the extent of annotation errors for explanations is outside of the scope of this work. We refer readers to Valentino et al. (2021) who report that explanations are valid logical relations 60% of the time and other times they are either redundant or non-sensical due to annotation errors.

[19] As described in Subsection 2.3, FTC-$\mathcal{K}$ is $1-$ the KL term, which has lower bound 0 and no upper bound. Hence the upper bound on FTC-$\mathcal{K}$ is 1.

2. We find that an off-the-shelf BertScore (Zhang et al., 2019) has a suprisingly low range of values for the different conditions (0.865 to 0.888). METEOR scores conditioned only on x are also very close to the full range of inputs (0.233 vs 0.245) indicating superficial word similarity of the explanations when conditioned on just x.

3. LAS scores are in the 'wrong' direction, namely that explanations generated with all of the relevant inputs perform worse than just having the label.

4 Related Work

We outline existing methods which evaluate free-text generated Natural Language explanations, their assumptions of faithfulness, and describe how they operationalise these assumptions. We focus on LAS and LRA, which are used in our experiments, and provided more discussion of related work in Appendix:Subsection 6.4.

4.1 Leakage Adjusted Simulation

This method assumes explanations are faithful if they allow a model to be more simulatable. A model is simulatable to the extent that an observer, or simulator, can predict its outputs (Doshi-Velez and Kim, 2017; Hase et al., 2020; Kumar and Talukdar, 2020). From this perspective, one might use a simulator's (either a human or model) accuracy with explanations as input, to measure explanation quality. However, as Hase et al. (2020) argues, the simulator's success does not reflect explanation quality when the explanation leaks the label to the simulator. They thus propose Leakage-adjusted Simulatability (LAS) which performs a macro-average of leakable and non-leakable explanations. However, a high occurrence of label leakage may overwrite the effect of macro-averaging (Pruthi et al., 2020).

4.2 Label Rationale Association

According to Wiegreffe et al. (2020), "at a minimum, rationales must be implicitly or explicitly tied to the model's prediction.". Their method tests whether label and explanations are similarly robust to noise in the input. Although designed to be highly generalisable to generative framework of explanations, this assumption may be overly general for more rigorous notions of faithfulness. Consider the scenario where merely changing the label to the generator results in "sufficiently different" explanations being generated (Kumar and Talukdar, 2020), whether or not the original explanation was actually faithful, LRA will assign this a high score.

4.3 Counterfactuals as Explanations

Our approach differs from the literature on Counterfactuals as explanations (Mothilal et al., 2020; Verma et al., 2020) as we do not generate counterfactual explanations, but generate counterfactual hypothesis based on the explantions. Camburu et al. (2019) work has a similar flavor, where they "reverse" a hypothesis. However, they focus on show the pathologies of a generator by searching for (adversarial) input hypothesis that cause the model to generate logically inconsistent explanations. Ge et al. (2021) also constructs 'counterfactual inputs', but search for existing features in the original input, and consequently is only applicable to extractive explanations.

4.4 Natural Logic vs Free Logic

Previous work on "Natural Logic" (MacCartney and Manning, 2007) relies on natural language features to guide inferences. For instance, changing specific terms to more general ones preserves entailment. This sidesteps the difficulties of translating sentences into First-order-Logic. Natural logic systems (Angeli and Manning, 2014) have been used in explainable fact verification (Krishna et al., 2021) which constructs "explanations" by presenting logical steps for inference. However these approaches still require a knowledge base to train or mine truth values, e.g, "in Paris" $\subseteq$ "in France". In contrast, our method does not require additional training and is a procedurally lightweight method relying on off-the-shelf pretrained models.

5 Conclusion

Measuring faithfulness of free-text explanations with respect to a task model is a challenging problem due to confounds introduced by testing explanations directly. In this work, we propose an approach to evaluating explanations for NLI tasks which uses the predicate logic expressed in explanations to construct counterfactual hypothesis, and tests the *satisfiability* of the resulting hypothesis. Our experiments on validating counterfactual hypothesis generation and simulatability of the counterfactual hypothesis show that our proposed automatic pipeline is a viable approximation to the theoretical method. Further, we show that our metric is sensitive to pathologically unfaithful explanations.

References

Gabor Angeli and Christopher D Manning. 2014. Naturalli: Natural logic inference for common sense reasoning. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 534–545.

Satanjeev Banerjee and Alon Lavie. 2005. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 65–72.

Ermanno Bencivenga. 2002. Free logics. In *Handbook of philosophical logic*, pages 147–196. Springer.

Sid Black, Leo Gao, Phil Wang, Connor Leahy, and Stella Biderman. 2021. Gpt-neo: Large scale autoregressive language modeling with mesh-tensorflow. *If you use this software, please cite it using these metadata*.

George S Boolos, John P Burgess, and Richard C Jeffrey. 2002. *Computability and logic*. Cambridge university press.

Samuel R Bowman, Gabor Angeli, Christopher Potts, and Christopher D Manning. 2015. A large annotated corpus for learning natural language inference. *arXiv preprint arXiv:1508.05326*.

Faeze Brahman, Vered Shwartz, Rachel Rudinger, and Yejin Choi. 2020. Learning to rationalize for nonmonotonic reasoning with distant supervision. *arXiv preprint arXiv:2012.08012*.

Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*.

Oana-Maria Camburu, Tim Rocktäschel, Thomas Lukasiewicz, and Phil Blunsom. 2018. e-snli: Natural language inference with natural language explanations. *arXiv preprint arXiv:1812.01193*.

Oana-Maria Camburu, Brendan Shillingford, Pasquale Minervini, Thomas Lukasiewicz, and Phil Blunsom. 2019. Make up your mind! adversarial generation of inconsistent natural language explanations. *arXiv preprint arXiv:1910.03065*.

Marina Danilevsky, Kun Qian, Ranit Aharonov, Yannis Katsis, Ban Kawas, and Prithviraj Sen. 2020. A survey of the state of explainable ai for natural language processing. *arXiv preprint arXiv:2010.00711*.

Mingkai Deng, Bowen Tan, Zhengzhong Liu, Eric Xing, and Zhiting Hu. 2021. Compression, transduction, and creation: A unified framework for evaluating natural language generation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7580–7605, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Jay DeYoung, Sarthak Jain, Nazneen Fatema Rajani, Eric Lehman, Caiming Xiong, Richard Socher, and Byron C Wallace. 2019. Eraser: A benchmark to evaluate rationalized nlp models. *arXiv preprint arXiv:1911.03429*.

Virginie Do, Oana-Maria Camburu, Zeynep Akata, and Thomas Lukasiewicz. 2020. e-snli-ve-2.0: Corrected visual-textual entailment with natural language explanations. *arXiv preprint arXiv:2004.03744*.

Finale Doshi-Velez and Been Kim. 2017. Towards a rigorous science of interpretable machine learning. *arXiv preprint arXiv:1702.08608*.

Melvin Fitting. 2012. *First-order logic and automated theorem proving*. Springer Science & Business Media.

Yingqiang Ge, Shuchang Liu, Zelong Li, Shuyuan Xu, Shijie Geng, Yunqi Li, Juntao Tan, Fei Sun, and Yongfeng Zhang. 2021. Counterfactual evaluation for explainable ai. *arXiv preprint arXiv:2109.01962*.

Peter Hase, Shiyue Zhang, Harry Xie, and Mohit Bansal. 2020. Leakage-adjusted simulatability: Can models generate non-trivial explanations of their behavior in natural language? *arXiv preprint arXiv:2010.04119*.

Alon Jacovi and Yoav Goldberg. 2020. Towards faithfully interpretable nlp systems: How should we define and evaluate faithfulness? *arXiv preprint arXiv:2004.03685*.

Maxime Kayser, Oana-Maria Camburu, Leonard Salewski, Cornelius Emde, Virginie Do, Zeynep Akata, and Thomas Lukasiewicz. 2021. e-vil: A dataset and benchmark for natural language explanations in vision-language tasks. *arXiv preprint arXiv:2105.03761*.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Amrith Krishna, Sebastian Riedel, and Andreas Vlachos. 2021. Proofver: Natural logic theorem proving for fact verification. *arXiv preprint arXiv:2108.11357*.

Sawan Kumar and Partha Talukdar. 2020. Nile: Natural language inference with faithful natural language explanations. *arXiv preprint arXiv:2005.12116*.

Karel Lambert. 1967. Free logic and the concept of existence. *Notre Dame Journal of Formal Logic*, 8(1-2):133–144.

J Richard Landis and Gary G Koch. 1977. The measurement of observer agreement for categorical data. *biometrics*, pages 159–174.

Celine Lee, Justin Gottschlich, and Dan Roth. 2021. Toward code generation: A survey and lessons from semantic parsing. *arXiv preprint arXiv:2105.03317*.

Zachary C Lipton. 2018. The mythos of model interpretability: In machine learning, the concept of interpretability is both important and slippery. *Queue*, 16(3):31–57.

Bill MacCartney and Christopher D Manning. 2007. Natural logic for textual inference. In *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, pages 193–200.

Ramaravind K Mothilal, Amit Sharma, and Chenhao Tan. 2020. Explaining machine learning classifiers through diverse counterfactual explanations. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, pages 607–617.

John Nolt. 2021. Free Logic. In Edward N. Zalta, editor, *The Stanford Encyclopedia of Philosophy*, Fall 2021 edition. Metaphysics Research Lab, Stanford University.

Danish Pruthi, Bhuwan Dhingra, Livio Baldini Soares, Michael Collins, Zachary C Lipton, Graham Neubig, and William W Cohen. 2020. Evaluating explanations: How much do explanations from the teacher aid students? *arXiv preprint arXiv:2012.00893*.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. *arXiv preprint arXiv:2103.00020*.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Nazneen Fatema Rajani, Bryan McCann, Caiming Xiong, and Richard Socher. 2019. Explain yourself! leveraging language models for commonsense reasoning. *arXiv preprint arXiv:1906.02361*.

Stuart Russell. 2015. Unifying logic and probability. *Communications of the ACM*, 58(7):88–97.

Marco Valentino, Ian Pratt-Hartman, and André Freitas. 2021. Do natural language explanations represent valid logical arguments? verifying entailment in explainable nli gold standards. *arXiv preprint arXiv:2105.01974*.

Sahil Verma, John Dickerson, and Keegan Hines. 2020. Counterfactual explanations for machine learning: A review. *arXiv preprint arXiv:2010.10596*.

Sarah Wiegreffe and Ana Marasović. 2021. Teach me to explain: A review of datasets for explainable nlp. *arXiv preprint arXiv:2102.12060*.

Sarah Wiegreffe, Ana Marasović, and Noah A Smith. 2020. Measuring association between labels and free-text rationales. *arXiv preprint arXiv:2010.12762*.

Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. 2014. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, 2:67–78.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

6 Appendix

6.1 Notation Table

f	task model (CLIP)
g	explanation generator (GPT2)
$\mathcal{H}$	counterfactual hypothesis generator
$\mathcal{H}^{\text{model}}$	counterfactual hypothesis generator (GPT-Neo)
$\mathcal{H}^{\text{human}}$	counterfactual hypothesis generator (human)
z^*	explanations (human)
$\hat{z}$	explanations (generator g)
u	premise
x	hypothesis (original)
x^{cf}	hypothesis (counterfactual)
y^*	task label (human)
$\hat{y}$	task label (model f)
y^{*cf}	task label on counterfactual (expected)
$\hat{y}^{cf}$	task label on counterfactual (model f)

Table 7: Summary of notation used in the paper

6.2 Satisfiability of the Counterfactual: Proposition 3

If the original label is N*, then there are* ***two*** *associated counterfactual labels,* E*, and* N*.*

For N, as $u \overset{(?)}{\Longrightarrow} x$ is truth-valueless under Open Universe Semantics (Russell, 2015)), the formula is either truth-valueless of every variables in x and u, or truth-valueless of some and false of others (Nolt, 2021). We start by considering the predicate relation $R(A, B) = \neg(A \Rightarrow B)$ which we rewrite as $A \wedge \neg B$, so that the two variables are separately testable (due to $\wedge$ relation).

For A, applying Assumption 1, we verify the satisfiability of $u \Rightarrow x^{\text{cf}}_{[A]}$ where the counterfactual hypothesis is $x^{\text{cf}}_{[A]} = x_{[A \backslash B]}$. We test the formula which excludes B but has A. The associated counterfactual result of $y^{\text{cf}}_{[A]} = \text{E}$.

Next we consider $\neg B$, with the counterfactual hypothesis $x^{\text{cf}}_{[\neg B]} = x_{[\neg B \backslash A]}$. Since the original propositional formula was truth-valueless, the same conditions apply for $u \overset{(?)}{\Longrightarrow} x^{\text{cf}}_{[\neg B]}$ which is still truth-valueless. The negation of a truth-valueless formula is still truth-valueless (Nolt, 2021)) hence $u \overset{(?)}{\Longrightarrow} x^{\text{cf}}_{[B]}$ has the counterfactual label $y^{\text{cf}} = \text{N}$.

6.3 Experiment Details

Dataset The dataset contains a reduced sample of the original 570k sentence pairs from SNLI where Do et al. (2020) apply various filtering methods to remove noise that occurred from combining e-SNLI and SNLI-VE. The training/val/test splits are 401718/14340/14741. Additional details about e-SNLI-VE including distribution are available at https://openaccess.thecvf.com/content/ICCV2021/supplemental/Kayser_E-ViL_A_Dataset_ICCV_2021_supplemental.pdf. Our use of this dataset is compatible with original access conditions and in research contexts.

Package Details For all models, we used Huggingface's `Transformers, v 4.12.2`. For Meteor Score calculation, we used `NLTK v 3.6.1` which lower cases tokens. For BertScore calculation, we used `roberta-large` and the metric implementation from Huggingface's `datasets v 1.15.1`. For training and validation loop, we used `PyTorch Lightning, v 1.5.2.`

Training Details Models are trained with Adam Optimizer (Kingma and Ba, 2014) with learning rate 10^{-5} and batch size 64. Models were trained using with Early stopping was used with `patience=5` and `threshold=0.0001`. We did not perform any hyperparameter search for this work, and used default values for training the models from the packages.

Computational Budget The largest model used was GPT-Neo 2.7B (2.7 Billion Parameters). The GPU hardware for the experiments used was `NVIDIA's Quadro RTX 6000`. The multimodal CLIP model takes about a day to fine-tune. To generate 14000 new counterfactual hypothesis using pretrained GPT-Neo 2.7B takes about 1-2 hours.

Prompt Selection Details We report results from a single run of random prompt selection, as hand-labelling multiple random prompt sets is manually intensive. A preliminary experiment comparing randomly sampled prompt sets, versus obtaining prompts using datapoints that are closest to the cluster centers from k-means clustering (where k=20) of the sentence unigram and bigram vectors, yielded nearly identical scores for Table 3. With a huge source dataset to sample from, we suspect that both types of prompt set selection might be close to random.

6.4 Related Work

Leakage Adjusted Simulatability For task inputs $X = \{x_i\}$, model outputs $\hat{Y} = \{\hat{y}_i\}$, and model explanations = $\hat{E} = \{\hat{e}_i\}$, Leakage Adjusted Simulatability (LAS) metric is computed as:

$$\mathrm{LAS}_0 = \frac{1}{n_0} \sum_{i:k_i=0} (\mathbb{1}[\hat{y}_i \mid x_i, \hat{e}_i] - \mathbb{1}[\hat{y}_i|x_i])$$
$$\mathrm{LAS}_1 = \frac{1}{n_0} \sum_{i:k_i=1} (\mathbb{1}[\hat{y}_i \mid x_i, \hat{e}_i] - \mathbb{1}[\hat{y}_i|x_i])$$
$$\mathrm{LAS} = \frac{1}{2}(\mathrm{LAS}_0 + \mathrm{LAS}_1)$$

where $k_i = \mathbb{1}[\hat{y}_i \mid \hat{e}_i]$ is a leakage indicator, and n_0 and n_1 are the number of examples in non-leaking and leaking groups respectively (Hase et al., 2020).

Following (Hase et al., 2020), we randomly drop out either $\hat{e}_i$ or x_i during training and can obtain the relevant conditional $\mathbb{1}[\hat{y}_i \mid x_i, \hat{e}_i]$, $\mathbb{1}[\hat{y}_i \mid x_i]$, or $\mathbb{1}[\hat{y}_i \mid \hat{e}_i]$ during inference time to compute LAS. We adopt the same set of final hyperparameters for dropout, 0.2 explanations only, 0.4 hypothesis and premise only, and 0.4 hypothesis, premise and explanations.

Label Rationale Association We adopt the "Robustness Equivalence measure" which tests whether both label and explanations are similarly or disimilarly robust to noise in the input. Following (Wiegreffe et al., 2020), we measure changes in label prediction as the number of predicted dev set labels that flip, i.e., change from their original prediction to something else, and measure changes in rationale quality using simulatability. Their approach aims to measure association between input perturbation and output explanation. They first add zero-mean Gaussian noise $\mathcal{N}(0, \sigma^2)$ to each input embedding at inference time, and measure the changes in label prediction by counting the number of predicted labels in the test set which flip. They then use the corresponding generated explanations, and get the task prediction change of a separate model that has been pretrained with explanations. The LRA is computed as:

$$F_i = \mathbb{1}[(\hat{y}_i|x_i) = (\hat{y}_i|x_i')] \quad (2)$$
$$Z_i = \mathbb{1}[\hat{y}_i \mid x_i, \hat{e}_i] - \mathbb{1}[\hat{y}_i|x_i] \quad (3)$$
$$\mathrm{LRA} = \frac{1}{n} \sum_{i}^{n} \mathbb{1}[F_i = Z_i] \quad (4)$$

Teacher-Student Paradigm Pruthi et al. (2020) introduce a student-teacher paradigm, where they target a notion of faithfulness being tied to the usefulness of explanations. They measure the extent to which explanations allow student models to simulate the teacher model on unseen examples for which explanations are unavailable. Student models incorporate explanations in training (but not prediction) procedures. The proposed framework for evaluating explanation quality suggests that explanations are effective if they help students learn about the teacher.

Erasure The vast majority of prior work that proposes models and evaluations of explanatory methods for faithfulness focus on removing tokens from the input text (DeYoung et al., 2019). This method directly intervenes in the input hypothesis by finding 'important' spans and is not applicable to free-text explanations. Our method has a similar spirit of doing direct interventions. However instead of removing tokens, we generate a counterfactual hypothesis which is grammatically close to the original hypothesis, but with propositions from the explanation.

	Tests task model	Reliance on Third Party	Resources Required
HS (Doshi-Velez and Kim, 2017)	No	Human verification	Human annotators
LAS (Hase et al., 2020)	No	Model verification	Training with explanations
TS (Pruthi et al., 2020)	No	Model verification	Training with explanations
LRA (Wiegreffe et al., 2020)	No	Model verification	Training with explanations
Faithfulness-through-Counterfactuals	Yes	Counterfactual hypothesis generator	Manual prompts

Table 8: Comparison of existing faithfulness metrics for free-text explanations. HS: Human Simulation, LAS: Leakage Adjusted Simulation, TS: Teacher-Student, LRA: Label Rationale Association. Compared to the other models, our proposed method 'Faithfulness-through-Counterfactuals' directly tests the task model without relying on third party Human or model verification. The reliance on third party is via counterfactual hypothesis generator, which we validate the efficacy in Subsection 3.2. Additionally, our method relies a one-off manual prompts generation which is cheaper than gathering explanation training data or relying on human annotators.

6.5 Annotation Details

We recruited annotators from the pool of Amazon Mechanical Turk workers who are located in the United States. In the counterfactual hypotheses rewriting task (Subsection 3.2), we paid between \$0.15 (for `E` and `C`) and \$0.20 (for `N`) per single rewrite, estimating that a sufficiently experienced worker can perform the task in under 45 seconds, thus yield a minimum of \$12 hourly rate. (We do not separately ask humans to extract logical spans.) In the NLI label collection task (Subsection 3.3), we paid \$0.10 per single NLI label and estimated that each task should take a sufficiently experienced worker at most 30 seconds, thus yielding a minimum of \$12 hourly rate.

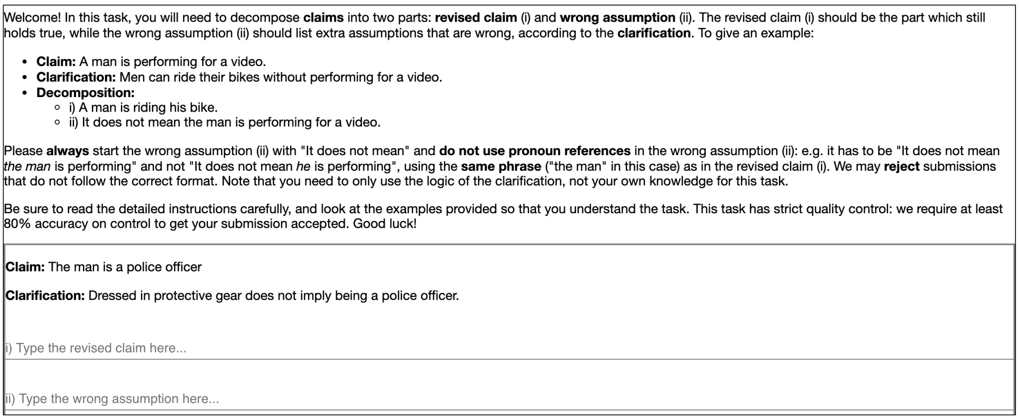

Figure 3: Interface shown to annotators in the counterfactual hypothesis rewriting task.

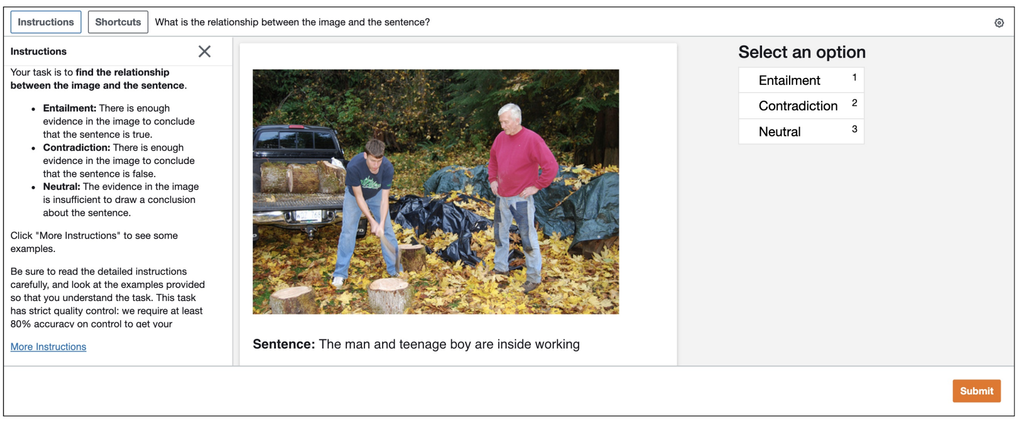

Figure 4: Interface shown to annotators in the NLI label collection task, mirroring the original SNLI protocol.

6.6 Inter-annotator Agreement

C ↦ E	E ↦ E	$N_{[A]}$ ↦ E	$N_{[B]}$ ↦ N
0.462	0.353	0.368	0.279

Table 9: Fleiss' kappa for agreement between three human annotators on the counterfactual label grouped by the original label (LHS of ↦ in header row).

C ↦ E	E ↦ E	$N_{[A]}$ ↦ E	$N_{[B]}$ ↦ N
0.534	0.867	0.756	0.790

Table 10: Accuracy of aggregated human label in predicting derived NLI label (RHS of ↦; Subsection 2.1).

6.7 Potential Limitations

We document and discuss some potential limitations of our proposed approach.

1. The suggested implementation (Subsection 2.4) relies on pre-trained models which are open-source,[20] but requires GPU memory of >10Gb memory to host the 2.7B parameter model and <10Gb to host the 1.3B parameter model.

2. Our metric is not directly differentiable as it involves discrete operations to generate the counterfactual hypothesis.

3. While most classification problems can be reformulated into ternary logic of `E, C, N`, it is more challenging to extend the notion of logical *satisfiability* for tasks that require open-ended text generation, and so the current work is limited to NLI task explanations.

6.8 Potential Risks

The proposed work is limited to investigating an evaluation methodology for faithfulness. Faithfulness as defined in this work is limited in scope to the NLI task, and is concerned with the decisions being made by the model. It has no influence on the actual classification task. As far as we know, there are **no known** potential malicious or unintended harmful effects and uses (e.g., disinformation, generating fake profiles, surveillance), environmental impact (e.g., training huge models), fairness considerations (e.g., deployment of technologies that could further disadvantage or exclude historically disadvantaged groups), privacy considerations (e.g., a paper on model/data stealing), and security considerations (e.g., adversarial attacks).

[20] https://huggingface.co/EleutherAI/gpt-neo-2.7B

6.9 Low Quality Explanations

Figure 5: Example of low quality explanation where the counterfactual hypothesis does not result in the expected logical label.

Hypothesis	Man riding a motorcycle.
Explanation	The man is on a lake with a monoboard, which would be impossible to ride a motorcycle on.
Counterfactual	Man is on a lake with a monoboard.
Old→New Label	Contradiction → Neutral

Figure 6: Example of low quality explanation where the counterfactual hypothesis does not result in the expected logical label.

Hypothesis	A dog is sitting on a porch.
Explanation	A dog cannot be running and sitting at the same time.
Counterfactual	A dog is running on a porch.
Old→New Label	Contradiction → Contradiction

Figure 7: Example of low quality explanation where the counterfactual hypothesis does not result in the expected logical label.

Hypothesis	Three ladies are gathered.
Explanation	If three ladies are standing in the same room, we can assume that they are gathered.
Counterfactual	Three ladies are standing in the same room.
Old→New Label	Entailment → Contradiction

6.10 Hypothesis Rewriting Validation

		Human Explanations $z^* \mapsto x^{\text{cf}}$				Generated Explanations $\hat{z} \mapsto x^{\text{cf}}$			
Extract	Transform	C	E	$\text{N}_{[A \setminus B]}$	$\text{N}_{[B \setminus A]}$	C	E	$\text{N}_{[A \setminus B]}$	$\text{N}_{[B \setminus A]}$
regex	regex	0.690	0.638	0.010	0.012	0.744	0.701	0.473	0.418
regex	GPT2-xl	0.696	0.690	0.716	0.748	0.813	0.817	0.795	0.727
GPT2-xl	regex	0.779	0.632	0.566	0.447	0.860	0.759	0.700	0.512
GPT2-xl	GPT2-xl	0.766	0.716	0.592	0.643	0.813	0.810	0.743	0.725
regex	Neo1.3B	0.684	0.675	0.734	0.723	0.798	0.834	**0.806**	0.702
Neo1.3B	regex	0.774	0.643	0.473	0.414	0.886	0.787	0.668	0.565
Neo1.3B	Neo1.3B	0.769	0.698	0.716	0.721	0.809	0.822	0.798	0.703
regex	Neo2.7B	0.690	0.709	0.710	**0.752**	0.840	0.860	**0.805**	**0.714**
Neo2.7B	regex	0.801	0.691	0.509	0.565	**0.893**	0.781	0.669	0.584
Neo2.7B	Neo2.7B	**0.822**	**0.782**	**0.743**	**0.754**	0.870	**0.881**	**0.807**	0.709

Table 11: METEOR scores for Hypothesis Revision by either using regex, GPT-Neo, or a combination of both. Bold is applied column-wise. We show the breakdown of revised hypothesis by class label (Contradiction C, Entailment E and Neutral N), and also whether the explanations are human or model generated. Note that there are two counterfactual hypothesis generated for N ($\text{N}_{[A \setminus B]}$ and $\text{N}_{[B \setminus A]}$) as described in Subsection 2.2: Proposition 3. As expected, the performance of Neo2.7B > Neo1.3B > regex for both z^* and $\hat{z}$ when the same 'model' is used for both extract and transform. Considering a combination of approaches (row-wise), for $\hat{z}$, the best combination of (regex + Neo1.3B) performs close to the best combination of (regex + Neo2.7B), which suggests that a smaller $\mathcal{H}^{\text{model}}$ may be sufficiently competitive.

6.11 Sensitivity Analysis

			FTC-$\mathcal{W}$				FTC-δ				FTC-$\mathcal{K}$			
x	u	y	gpt-gpt	gpt-regex	regex-gpt	regex-regex	gpt-gpt	gpt-regex	regex-gpt	regex-regex	gpt-gpt	gpt-regex	regex-gpt	regex-regex
✓	✓	✓	0.608	0.585	0.589	0.571	0.684	0.585	0.656	0.562	0.126	-0.112	0.032	-0.224
✓			0.539	0.534	0.536	0.394	0.591	0.522	0.584	0.281	-0.200	-0.299	-0.229	-1.407
	✓		0.580	0.421	0.557	0.423	0.650	0.340	0.618	0.323	-0.055	-1.261	-0.182	-1.229
		✓	0.553	0.445	0.551	0.447	0.621	0.396	0.613	0.375	-0.245	-1.111	-0.225	-1.080

Table 12: Raw scores for FTC metric variants by perturbing inputs to the explanation generator (sensitivity analysis). x, u, and y refers to hypothesis, premise, and label respectively. We vary different parts of the extract-transform pipeline (either with regex-regex, regex-gpt, gpt-regex, gpt-gpt), using GPT-Neo2.7B. We find that all columns have the first row as the largest number as expected, but FTC-$\mathcal{K}$ produces larger difference between the first row and other rows.

6.12 Prompt templates

Explanation	A	B
Contradiction		
Jail cells aren't pastel-colored or a classroom.	jail cells	pastel-colored or a classroom
The two men cannot both be at a construction site and also never have been at a construction site.	at a construction site	never have been at a construction site.
One cannot be happy and angry at the same time.	happy	angry
a dog is not a cat.	dog	cat
Star is not balloon	star	balloon
Entailment		
The guy on the phone is sitting at a desk.	guy on the phone	sitting at a desk
People is a generalization of many people.	people	many people
Jockeys are people, so jockeys riding horses are people riding horses.	jockeys	people
If people are sitting on blankets and the blankets are in the park, then the people are sitting at the park as well, because they cannot be in a separate location from the blankets they are sitting upon.	people sitting on blankets	people sitting on blankets in the park
If your pausing for a photo then they are having their photo taken	pausing for a photo	having photo taken
Neutral		
Looking on does not mean waiting for their turn and doesn't mean they are at a competition.	looking	waiting for their turn and they are at a competition
Just because a track athlete is carrying a large pole it does not mean they gets ready for their turn at the pole vault.	A track athlete is carrying a large pole	A track athlete gets ready for their turn at the pole vault.
There is no evidence that the women talking to the man outside the building are sad	The women talking to the man outside the building	the women are sad
Just because he has a mop, does not mean he has a broom.	he has a map	he has a broom
No way to know that is alone because he lost his friends in a war.	he is alone	he lost his friends in a war

Table 13: Example of prompts for each label, Contradiction, Entailment and Neutral provided to a few-shot priming model (GPT-Neo) for Extraction. We show the model 20 prompts in all experiments.

hyp_original	hyp_cf	A	B
Contradiction / Entailment			
They are waiting for parole in their jail cells.	They are waiting for parole in their pastel-colored classroom	jail cells	pastel-colored classroom
The two men have never been to a construction site.	The two men are at a construction site.	at a construction site	never been to a construction site
an angry man grills vegetables on a barbecue.	A happy man grills vegetables on a barbecue.	angry	happy
There is a cat running	There is a dog running	cat	dog
The young man is holding the balloon inside the large stone building.	The young man is holding the star inside the large stone building.	balloon	star
Neutral			
The skateboarders are waiting for their turn at a competition.	A: The skateboarders are looking.; B: The skateboarders are waiting for their turn.	looking	waiting for their turn
A track athlete gets ready for their turn at the pole vault.	A: A track athlete carrying a large pole.; B: A track athlete gets ready for their turn at the pole vault.	carrying a large pole	gets ready for their turn at the pole vault
Three sad women standing outside a building talking to a man.	A: Three talking to the man outside the building; B: Three sad women	the women talking to the man outside the building	the women are sad
There is an old man sitting alone because he lost his friends in a war.	A: There is an old man sitting alone; B: There is an old man who lost his friends in a war.	he is alone	he lost his friends in a war

Table 14: Example of prompts for each label, Contradiction, Entailment and Neutral provided to a few-shot priming model (GPT-Neo) for Transformation. Contradiction and Entailment uses the same set of prompts. We show the model 20 prompts in all experiments.

6.13 Regex templates

R1	a/an, a type of, a way of saying, the same as, a rephrasing of, a/another form of, synonymous with
R2	is, are
R3	synonyms
R4	then, so, must be, has to be, have to be
P1	A is R1 B
P2	A implies B
P3	A and B are R3
P4	A and B R2 the same thing
P5	if A then B
P6	A R4 B
P7	A R2 B

Table 15: Regex for extracting logical variables "A" and "B" from Entailment (E).

R1	cant, cannot, can't, can not
R2	at the same time,simultaneously, at once
R3	is, are
R4	not the same as, not, the opposite of, different than
R5	he, she, they
R6	a, an
P1	A R3 not R6 B
P2	R1 be A and B R2
P3	A R1 be B
P4	A R3 R4 B
P5	R3 either A or B
P6	A R3 not B
P7	A R3 different than B
P8	R1 be A if R3 B
P9	R1 be A if R5 is B
P13	R1 A if B
P10	A and B R3 different
P11	A would not be able to B
P12	A R1 be B

Table 16: Regex for extracting logical variables "A" and "B" from Contradiction (C).

R1	is, are
R2	not all, not every
R3	mean, necessarily mean, make, necessarily make, imply, indicate
R4	does not, doesnt, doesn't
R5	did not, didn't, didnt
P1	R2 A R1 B
P2	there is more A than B
P3	there is more A than B
P4	just because A R4 R3 B
P5	A R1 not necessarily B
P6	A R4 have to be B
P7	A R4 necessarily B
P8	A R4 R3 B
P9	can A without B
P10	could be A not just B
P12	we R5 know A to B
P11	we R5 know if A or B
P12	we can't tell if A is B
P13	if A then B
P14	this R4 imply A or B
P15	A and B R1 two different
P16	A and B R1 different
P17	not everyone A will B
P18	A may not be B
P19	it cannot be assumed that A is B
P20	some A or B
P21	A might not be B
P22	there is not evidence A or B
P23	R4 have to be A to B
P24	no way to know A or B

Table 17: Regex for extracting logical variables "A" and "B" from Neutral (N).