

# KNOW How to Make Up Your Mind! Adversarially Detecting and Alleviating Inconsistencies in Natural Language Explanations

Myeongjun Jang<sup>1</sup> Bodhisattwa Prasad Majumder<sup>3</sup> Julian McAuley<sup>3</sup>  
Thomas Lukasiewicz<sup>2,1</sup> Oana-Maria Camburu<sup>4</sup>

<sup>1</sup>University of Oxford, UK <sup>2</sup>Vienna University of Technology, Austria

<sup>3</sup>University of California San Diego <sup>4</sup>University College London, UK

myeongjun.jang@cs.ox.ac.uk

## Abstract

While recent works have been considerably improving the quality of the natural language explanations (NLEs) generated by a model to justify its predictions, there is very limited research in detecting and alleviating inconsistencies among generated NLEs. In this work, we leverage external knowledge bases to significantly improve on an existing adversarial attack for detecting inconsistent NLEs. We apply our attack to high-performing NLE models and show that models with higher NLE quality do not necessarily generate fewer inconsistencies. Moreover, we propose an off-the-shelf mitigation method to alleviate inconsistencies by grounding the model into external background knowledge. Our method decreases the inconsistencies of previous high-performing NLE models as detected by our attack.

## 1 Introduction

The accurate yet black-box nature of deep neural networks has accelerated studies on explainable AI. The advent of human-written natural language explanations (NLEs) datasets (Wiegrefe and Marasovic, 2021) has paved the way for the development of models that provide NLEs for their predictions. However, by introducing an adversarial attack, which we hereafter refer to as eIA (explanation Inconsistency Attack), Camburu et al. (2020) found that an early NLE model (Camburu et al., 2018) was prone to generate inconsistent NLEs (In-NLEs). More precisely, two *logically contradictory* NLEs generated by a model for two instances that have the same context are considered to form an *inconsistency*. For example, assume a self-driving car stops in a given traffic environment (the context). If the passenger asks the car Q1: “Why did you stop?”, and it provides NLE1: “Because the traffic light is red.”, and, for the same context, if the passenger instead asks Q2: “Why did you decide to stop here?” and the car provides

NLE2: “Because the traffic light is green”, then NLE1 and NLE2 form an inconsistency.

A model that generates In-NLEs is undesirable, as it *either has a faulty decision-making process* (e.g., the traffic light was green, so the car should not have stopped), or it *generates NLEs that are not faithfully describing its decision-making process* (e.g., the car stopped for a red traffic light, but states that it was green) (Camburu et al., 2020). While recent high-performing NLE models have largely improved in terms of the quality (plausibility) of the generated NLEs, to our knowledge, these models have not been tested against generating inconsistent NLEs.

In this work, we first propose a fast, efficient, and task-generalizable adversarial attack that utilizes external knowledge bases. Through experiments on two datasets and four models, we verify the increased efficiency of our approach over the eIA attack, the only inconsistency attack for NLE models, to our knowledge. We also show that the high-performing NLE models are still prone to generating significantly many In-NLEs and, surprisingly, that a higher NLE quality does not necessarily imply fewer inconsistencies. Second, we propose a simple yet efficient off-the-shelf method for alleviating inconsistencies that grounds any NLE model into background knowledge, leading to fewer inconsistencies. The code for this paper is available at <https://github.com/MJ-Jang/eKnowIA>.

## 2 Inconsistency Attack

We propose eKnowIA (explanations **K**nowledge-grounded **I**nconsistency **A**ttack), which detects more In-NLEs in a faster and more general manner than eIA.

### 2.1 Original eIA Attack

**Setting.** Given an instance  $x$ , Camburu et al. (2020) divide it into: the *context* part  $x_c$  that remains fixed, and the *variable* part  $x_v$  that is

changed during the attack. For example,  $x_c$  and  $x_v$  would be a *premise* and a *hypothesis*, respectively, for natural language inference (NLI — detailed below). Let  $e_m(x)$  denote the NLE generated by a model  $m$  for the input  $x = (x_c, \hat{x}_v)$ . The objective is to find  $\hat{x}_v$  such that  $e_m(x)$  and  $e_m((x_c, \hat{x}_v))$  are logically contradictory (see examples in Table 10).

**Steps.** The eIA attack has the following steps:

1. Train a neural model to act as a reverse explainer, called REVEXPL, that takes  $x_c$  and  $e_m(x)$  as input and generates  $x_v$ , i.e.,  $\text{REVEXPL}(x_c; e_m(x)) = x_v$ .
2. For each generated NLE  $e_m(x)$ :
  - (a) Automatically create a set of statements  $\mathcal{I}_e$  that are inconsistent with  $e_m(x)$ .
  - (b) For each  $\hat{e} \in \mathcal{I}_e$ , generate a variable part  $\hat{x}_v = \text{REVEXPL}(x_c; \hat{e})$ .
  - (c) Query  $m$  on  $\hat{x} = (x_c, \hat{x}_v)$  to get  $e_m(\hat{x})$ .
  - (d) Check whether  $e_m(\hat{x})$  is indeed inconsistent with  $e_m(x)$  by checking whether  $e_m(\hat{x})$  is included in  $\mathcal{I}_e$ .

**Creating  $\mathcal{I}_e$ .** Camburu et al. (2020) used simple elimination of negation (removing “not” or “n’t”) and a task-specific template-based approach for this step. For the template-based approach, they manually create a set of label-specific templates for NLEs such that introducing the instance-specific terms of an NLE from one template into any template from another label creates an inconsistency. They illustrate this process only on the e-SNLI dataset (Camburu et al., 2018), leaving room to question how easily it generalizes to other datasets. e-SNLI contains NLEs for the SNLI dataset (Bowman et al., 2015), where the NLI task consists in identifying whether a *premise* and a *hypothesis* are in a relation of *entailment* (if the premise entails the hypothesis), *contradiction* (if the hypothesis contradicts the premise), or *neutral* (if neither entailment nor contradiction hold). Examples of their templates are: “<X> is <Y>” (for *entailment*) and “<X> cannot be <Y>” (for *contradiction*). Based on the templates, for a  $e_m(x)$  of “A dog is an animal.”, an inconsistent statement of “A dog cannot be an animal” is obtained (<X> = “A dog”, <Y> = “an animal”). They manually identified an average of 10 templates per label.

## 2.2 Our eKnowIA Attack

The template-based approach in eIA has two major drawbacks: (1) requires substantial human effort to find an exhaustive set of templates for *each*

dataset, (2) many different ways of obtaining inconsistencies (e.g., using antonyms) are not taken into account. Moreover, even their negation rule can also be improved. To alleviate these drawbacks, we adopt three rules.

**Negation.** We remove and *add* negation tokens to negated and non-negated sentences, respectively. To avoid grammatical errors, we add one negation per sentence only if the sentence belongs to one of the following two templates:

- <A> is <B>, <A> are <B> (add “not”),
- <A> has <B>, <A> have <B> (add “does/do not” only if <B> is a noun).

### Antonym replacement for adjectives/adverbs.

We replace adjectives/adverbs with their antonyms from ConceptNet (Speer et al., 2017) (using the NLTK POS tagger). Only one adjective or adverb at a time is replaced for each NLE, to avoid deteriorating the contradictory meaning. Employing other abundant thesauruses could improve our approach, which we leave as future work.

**Unrelated noun replacement.** We replace a noun with an unrelated one, e.g., “human” with “plant”. This is only applied to the noun that is the last word of the sentence, to reduce the possibility of false inconsistencies as the part-of-speech (POS) tagger occasionally made incorrect predictions for words in the middle of a sentence. To get unrelated nouns, we use the *DistinctFrom* and *Antonym* relations in ConceptNet. However, we noticed that ConceptNet contains noisy triplets where the subject and object are not antonyms, such as “man” and “people” for “person”<sup>1</sup>. To avoid these, we created a list (see Table 8 in the appendix) of triplets from ConceptNet to be ignored, by manually investigating a random subset of 3000 detected inconsistencies. While this involved human effort, we highlight that this is due to the nature of ConceptNet and other knowledge bases with more accurate instances may be used instead. However, since we only found eight noisy triplets, we decided to keep ConceptNet, which otherwise worked well for our datasets. Finally, we also noticed that our rules may not lead to In-NLEs if both the context and variable part contain negations. Examples are in Table 9 in the appendix. We filter out such pairs.

## 2.3 Experiments

**Datasets.** We consider two tasks: NLI (with the e-SNLI dataset described in Sec. 2.1) and com-

<sup>1</sup>A pair of words with *opposite* meanings (from Wikipedia).

Model	e-SNLI				Cos-E			
	Acc.	$\mathcal{S}_r$	$\mathcal{H}_r$	e-ViL	Acc.	$\mathcal{S}_r$	$\mathcal{H}_r$	e-ViL
NILE	90.7	3.13	2.27	0.80	-	-	-	-
KnowNILE	<b>90.9</b>	<b>2.42†</b>	<b>1.99†</b>	<b>0.82</b>	-	-	-	-
CAGE	-	-	-	-	61.4	0.42	0.06	0.43
KnowCAGE	-	-	-	-	<b>62.6</b>	<b>0.11†</b>	<b>0.01†</b>	<b>0.44</b>
WT5-base	90.6	12.88	1.70	0.76	65.1	0.95	0.12	0.55
KnowWT5-base	<b>90.9</b>	<b>11.45</b>	<b>1.19†</b>	<b>0.80†</b>	<b>65.5</b>	<b>0.84†</b>	<b>0.09†</b>	<b>0.56</b>

Table 1: Results of our eKnowIA attack and our method for mitigating In-NLEs. The best results for each pair of (model, Know-model) are in bold;  $\mathcal{S}_r$  and  $\mathcal{H}_r$  are given in %; † indicates that Know-models showed statistically significant difference with  $p$ -value < 0.05 (†) using the t-test.

Dataset	Method	Time	$\mathcal{S}_r$	$\mathcal{H}_r$
e-SNLI	eIA	10 days	2.19	384/24M
	eKnowIA	<b>40 min</b>	<b>12.88</b>	<b>1,494/88K</b>
Cos-E	eIA	2.5 days	0.32	5/5M
	eKnowIA	<b>5 min</b>	<b>0.95</b>	<b>13/11K</b>

Table 2: Comparison between eIA and eKnowIA on WT5-base. The best results are in bold;  $\mathcal{S}_r$  is given in %;  $\mathcal{H}_r$  values are in fractions to emphasise the high denominators of the eIA.

monsense question answering (CQA). The Cos-E 1.0 dataset (Rajani et al., 2019) contains CQA instances formed of a *question*, three *answer candidates*, and an NLE for the correct answer. The objective of the Cos-E (Rajani et al., 2019) dataset is to select an *answer* among the three candidates given a *question* and to generate an NLE to support the answer. Following Camburu et al. (2020), we set the premise as context and the hypothesis as the variable part for e-SNLI. For Cos-E, to avoid omitting the correct answer, we set the question and the correct answer as the context, and the remaining two answer candidates as the variable part. Just like eIA, our attack is solely intended for detecting In-NLEs and not as a label attack (which may or may not happen).

**Evaluation metrics.** Let  $\mathcal{I}_e$  be generated at Step 2a for each instance in a test set  $\mathcal{D}_{test}$ , and let  $\mathcal{I}_s \subseteq \mathcal{I}_e$  be the set of detected In-NLEs (after Step 2d). For each instance, our attack can identify multiple inconsistencies (via multiple variable parts). We, therefore, use two evaluation metrics: hit-rate ( $\mathcal{H}_r$ ) and success-rate ( $\mathcal{S}_r$ ):

$$\mathcal{S}_r = N_c / |\mathcal{D}_{test}| \text{ and } \mathcal{H}_r = |\mathcal{I}_s| / |\mathcal{I}_e|,$$

where  $N_c$  is the number of unique instances for which the attack identified at least one inconsistency. Intuitively,  $\mathcal{S}_r$  denotes the ratio of the test instances where the attack is successful, while  $\mathcal{H}_r$  denotes the ratio of detected In-NLEs to that of the proposed In-NLEs.

**Models.** We consider the following high-performing NLE models, with their implemen-

tation detailed in Appendix A.1: NILE (Kumar and Talukdar, 2020) for NLI, CAGE (Rajani et al., 2019) for CQA, and WT5-base (220M parameters) (Narang et al., 2020) for both tasks. WT5 models with more parameters (e.g., WT5-11B) would require considerably more computing while providing relatively small gains in NLE quality (32.4 for WT5-base vs. 33.7 for WT5-11B (Narang et al., 2020)). Therefore, they are not considered here due to limited computing resources. Implementation details are given in Appendix A.1.

## 2.4 Results

**eKnowIA vs. eIA.** We compare eKnowIA with eIA only on the WT5-base model, since eIA requires a prohibiting amount of time. As in Camburu et al. (2020), we manually verified the naturalness of adversarial hypotheses on 50 random samples for each model. Sentences that go against common sense are considered unnatural. Minor grammatical errors and typos are ignored. We observe that 81.5% of the adversarial hypotheses were natural, on average, for each model. Details are in Appendix A.4. The results are summarized in Table 2. The e-SNLI results are adjusted to reflect the proportion of natural adversarial hypotheses by multiplying the number of detected pairs of In-NLEs for each model with the estimated naturalness ratio. For Cos-E, an unnatural variable part would consist of stop words or a repetition of another answer candidate. We automatically found 2 out of 22 examples to be unnatural, which were removed. We observe that eIA generates a tremendous amount of inconsistent candidates ( $\mathcal{I}_e$ ), e.g., 24M for e-SNLI, thus being extremely slow (e.g., 10 days vs. 40 min for eKnowIA), while also obtaining lower  $\mathcal{S}_r$  and  $\mathcal{H}_r$  than eKnowIA (e.g., 2.19% vs. 12.88%  $\mathcal{S}_r$ ).

**eKnowIA on NLE models.** The results of eKnowIA applied to NILE, CAGE, and WT5 are in

PREMISE: A man is riding his dirt bike through the air in the desert.	
HYPOTHESIS: A man is on a motorbike PREDICTED LABEL: entailment EXPLANATION: A dirt bike is a motorbike.	HYPOTHESIS: The man is riding a motorbike. PREDICTED LABEL: contradiction EXPLANATION: A dirt bike is not a motorbike.
QUESTION: John knew that the sun produced a massive amount of energy in two forms. If you were on the surface of the sun, what would kill you first?	
CHOICES: heat, light, life on earth PREDICTED LABEL: heat EXPLANATION: the sun produces heat and light.	CHOICES: heat, light, darkness PREDICTED LABEL: heat EXPLANATION: the sun produces heat and darkness.

Table 3: Examples of inconsistent NLEs detected by eKnowIA for WT5 on e-SNLI and CAGE on Cos-E. The first column shows the original variable part, and the second column shows the adversarial one.

PREMISE: A man is riding his dirt bike through the air in the desert.	
HYPOTHESIS: A man is on a motorbike PREDICTED LABEL: entailment EXTRACTED KNOWLEDGE: {dirt bike, IsA, motorcycle}, {desert, MannerOf, leave}, {air, HasA, oxygen} EXPLANATION: A dirt bike is a motorbike.	HYPOTHESIS: The man is riding a motorbike. PREDICTED LABEL: entailment EXTRACTED KNOWLEDGE: {dirt bike, IsA, motorcycle}, {desert, MannerOf, leave}, {air, HasA, oxygen} EXPLANATION: A dirt bike is a motorbike.
QUESTION: John knew that the sun produced a massive amount of energy in two forms. If you were on the surface of the sun, what would kill you first?	
CHOICES: heat, light, life on earth PREDICTED LABEL: heat EXTRACTED KNOWLEDGE: {light, IsA, energy}, {heat, IsA, energy} EXPLANATION: light and heat are two forms of energy.	CHOICES: heat, light, darkness PREDICTED LABEL: heat EXTRACTED KNOWLEDGE: {light, Antonym, dark}, {heat, IsA, energy} EXPLANATION: the sun produces heat and light.

Table 4: Examples of successfully defended instances by KnowWT5 on e-SNLI and KnowCAGE on Cos-E. This table should be read together with Table 3 to appreciate the defence.

the upper lines of each block in Table 1. All models are vulnerable to the inconsistency attack. Also, a better NLE quality may not necessarily guarantee fewer inconsistencies. For example, WT5-base has a better NLE quality than CAGE on Cos-E (0.55 vs. 0.43 e-ViL score; see below), but eKnowIA detected more inconsistencies for WT5-base than for CAGE (0.95 vs. 0.42 success rate). Examples of generated In-NLEs are in Table 3. More examples are in Tables 10–12 in Appendix A.7. We observe that the In-NLEs usually contradict common sense, which is aligned with previous studies showing that language models, used as pre-trained components in the NLE models, often suffer from factual incorrectness (Mielke et al., 2020; Zhang et al., 2021).

### 3 Our KNOW Method for Alleviating Inconsistencies

Our approach for alleviating inconsistencies in NLE models consists of two steps: (1) extraction of knowledge related to the input and (2) knowledge injection.

**Extracting related knowledge.** We leverage a knowledge extraction heuristic proposed by Xu et al. (2021) as follows:

1. Extract entities from an input’s context part.
2. Find all knowledge triplets that contain the

entities.

3. For each entity, calculate a weight  $s_j$  for each extracted triplet as:

$$s_j = w_j \times N / N_{r_j} \text{ and } N = \sum_{j=1}^K N_{r_j},$$

where  $w_j$  is the weight of the  $j$ -th triplet pre-defined by the knowledge base (e.g., Concept-Net),  $N_{r_j}$  is the number of extracted triplets of the relation  $r_j$  for the given instance, and  $K$  is the total number of triplets containing the entity for the given instance.

4. For each entity, extract the triplet with the highest score.

**Grounding with the extracted knowledge.** After extracting the triplet with the highest weight per entity in an instance, we transform each of them into natural language and concatenate them to the instance. We use “Context:” as a separator between the input and the triplets. We leverage the templates that transform a relation into free-text (e.g., *IsA* to “is a”) from Petroni et al. (2019).

#### 3.1 Experiments

We apply our KNOW approach to NILE, CAGE, and WT5-base, and name them KnowNILE, KnowCAGE, and KnowWT5-base, respectively.

**Inconsistencies.** The results in Table 1 show that grounding in commonsense knowledge diminishes the number of In-NLEs for all models and tasks. The KNOW models defended against 58% of the examples attacked by eKnowIA. Also, we observed that, among the inconsistent examples of KNOW models, 20% of them on average were newly introduced instances. Examples that failed to be defended, as well as newly introduced In-NLEs are provided in Tables 15-16 in Appendix A.7. Successfully defended examples are provided in Table 4. More successfully defended examples, non-defended examples, and newly attacked examples can be found in Tables 13-14 in Appendix A.7.

First, we highlight that a successfully defended example means that our eKnowIE attack did not find an adversarial instance together with which the KNOW model would form a pair of In-NLEs, while our attack did find at least one such adversarial instance for the original model. Second, we notice that even when the selected knowledge might not be the exact knowledge needed to label an instance correctly, the model can still benefit from this additional knowledge. For example, in the first sample in Table 14 in Appendix A.7, the most proper knowledge triplet would be {dog, DistinctFrom, bird}. However, despite the indirect knowledge given,<sup>2</sup> i.e., {dog, DistinctFrom, cat}, the model is able to defend the In-NLE by inferring that dogs are different from other animals. To examine whether the improved consistency of the KNOW models stems from *knowledge leakage* (using the same knowledge triplets in the mitigation method as in the attack), we calculate the overlap of triplets. On the e-SNLI dataset, we find that only 0.3% of knowledge triplets are reused for the attack on the KNOW models, and no overlap was found for the Cos-E dataset. This indicates that the leakage is not significant.

**NLE quality.** To evaluate the quality of generated NLEs, we conducted a human evaluation using Amazon MTurk, as automatic evaluation metrics only weakly reflect human judgements (Kayser et al., 2021). We follow the setup from Kayser et al. (2021): we asked annotators (three per instance) to judge whether the generated NLEs justify the answer with four options: {no, weak no, weak yes, yes} and calculated the e-ViL score by mapping them to {0, 1/3, 2/3, 1}, respectively. Details of the human evaluation are in Appendix A.5. In Ta-

ble 1, the KNOW models show similar NLE quality to their original counterparts, suggesting that our KNOW method preserves NLE quality while decreasing inconsistencies. Similar results are observed on the automatic evaluation of NLEs (see Appendix A.6).

## 4 Related Work

A growing number of works focus on building NLE models in different areas such as natural language inference (Camburu et al., 2018), question answering (Narang et al., 2020), visual-textual reasoning (Hendricks et al., 2018; Kayser et al., 2021; Majumder et al., 2022), medical imaging (Kayser et al., 2022), self-driving cars (Kim et al., 2018), and offensiveness classification (Sap et al., 2019). Most commonly, the performance of these models is assessed only in terms of how plausible the reasons provided by their NLEs are. To our knowledge, Camburu et al. (2020) is the only work to investigate inconsistencies in NLEs. We improve their adversarial attack as well as bring an approach to alleviate inconsistencies. Works have also been conducted to analyse and make dialogue models generate responses consistent with the dialogue history (Zhang et al., 2018; Welleck et al., 2019; Li et al., 2020). However, these works are difficult to be applied to NLE models, in part because they require specific auxiliary datasets, such as pairs of inconsistent sentences. Other works investigated the logical consistency of a model’s predictions (Elazar et al., 2021; Mitchell et al., 2022; Kumar and Joshi, 2022; Lin and Ng, 2022), but would not have straightforward extensions for investigating NLEs inconsistencies. Besides consistency, NLEs can also be assessed for their faithfulness w.r.t. the decision-making process of the model that they aim to explain (Wiegrefe et al., 2021; Atanasova et al., 2023).

## 5 Summary and Outlook

We proposed the eKnowIA attack, which is more generalizable, successful, and faster than the previous eIA attack in detecting In-NLEs. Our experiments show that current NLE models generate a significant number of In-NLEs, and that higher NLE quality does not necessarily imply fewer inconsistencies. We also introduced a simple but efficient method that grounds a model into relevant knowledge, decreasing the number of In-NLEs. Our work paves the way for further work on detecting and alleviating inconsistencies in NLE models.

<sup>2</sup>ConceptNet does not contain {dog, DistinctFrom, bird}.

## Limitations

Our eKnowIA attack contains logical rules designed specifically for the English language. While these rules may apply or be adapted to other languages with simple morphology, there could be languages in which completely new rules may be needed. Both our attack and the KNOW method rely on knowledge bases, which may sometimes be noisy. We employed manual efforts to eliminate (a small number of) noisy triples from ConceptNet. Our attack also relies on a manual annotation to ensure that the adversarial inputs are natural (estimated to be the case 81.5% of the time). Finally, we were not able to test our methods on instances with long text, as we are not aware of datasets with NLEs for long text inputs or long NLEs.

## Acknowledgements

This work was partially supported by the Alan Turing Institute under the EPSRC grant EP/N510129/1, by the AXA Research Fund, and by the EU TAILOR grant 952215. Oana-Maria Camburu was supported by a Leverhulme Early Career Fellowship. We also acknowledge the use of Oxford’s ARC facility, of the EPSRC-funded Tier 2 facility JADE II (EP/ T022205/1), and of GPU computing support by Scan Computers International Ltd.

## References

- Pepa Atanasova, Oana-Maria Camburu, Christina Lioma, Thomas Lukasiewicz, Jakob Grue Simonsen, and Isabelle Augenstein. 2023. Faithfulness Tests for Natural Language Explanations. In *ACL*.
- Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*.
- Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In *EMNLP*.
- Oana-Maria Camburu, Tim Rocktäschel, Thomas Lukasiewicz, and Phil Blunsom. 2018. e-SNLI: Natural language inference with natural language explanations. In *NeurIPS*, volume 31.
- Oana-Maria Camburu, Brendan Shillingford, Pasquale Minervini, Thomas Lukasiewicz, and Phil Blunsom. 2020. Make up your mind! Adversarial generation of inconsistent natural language explanations. In *ACL*.
- Yanai Elazar, Nora Kassner, Shauli Ravfogel, Abhishava Ravichander, Eduard Hovy, Hinrich Schütze, and Yoav Goldberg. 2021. Measuring and improving consistency in pretrained language models. *arXiv preprint arXiv:2102.01017*.
- Lisa Anne Hendricks, Ronghang Hu, Trevor Darrell, and Zeynep Akata. 2018. Grounding visual explanations. In *ECCV*.
- 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. In *ICCV*.
- Maxime Kayser, Cornelius Emde, Oana-Maria Camburu, Guy Parsons, Bartłomiej Papież, and Thomas Lukasiewicz. 2022. Explaining chest x-ray pathologies in natural language. In *MICCAI*, pages 701–713.
- Jinkyu Kim, Anna Rohrbach, Trevor Darrell, John Canny, and Zeynep Akata. 2018. Textual explanations for self-driving vehicles. In *ECCV*.
- Ashutosh Kumar and Aditya Joshi. 2022. Striking a balance: Alleviating inconsistency in pre-trained models for symmetric classification tasks. In *Findings of ACL*, pages 1887–1895.
- Sawan Kumar and Partha Talukdar. 2020. NILE: Natural language inference with faithful natural language explanations. In *ACL*.
- Margaret Li, Stephen Roller, Ilya Kulikov, Sean Welleck, Y-Lan Boureau, Kyunghyun Cho, and Jason Weston. 2020. Don’t say that! Making inconsistent dialogue unlikely with unlikelihood training. In *ACL*.
- Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*.
- Ruixi Lin and Hwee Tou Ng. 2022. Does BERT know that the IS-a relation is transitive? In *ACL*, pages 94–99.
- Ilya Loshchilov and Frank Hutter. 2018. Decoupled weight decay regularization. In *ICLR*.
- Bodhisattwa Prasad Majumder, Oana-Maria Camburu, Thomas Lukasiewicz, and Julian McAuley. 2022. Rationale-inspired natural language explanations with commonsense. In *ICML*.
- Ana Marasović, Chandra Bhagavatula, Jae sung Park, Ronan Le Bras, Noah A. Smith, and Yejin Choi. 2020. Natural language rationales with full-stack visual reasoning: From pixels to semantic frames to commonsense graphs. In *Findings of EMNLP*.
- Ana Marasović, Iz Beltagy, Doug Downey, and Matthew E. Peters. 2022. Few-shot self-rationalization with natural language prompts. In *Findings of NAACL*.

- Sabrina J. Mielke, Arthur Szlam, Y-Lan Boureau, and Emily Dinan. 2020. Linguistic calibration through metacognition: Aligning dialogue agent responses with expected correctness. *arXiv preprint arXiv:2012.14983*.
- Eric Mitchell, Joseph J. Noh, Siyan Li, William S. Armstrong, Ananth Agarwal, Patrick Liu, Chelsea Finn, and Christopher D. Manning. 2022. Enhancing self-consistency and performance of pre-trained language models through natural language inference. *arXiv preprint arXiv:2211.11875*.
- Sharan Narang, Colin Raffel, Katherine Lee, Adam Roberts, Noah Fiedel, and Karishma Malkan. 2020. WT5?! Training text-to-text models to explain their predictions. *arXiv preprint arXiv:2004.14546*.
- Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: A method for automatic evaluation of machine translation. In *ACL*.
- Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. 2019. Language models as knowledge bases? In *EMNLP-IJCNLP*.
- Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *JMLR*, 21.
- Nazneen Fatema Rajani, Bryan McCann, Caiming Xiong, and Richard Socher. 2019. Explain yourself! Leveraging language models for commonsense reasoning. In *ACL*.
- Steven V. Rouse. 2019. Reliability of MTurk data from masters and workers. *Journal of Individual Differences*.
- Maarten Sap, Saadia Gabriel, Lianhui Qin, Dan Jurafsky, Noah A. Smith, and Yejin Choi. 2019. Social bias frames: Reasoning about social and power implications of language. *arXiv preprint arXiv:1911.03891*.
- Robyn Speer, Joshua Chin, and Catherine Havasi. 2017. ConceptNet 5.5: An open multilingual graph of general knowledge. In *AAAI*.
- Sean Welleck, Jason Weston, Arthur Szlam, and Kyunghyun Cho. 2019. Dialogue natural language inference. In *ACL*.
- Sarah Wiegrefe and Ana Marasovic. 2021. Teach me to explain: A review of datasets for explainable NLP. In *NeurIPS*, volume 35.
- Sarah Wiegrefe, Ana Marasovic, and Noah A. Smith. 2021. Measuring association between labels and free-text rationales. In *EMNLP*.
- Yichong Xu, Chenguang Zhu, Ruochen Xu, Yang Liu, Michael Zeng, and Xuedong Huang. 2021. Fusing context into knowledge graph for commonsense question answering. In *Findings of ACL*.
- Yordan Yordanov, Vid Kocijan, Thomas Lukasiewicz, and Oana-Maria Camburu. 2022. Few-Shot Out-of-Domain Transfer of Natural Language Explanations. In *Findings of EMNLP*.
- Saizheng Zhang, Emily Dinan, Jack Urbanek, Arthur Szlam, Douwe Kiela, and Jason Weston. 2018. Personalizing dialogue agents: I have a dog, do you have pets too? In *ACL*.
- Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. BERTScore: Evaluating text generation with BERT. In *ICLR*.
- Weifeng Zhang, Jing Yu, Wenhong Zhao, and Chuan Ran. 2021. DMRFNet: Deep multimodal reasoning and fusion for visual question answering and explanation generation. *Information Fusion*, 72:70–79.

## A Appendix

### A.1 Implementation Details

We implemented the WT5-base model based on the HuggingFace transformers package<sup>3</sup> and replicated performance close to the reported results (see Section A.3). For the other models, we used the implementations provided by the respective authors. A single Titan X GPU was used.

### A.2 Training REVEXPL

We adopted T5-base (Raffel et al., 2020) for training the reverse explainer (REVEXPL). We trained the model for 30 epochs with a batch size of 8. For efficient training, early stopping was applied if the validation loss increases for 10 consecutive logging steps, which were set to 30,000 iterations. The dropout ratio was set to 0.1. We used the AdamW optimiser (Loshchilov and Hutter, 2018) with learning rate  $1e^{-4}$  and epsilon  $1e^{-8}$ . We also used gradient clipping to a maximum norm of 1.0 and a linear learning rate schedule decaying from  $5e^{-5}$ .

For Cos-E, we used 10% of the training data as the validation set, and the original validation set as the test set.

### A.3 WT5-base Performance Replication

This section describes the performance of our trained WT5-base model. We report the accuracy for measuring the performance on the natural language inference (NLI) and CQA tasks. To automatically evaluate the quality of generated NLEs, we use the BLEU score (Papineni et al., 2002), ROUGE (Lin, 2004), Meteor (Banerjee and Lavie, 2005), and the BERT score (Zhang et al., 2020), which are widely used automatic evaluation metrics. The results are summarised in Table 5. In terms of accuracy and BLEU score, our replication performs better than originally reported for Cos-E, but produced slightly lower results for e-SNLI.

	Acc.	BLEU	R-1	R-2	R-L	Meteor	BERT-S	
e-SNLI	ours	90.6	28.4	45.8	22.5	40.6	33.7	89.8
	reported	90.9	32.4	-	-	-	-	-
Cos-E	ours	65.3	7.3	25.0	8.3	21.6	20.2	86.3
	reported	59.4	4.6	-	-	-	-	-

Table 5: Performance of our implementation of WT5-base on e-SNLI and Cos-E. The notations R-1, R-2, R-L, and BERT-S denote ROUGE-1, ROUGE-2, ROUGE-L score, and BERT-Score, respectively.

<sup>3</sup><https://github.com/huggingface/transformers>

### A.4 Naturalness Evaluation of the Generated Variable Parts

It could be unfair to consider that a model generates inconsistent NLEs if the adversarial variable parts are unnatural. Hence, we manually evaluated 50 random samples of generated adversarial variable parts for each model (or all samples when there were less than 50 pairs of inconsistencies found).

On e-SNLI, we observe that, on average, 81.5% ( $\pm 1.91$ ) of the reverse variable parts were natural instances, i.e., semantically valid and not contradicting commonsense. The specific figures for each e-SNLI model were 80%, 80%, 84%, and 82% for KnowNILE, NILE, WT5, and KnowWT5, respectively. We adapted the results in Table 1 to reflect the number of inconsistencies caused only by natural variable parts.

For the Cos-E dataset, we considered that the variable parts (the two incorrect answer choices) are unnatural if (1) the answer choices are stopwords of the NLTK package or (2) the correct answer is repeated. We observed only one unnatural case for KnowWT5 and WT5, respectively, and none for the other two models. We eliminated the two cases from the counts.

### A.5 Design of Human Evaluation Process for Assessing NLE Quality

For the human evaluation, we sampled 200 generated NLEs for each model. Three Anglophone annotators are employed per instance. We selected annotators with a Lifetime HITs acceptance rate of at least 98% and an accepted number of HITs greater than 1,000. However, it is widely known that the quality of MTurk annotation is not guaranteed even for Master workers (Rouse, 2019). When we used the e-ViL evaluation framework off-the-shelf (Kayser et al., 2021), we found that many workers do annotations without due consideration by simply checking “yes” in most cases. We also initially obtained an inter-annotator agreement captured by Fleiss’s Kappa ( $\mathcal{K}$ ) of only 0.06 on average for Cos-E, which casted doubt on the quality of the evaluation. This prompted us to add a quality control measure to the evaluation framework. We carefully collected *trusted examples* where the quality of the NLEs is objectively “yes” or “no”. For each HIT consisting of 10 examples, we incorporated in random locations two trusted examples with the correct answers being “yes” and “no”, respectively. After annotation, we discarded the HITs where

Model	e-SNLI			Cos-E		
	e-ViL	W/Yes	W/No	e-ViL	W/Yes	W/No
CAGE	-	-	-	0.43	46	54
KnowCAGE	-	-	-	<b>0.44</b>	<b>47</b>	<b>53</b>
NILE	0.80	83	17	-	-	-
KnowNILE	<b>0.82</b>	<b>86</b>	<b>14</b>	-	-	-
WT5	0.76	80	20	0.55	55	45
KnowWT5	<b>0.80</b>	<b>84</b>	<b>16</b>	<b>0.56</b>	<b>57</b>	<b>43</b>

Table 6: Human evaluation results on the generated NLEs. The number of “W/Yes” (merged “Yes” and “Weak Yes”) and “W/No” (“No” and “Weak No”) are in %. The best results are in bold.

		BLEU	R-1	R-2	R-L	Meteor	BERT-S
e-SNLI	WT5-base	28.4	45.8	22.5	40.6	33.7	89.8
	KnowWT5-base	<b>30.6</b>	<b>48.2</b>	<b>24.6</b>	<b>43.0</b>	<b>38.0</b>	<b>90.5</b>
	NILE	22.3	41.7	18.7	36.3	30.2	90.0
	KnowNILE	<b>22.4</b>	<b>42.0</b>	<b>18.9</b>	<b>36.5</b>	<b>30.5</b>	<b>90.1</b>
Cos-E	WT5-base	7.3	25.0	8.3	21.6	20.2	86.3
	KnowWT5-base	<b>7.9</b>	<b>26.7</b>	<b>9.6</b>	<b>22.9</b>	<b>21.8</b>	<b>86.7</b>
	CAGE	3.0	9.7	<b>1.1</b>	9.0	6.3	85.1
	KnowCAGE	3.0	<b>9.8</b>	1.0	9.0	<b>6.4</b>	85.1

Table 7: Automatic evaluation results on generated NLEs on the e-SNLI and Cos-E datasets. The notations R-1, R-2, R-L, and BERT-S denote ROUGE-1, ROUGE-2, ROUGE-L score, and BERT-Score, respectively. The best results are in bold.

the annotators gave a wrong answer for any of the trusted examples (we consider correct a “weak yes” answer for a “yes” trusted example and a “weak no” for a “no” trusted example). We repeated this process until the number of rejected HITs was fewer than 15% of the total HITs. We achieved an increased  $\mathcal{K}$  value of 0.46 and 0.34 for e-SNLI and Cos-E, respectively, from 0.35 and 0.06 (without trusted examples). Similar levels of  $\mathcal{K}$  as ours were obtained in other studies, such as (Marasović et al., 2022; Yordanov et al., 2022).

## A.6 Quality Evaluation on the Generated NLEs

Table 6 shows the detailed results of human evaluation on the quality of generated NLEs. In addition to the e-ViL score, we followed the evaluation method of Marasović et al. (2020) by merging *weak no* and *weak yes* to *no* and *yes*, respectively, and reporting the ratios of *w/yes* and *w/no*. Also, the results of the automatic evaluation metrics are provided in Table 7. The results show that all the Know-models show similar or better results than their original counterparts.

Subject	Relation	Object
men	Antonym	humans
man	Antonym	person
woman	Antonym	person
people	Antonym	person
flower	DistinctFrom	plant
politician	Antonym	man
children	Antonym	people

Table 8: List of filtered noisy triplets in ConceptNet.

## A.7 Examples

ORIGINAL EXPLANATION	REVERSE EXPLANATION
Not all men are teaching science. A dog is not a car. The boy is not necessarily looking at another boy. A child is not a man. A bird is not a squirrel. A group of dogs is not a woman.	Not all men are teaching biology. A dog is not a bike. The boy is not necessarily looking at another female. A child is not a wife. A bird is not a moose. A group of dogs is not a person.

Table 9: Examples where both the original and reverse NLEs contain negation expressions. These NLEs are not contradictory with each other.

PREMISE: Two hussars sit perched on horses, dressed in extravagant ceremonial wear, each holding a sabre in their right hand, reigns to the horse in their left. HYPOTHESIS: There are professional riders at a ceremony. PREDICTED LABEL: Entailment EXPLANATION: Hussars are professional riders.	HYPOTHESIS: Two amateur riders are riding horses. PREDICTED LABEL: Entailment EXPLANATION: Hussars are amateur riders.
PREMISE: A cheerleader in a tight red and white uniform is passing out white t-shirts at a sporting event. HYPOTHESIS: A player passes out hotdogs. PREDICTED LABEL: Contradiction EXPLANATION: A cheerleader is not a player.	HYPOTHESIS: A player is passing out shirts. PREDICTED LABEL: Entailment EXPLANATION: A cheerleader is a player.
PREMISE: Two people using a water buffalo to cultivate a watery field. HYPOTHESIS: Two people are outside with animals. PREDICTED LABEL: Entailment EXPLANATION: A water buffalo is an animal.	HYPOTHESIS: Two people are using a plant. PREDICTED LABEL: Entailment EXPLANATION: A water buffalo is a plant.
QUESTION: Crabs live in what sort of environment? CHOICES: bodies of water, saltwater, galapagos PREDICTED ANSWER: bodies of water EXPLANATION: Crabs live in bodies of water.	CHOICES: bodies of earth, saltwater, atlantic ocean PREDICTED ANSWER: bodies of earth EXPLANATION: Crabs live in bodies of earth.
QUESTION: The piece of paper was worth a lot of money, it was an old Apple Inc what? CHOICES: stock certificate, copy machine, ream PREDICTED ANSWER: stock certificate EXPLANATION: A stock certificate is the only thing that is not a piece of paper.	CHOICES: stock certificate, piece of stone, book PREDICTED ANSWER: stock certificate EXPLANATION: A stock certificate is the only thing that is a piece of paper.
QUESTION: When a person admits his mistakes, what are they doing? CHOICES: act responsibly, learn to swim, feel relieved PREDICTED ANSWER: act responsibly EXPLANATION: when a person admits his mistakes, they act responsibly.	CHOICES: act responsibly, think critically, act irresponsibly PREDICTED ANSWER: act irresponsibly EXPLANATION: when a person admits his mistakes, they act irresponsibly.

Table 10: Examples of inconsistent NLEs detected by eKnowIA for the WT5-base model on e-SNLI and Cos-E. The first column shows the original variable part and the second column shows the adversarial one.

PREMISE: These are young adults who seem to be working together to protect the plants surrounding the white pole. HYPOTHESIS: The adults are young. PREDICTED LABEL: Entailment EXPLANATION: Young adults are young.	HYPOTHESIS: The young adults are underdeveloped. PREDICTED LABEL: Entailment EXPLANATION: Young adults are seniors.
PREMISE: A young boy with close-cropped hair, wearing a red robe, is holding a black kettle as someone is about to pour something in it. HYPOTHESIS: a small boy is holding a kettle that will soon be filled PREDICTED LABEL: Entailment EXPLANATION: A young boy is a small boy.	HYPOTHESIS: The small boy is not underdeveloped. PREDICTED LABEL: Entailment EXPLANATION: A young boy is a small female.
PREMISE: A group of people in a room, most with computers, some raising their hands. HYPOTHESIS: Some people have questions. PREDICTED LABEL: Neutral EXPLANATION: People raising their hands are not asking questions.	HYPOTHESIS: A group of people are asking questions. PREDICTED LABEL: Contradiction EXPLANATION: People raising their hands are asking questions.

Table 11: Examples of inconsistent NLEs detected by our eKnowIA attack for the NILE model on e-SNLI. The first column shows the original hypothesis, and the second one shows the adversarial hypothesis from our attack.

QUESTION: A good interview after applying for a job may cause you to feel what? CHOICES: hope, income, offer PREDICTED ANSWER: hope EXPLANATION: hope is the only thing that would cause you to feel hope.	CHOICES: hope, resentment, fear REVERSE ANSWER: hope EXPLANATION: hope is the only thing that would cause you to feel fear.
QUESTION: What does a stove do to the place that it's in? CHOICES: warm room, brown meat, gas or electric PREDICTED ANSWER: warm room EXPLANATION: a stove heats the room.	CHOICES: warm room, cook food, heat the outside REVERSE ANSWER: heat the outside EXPLANATION: a stove heats the outside.

Table 12: Examples of inconsistent NLEs detected by our eKnowIA attack for the CAGE model on Cos-E. The first column shows the original hypothesis, and the second column shows the adversarial hypothesis from our attack.

PREMISE: A dog standing near snow looking at water. HYPOTHESIS: A bird is standing near snow. PREDICTED LABEL: contradiction EXPLANATION: A dog is not a bird.	HYPOTHESIS: A bird is near water. PREDICTED LABEL: entailment EXPLANATION: A dog is a bird.
QUESTION: What is a person who is good at sports considered? CHOICES: talented, affluent, reproduce PREDICTED LABEL: talented EXPLANATION: a person who is good at sports is considered talented.	CHOICES: talented, untalented, good at PREDICTED LABEL: untalented EXPLANATION: a person who is good at sports is considered untalented

Table 13: Examples of inconsistent NLEs detected by eKnowIA for NILE on e-SNLI and WT5 on Cos-E. The first column shows the original variable part, and the second column shows the adversarial one.

PREMISE: A dog standing near snow looking at water. HYPOTHESIS: A bird is standing near snow. PREDICTED LABEL: contradiction EXPLANATION: A dog is not a bird. EXTRACTED KNOWLEDGE: {snow, RelatedTo, water}, {dog, DistinctFrom, cat}	HYPOTHESIS: A bird is near water. PREDICTED LABEL: neutral EXPLANATION: A dog looking at water does not imply a bird is near water. EXTRACTED KNOWLEDGE: {snow, RelatedTo, water}, {dog, DistinctFrom, cat}
QUESTION: What is a person who is good at sports considered? CHOICES: talented, affluent, reproduce PREDICTED LABEL: talented EXPLANATION: a person who is good at sports is considered talented. EXTRACTED KNOWLEDGE: {talent, RelatedTo, sports}	CHOICES: talented, untalented, good at PREDICTED LABEL: talented EXPLANATION: a person who is good at sports is considered talented. EXTRACTED KNOWLEDGE: {talent, RelatedTo, sports}

Table 14: Examples of successfully defended instances by the KnowNILE model on e-SNLI and by the KnowWT5 model on Cos-E. This table should be read together with Table 13 to appreciate the defence.

MODEL: KnowWT5	
PREMISE: A young family enjoys feeling ocean waves lap at their feet.	
ORIGINAL	ADVERSARIAL
HYPOTHESIS: A family is at the beach.	HYPOTHESIS: The family is not at the beach.
PREDICTED LABEL: entailment	PREDICTED LABEL: entailment
EXPLANATION: Ocean waves lap at their feet implies that they are at the beach.	EXPLANATION: Ocean waves lap at their feet implies that they are not at the beach.
	EXTRACTED KNOWLEDGE: {feet, HasA, fingerprints}
MODEL: KnowNILE	
PREMISE: Some dogs are running on a deserted beach.	
ORIGINAL	ADVERSARIAL
HYPOTHESIS: There are multiple dogs present.	HYPOTHESIS: There are not many dogs on the beach.
PREDICTED LABEL: entailment	PREDICTED LABEL: entailment
EXPLANATION: Some dogs implies multiple dogs.	EXPLANATION: Some dogs implies not many dogs.
	EXTRACTED KNOWLEDGE: {dog, DistinctFrom, cat}
MODEL: KnowCAGE	
QUESTION: What does the sky do before a rain?	
CHOICES: cloud over, rain water, appear blue	CHOICES: cloud over, appear blue, appear green
PREDICTED LABEL: appear blue	PREDICTED LABEL: appear green
EXPLANATION: the sky appears blue before a rain	EXPLANATION: the sky appears green before a rain
	EXTRACTED KNOWLEDGE: {sky, UsedFor, rain}

Table 15: Examples of inconsistent NLEs detected by eKnowIA but not defended by Know-models. The extracted knowledge triplets are not highly related to generating correct explanations.

MODEL: KnowWT5	
PREMISE: The collie is standing outdoors on a sandy area.	
ORIGINAL	ADVERSARIAL
HYPOTHESIS: The collie is standing in the sand.	HYPOTHESIS: The collie is standing on stone.
PREDICTED LABEL: entailment	PREDICTED LABEL: entailment
EXPLANATION: A sandy area is made of sand.	EXPLANATION: A sandy area is made of stone.
	EXTRACTED KNOWLEDGE: {sand, RelatedTo, rock}
MODEL: KnowNILE	
PREMISE: Coach talks with football player, other players and crowd in background.	
ORIGINAL	ADVERSARIAL
HYPOTHESIS: A football player is climbing into the stands at a game.	HYPOTHESIS: A football player talks to a crowd.
PREDICTED LABEL: contradiction	PREDICTED LABEL: entailment
EXPLANATION: A coach is not a football player.	EXPLANATION: A coach is a football player.
	EXTRACTED KNOWLEDGE: {crowd, IsA, gathering}, {player, PartOf, team}, {football player, DerivedFrom, football}

Table 16: Examples of newly detected instances with inconsistent NLEs by eKnowIA for the KNOW models. The extracted knowledge triplets exhibit low relevance and confuse the model to generate incorrect explanations.

## ACL 2023 Responsible NLP Checklist

---

### A For every submission:

- A1. Did you describe the limitations of your work?  
*We discuss our limitations in the "Limitations" section.*
- A2. Did you discuss any potential risks of your work?  
*Not applicable. Left blank.*
- A3. Do the abstract and introduction summarize the paper's main claims?  
*Abstract and Section 1. Introduction*
- A4. Have you used AI writing assistants when working on this paper?  
*Left blank.*

### B Did you use or create scientific artifacts?

*Left blank.*

- B1. Did you cite the creators of artifacts you used?  
*No response.*
- B2. Did you discuss the license or terms for use and / or distribution of any artifacts?  
*No response.*
- B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?  
*No response.*
- B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?  
*No response.*
- B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?  
*No response.*
- B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.  
*No response.*

### C Did you run computational experiments?

*Sections 3 and 4.*

- C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?  
*Appendix A.1.*

---

*The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.*

- C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?  
*Sections 2 and 4.*
- C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?  
*Table 2.*
- C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?  
*Section 4, Appendix A.1 A. 4.*
- D**  **Did you use human annotators (e.g., crowdworkers) or research with human participants?**  
*Appendix A.5.*
- D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?  
*Appendix A.5.*
- D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?  
*Appendix A.5.*
- D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?  
*Section 4.1. and Appendix A.5.*
- D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?  
*Not applicable. Left blank.*
- D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?  
*Appendix A.5.*