Few-Shot Out-of-Domain Transfer Learning of Natural Language Explanations

Yordan Yordanov[1], Vid Kocijan[1], Thomas Lukasiewicz[1,2], Oana-Maria Camburu[1]
[1]Department of Computer Science, University of Oxford
[2]Alan Turing Institute, London
firstname.lastname@cs.ox.ac.uk

Abstract

Recently, there has been an increasing interest in models that generate natural language explanations (NLEs) for their decisions. However, training a model to provide NLEs requires the acquisition of task-specific NLEs, which is time- and resource-consuming. A potential solution is the out-of-domain transfer of NLEs from a domain with a large number of NLEs to a domain with scarce NLEs but potentially a large number of labels, via few-shot transfer learning. In this work, we introduce three vanilla approaches for few-shot transfer learning of NLEs for the case of few NLEs but abundant labels, along with an adaptation of an existing vanilla fine-tuning approach. We transfer explainability from the natural language inference domain, where a large dataset of human-written NLEs exists (e-SNLI), to the domains of (1) hard cases of pronoun resolution, where we introduce a small dataset of NLEs on top of the WinoGrande dataset (small-e-WinoGrande), and (2) commonsense validation (ComVE). Our results demonstrate that the transfer of NLEs outperforms the single-task methods, and establish the best strategies out of the four identified training regimes. We also investigate the scalability of the best methods, both in terms of training data and model size.

1 Introduction

Recent developments have made it possible for AI models to learn from natural language explanations (NLEs) for the ground-truth labels at training time and generate such explanations for their decisions at deployment time [Park et al., 2018, Camburu et al., 2018, Hendricks et al., 2016, Kim et al., 2018, Ling et al., 2017, Rajani et al., 2019, Camburu et al., 2020, Narang et al., 2020, Kumar and Talukdar, 2020]. Such models are inspired by how humans learn (not only from labels but also from demonstrations and explanations Lombrozo, 2012, 2006) and explain themselves in natural language.

In order to train a model to generate NLEs, it is required that humans annotate a training dataset with NLEs. However, large datasets of explanations, such as e-SNLI [Camburu et al., 2018], are time-consuming and expensive to gather. One approach to solve this problem is to transfer explanations from a different domain, via few-shot or zero-shot transfer learning. The usual setup for few-shot out-of-domain transfer learning consists of transfer learning from a "parent" task, with abundant training examples, to a "child" task that only has a few training examples [Thrun, 1996, Ravi and Larochelle, 2017]. In a contemporary work, Marasović et al. [2021] show that prompt engineering can help in few-shot out-of-domain transfer of NLEs in the case where the training labels are also scarce.

In this work, we assume that apart from the few training NLEs on the child task and the abundant NLEs on the parent task, there are abundant training labels for both tasks. Given the advent of deep learning in the last years, one may easily find themselves in this scenario. If one already has a large dataset with labels as child task, it would not be of any use to give up a large proportion of it just

Workshop on Deep Generative Models and Downstream Applications at NeurIPS 2021.

so that the few-shot regime applies equally to labels and NLEs. To our knowledge, there is only one existing work in this setting, that of Erliksson et al. [2021], who introduce a vanilla fine-tuning method on top of the zero-shot WT5 model [Narang et al., 2020]. Their work follows that of Narang et al. [2020] who show a proof-of-concept for NLE transfer across domains without any training NLEs on the child task (zero-shot), but who use the largest T5 model (with 11B parameters) [Raffel et al., 2019] to obtain those results. Erliksson et al. [2021] adapt this approach to the more practical few-shot setup via a simple fine-tuning method on top of a smaller zero-shot WT5 model [Narang et al., 2020]. Unfortunately, the strength of their conclusions is limited by the fact that they use only automatic evaluation metrics, which have been shown to only weakly correlate with human judgment [Kayser et al., 2021].

In this work, we introduce three few-shot transfer learning methods for NLEs that utilize the abundant training labels for both parent and child tasks. Together with the fourth method adapted from Erliksson et al. [2021], they are vanilla combinations of multi-task learning and fine-tuning between a parent and a child tasks with few training NLEs but abundant labels. We instantiate our few-shot learning approaches on e-SNLI [Camburu et al., 2018] as parent task and WinoGrande [Sakaguchi et al., 2020] and ComVE [Wang et al., 2020] as child tasks. As the WinoGrande dataset does not come with NLEs, we introduce small-e-WinoGrande, which provides 100/50/100 NLEs for the training, development, and test sets, respectively.[1] We show the extent to which few-shot out-of-domain transfer learning of NLEs is currently feasible, and provide insight into which learning techniques work best in this setup. We perform both human and automatic evaluation and compare against single-task and zero-shot baselines. We also investigate the scalability of the best approaches, both in terms of data and model sizes.

2 Experimental Setup

2.1 Datasets

e-SNLI. The task of natural language inference [Dagan et al., 2006] is a common task for measuring natural language understanding. It consists of a premise and a hypothesis which are in a relation of either (i) *entailment* (if the premise entails the hypothesis), (ii) *contradiction* (if the hypothesis contradict the premise), or (iii) *neutral* (if neither entailment nor contradiction holds). The e-SNLI dataset [Camburu et al., 2018] consists of human-written explanations on top of the SNLI dataset [Bowman et al., 2015]. An example from e-SNLI is:

> **Premise:** An adult dressed in black holds a stick.
> **Hypothesis:** An adult is walking away, empty-handed.
> **Label:** contradiction
> **Explanation:** Holds a stick implies using hands so it is not empty-handed.

We select e-SNLI as parent dataset due to its large size (570K instances) and high-quality NLEs.

WinoGrande. We select WinoGrande [Sakaguchi et al., 2020] as a child task, since it requires implicit knowledge, which we want to capture in the NLEs. The WinoGrande dataset consists of 40,398 binary questions of pronoun resolution that follow the Winograd Schema format [Levesque et al., 2012]. Because of the lack of a publicly available test set (testing happens through its leaderboard,[2] which has submission limitations), we do a random split of the original training dataset into 39,130 training instances (called WG-train) and 1,268 validation instances (called WG-dev). For testing, we use the original WinoGrande development set, which we denote by WG-test. We manually construct NLEs for 100 examples from WG-train, 50 examples from WG-dev, and 100 examples from WG-test. We call this dataset small-e-WinoGrande. An example is:

> The geese prefer to nest in the fields rather than the forests because in the ___ predators are very visible.
> **Options:** fields, forests. **Answer:** fields.
> **Explanation:** The fields are more open spaces than the forests, hence predators are more visible there.

[1] small-e-WinoGrande is available at `https://github.com/YDYordanov/Few-shot-NLEs`.

[2] `https://leaderboard.allenai.org/winogrande/submissions/public`

Table 1: T5 input/target formats for each task, used for all models.

Task	Input Format	Target Format
e-SNLI	explain nli premise: [premise] hypothesis: [hypothesis]	[relation] explanation: [explanation]
small-e-WinoGrande	explain schema: [schema start] __ [schema end] options: [option 1], [option 2].	[correct option] explanation: [explanation]
ComVE	explain ComVE Sentence 1: [statement 1] Sentence 2: [statement 2]	[nonsensical statement id] explanation: [explanation]

Table 2: Legend of the model names. Notations should be read from top to bottom.

Abbreviation	Meaning
[PT]	The full training dataset of the parent task, with explanations.
[CT]	The full training dataset of the child task, without explanations.
e[CT][number]	The dataset formed by a [number] of training examples from CT with explanations.
T5B / T5L	T5-base / T5-large pre-trained models.
[model]–([datasets])	Fine-tuning of the [model] on the union of the [datasets].
([datasets])	T5B–([datasets]), by default.
WT5	T5B–(e-SNLI, SNLI, CT)
WT5–CT	T5B–(e-SNLI, SNLI)–CT
Heuristic baseline	A ComVE baseline that uses the correct statement as an NLE.

ComVE. We also select Commonsense Validation and Explanation (ComVE) [Wang et al., 2020] as a child task, because it is a commonsense reasoning task for which there are good-quality human-generated NLEs. Originally, ComVE consists of three tasks: A, B, and C, where only tasks A and C are relevant for this work. ComVE-A is the classification task of identifying which statement out of a pair of statements does not make sense. The ComVE-C task provides only the statement that does not make sense (from the pair) and requires the model to generate an NLE for why that is the case. In order to form a classification task with explanations, we merge tasks A and C by matching the nonsensical statements, as done by Majumder et al. [2021]. The resulting task can be described as: "given a pair of sentences, identify which one does **not** make sense, and explain why", which we refer to simply as ComVE. Here is an example from the resulting ComVE dataset:

> **Statement 1:** He drinks milk.
> **Statement 2:** He drinks apple.
> **Label:** Statement 2 (does not make sense).
> **Explanation:** An apple is a whole food and unable to be drunk without being juiced.

The ComVE dataset consists of 10,000 training, 1,000 validation, and 1,000 test instances. Each instance consists of a pair of statements, a label, and three human-generated NLEs. We use all three NLEs per example only in the full test set. For training, we use up to one NLE per example, assuming a strict few-shot regime where each one NLE annotation is expensive to get. For human evaluation, we randomly sample the test dataset down to 100 instances, to save human-annotation costs.

2.2 Base Model

We use the T5 [Raffel et al., 2019] generative language model due to its good generative abilities, and because it is used in the WT5 model to generate high-quality NLEs [Narang et al., 2020]. More specifically, we choose the "Base" model [Raffel et al., 2019] with 220M parameters (we call it T5-Base) due to its good trade-off of performance and computational requirements.

For T5, tasks are distinguished only via their task-specific input/target formats. We follow the input/target format for e-SNLI from Narang et al. [2020], and obtain the input formats for WinoGrande and ComVE in a similar manner (see Table 1). When training on examples without NLEs, "explain" and "explanation:" are not included in the input/target format. We observed in early experiments that the exact choice of input/target formats does not significantly affect performance.

2.3 Few-Shot Transfer Learning Methods

In Table 2, we describe all models that we use. This includes the three new few-shot transfer learning methods for NLE generation, namely (PT, CT, eCT[number]), (PT, CT)–eCT[number], and PT–CT–eCT[number], and a fourth method which we adapt from Erliksson et al. [2021]: PT–(CT, eCT[number]). These four methods correspond to all combinations of fine-tuning (–) and multi-task learning (in brackets) between a parent task (PT) and a child task (CT) with [number] of NLEs used for few-shot transfer (eCT[number]). Note that when training on the union of the child dataset (CT) and eCT[number] ([number] of NLEs from CT), we avoid repeating examples from CT that overlap with eCT[number].

The method by Erliksson et al. [2021] differs from PT–(CT, eCT[number]) by using the union of the parent dataset (PT) with and without explanations. Erliksson et al. [2021] follows this choice from Narang et al. [2020], where this training trick is used to improve zero-shot prediction by helping the model to switch between classification and NLE generation modes on the child task. In our case, the availability of (few) training NLEs for the child task makes this redundant, and the proposed four models only use the parent dataset with NLEs.

Along with the four few-shot transfer learning methods, we add two single-task baselines that aim to verify the extent to which the parent task helps with the transfer of NLEs. The first single-task baseline, T5B–(CT, eCT[number]), is trained on the child task with all labels but only [number] NLEs. The second single-task baseline, T5B–CT–eCT[number], is first trained on the child task and then fine-tuned on [number] NLEs.

To measure the contribution of the few training NLEs, we also introduce two zero-shot baselines, called WT5 and WT5–CT. The WT5 baseline is the training approach from Narang et al. [2020], which consists of multi-task learning on the union of the e-SNLI, the SNLI, and the child dataset. WT5–CT is a variation of WT5 that uses the child task for fine-tuning instead of multi-task learning. These baselines combine the e-SNLI and SNLI datasets in a multi-task setting, to train the model to switch between classification and NLE generation for a better zero-shot downstream performance.

For ComVE, we also add a heuristic baseline (as in [Majumder et al., 2021]), given by selecting as an NLE the correct statement of the pair of statements. This baseline serves to judge the triviality of the NLEs generated by the other approaches.

The training objective is given by cross-entropy loss with targets as described in Table 1. The rest of the training details can be found in Appendix A.

2.4 Human Evaluation

We use Amazon Mechanical Turk to evaluate the model-generated NLEs, with three annotators per instance. The evaluation procedure for each test example is in three steps and follows existing works [Kayser et al., 2021, Majumder et al., 2021, Marasović et al., 2021]. First, annotators are required to predict the correct classification label for the example. This forces them to resolve the example themselves. Second, they have to select one of four options for whether the NLE is a valid and satisfactory explanation to justify the selected label: Yes, Weak Yes, Weak No, or No. Third, they have to select shortcomings of the explanation, if any. The multiple-choice options are: "does not make sense", "insufficient justification", "irrelevant to the task", "too trivial", and "none". These choices may not only provide insight into the problems that the NLEs may have, but also guide the annotators to carefully think about the answer to the main question about NLE quality.

As suggested by Kayser et al. [2021], for each example, the annotators are provided with two (shuffled) NLEs, one from a model and one ground-truth from the test set. This serves for mentally grounding the annotator's score of the model-generated NLE.

Additionally, there are multiple checks placed in the data collection form to ensure high-quality annotations. Most notably, in each group of 10 instances, at least 90% of the labels have to be answered correctly and at least 90% of the ground-truth NLEs have to be annotated by Yes or Weak Yes. The final check requires that at most 80% of the model-generated NLEs should be annotated by Yes or Weak Yes. We included this check to ensure that the annotators are more critical, and we estimated this threshold manually. These are reasonable assumptions for both WinoGrande and ComVE, judging by the quality of the ground-truth and model-generated NLEs.

For each of the two child tasks, all models are evaluated on 100 examples from the test dataset of the task. Similarly to previous works [Camburu et al., 2018, Kayser et al., 2021, Majumder et al., 2021], the NLE evaluation is only done on correctly labeled examples, as it is expected that an incorrect label is not supported by the model with a correct explanation. For each model, we report the percentage of each of the four responses given by the annotators: Yes, Weak Yes, Weak No, and No. See Appendix B for screenshots of the forms that were used to collect the data from the annotators.

We had 130 annotators for ComVE and 113 for WinoGrande. Most of the annotators annotated only ten model-generated NLEs each. To further ensure high-quality annotations, we re-annotated all the instances of the annotators who annotated many instances (more than 60 for WinoGrande and more than 100 for ComVE) but selected more than five wrong shortcomings from a sample of ten random instances, after manual inspection. We found two such annotators for ComVE and one for WinoGrande. The annotators were paid 1$ per 10 pairs of NLEs.

3 Results

Following Kayser et al. [2021], we use an aggregated score (we call "NLE score") of the four categories (Yes, Weak Yes, No, Weak No) to compare the NLE generation quality, where Yes, Weak Yes, Weak No, and No are given weights 1, $2/3$, $1/3$, and 0, respectively. This aggregation has two goals: first, to provide a single metric to compare the methods, and second, to account for the subjective nature of choosing between close labels such as Yes and Weak Yes.

For every model comparison, we report the statistical significance via the paired Student's t-test for equal variances [Yuen and Dixon, 1973], with single-tailed p-values and 0.05 statistical significance threshold. We assume that all individual scores are independent.

For all models, we report the inter-annotator agreement on the scores (Yes, Weak Yes, Weak No, No) via the Fleiss' kappa measure [Fleiss et al., 1971]. Higher values of Fleiss' kappa mean that the annotators agree more about the scores. The kappa values can be interpreted as suggested by Landis and Koch [1977], where negative values signify poor agreement, values between 0.01 and 0.20 are slight agreement, and values between 0.21 and 0.40 are fair agreement. In this work, we do not obtain values higher than 0.40, and most values are around or higher than 0.10.

Overall, the observed inter-annotator agreement is slight-to-fair (see Table 3, 3 and 5), and models trained on ComVE yield higher agreement scores than those trained on small-e-WinoGrande. For each of the two child tasks, we manually analysed a random sample of 28 instances with diverse NLE scores, and estimate that over 50% of the disagreement cases are due to subjectivity, and less than half are due to potential annotator errors (in our opinion), despite the sheer amount of quality-checks inserted throughout the annotation framework. An example of subjectivity on the NLE quality from WinoGrande is: *James wanted to wear the corsage but it wouldn't fit around his wrist because his _ was too small. NLE: The corsage would not fit around the wrist if it was too small.* The annotators gave: Weak No, Yes, and No, which can all be valid as the NLE is somewhat trivial but technically correct, hence some annotators may be satisfied with it while others not.

Note that the Fleiss' kappa is not the best fit for our four categories because closer categories such as Yes and Weak Yes would get the same disagreement as more distant categories such as Yes and No. Furthermore, particularly for skewed distribution of categories such as in WT5–CT for WinoGrande (Table 3), the low kappa value of 0.06 contradicts the overwhelming 87.2% No score. This is a good example of what Randolph [2010] describe as "prevalence and bias, which can lead to the paradox of high agreement but low kappa". However, we report the Fleiss' kappa as it is the standard metric in the literature for the same evaluation framework [Marasović et al., 2021, Majumder et al., 2021] and out of a lack of a better metric.

3.1 WinoGrande

Quantitative results. The results in Table 3 show that out of the four compared approaches, only (PT, CT)–eCT and PT–CT–eCT outperform all baselines in terms of the aggregated NLE score, but only (PT, CT)–eCT outperforms them in a statistically significant way ($p < 0.05$). Amongst the two best approaches, (PT, CT)–eCT outperforms PT–CT–eCT in terms of NLE score, but the difference is not statistically significant, with $p = 0.3$. Both (PT, CT)–eCT and PT–CT–eCT, which use the 50 child task's NLEs in separate training regimes, significantly outperform (PT, CT, eCT) and PT–(CT, eCT), which use a combination of the child dataset with 50 NLEs. This suggests that the 50 NLEs require their own training regime, as they are insignificant relative to the sizes of WinoGrande (approx. 40k) and e-SNLI (approx. 570k). Another possible explanation for the low quality of NLEs of (PT, CT, eCT) and PT–(CT, eCT) could be the close-to-chance task accuracy of these models (53.6% and 54.6%, resp.).

Table 3: Performance of models based on T5-Base on WinoGrande and ComVE as child tasks (CT). The columns Yes, Weak Yes, Weak No, and No present the percentages of NLE validity scores given by the human annotators. Only correctly classified examples are included in these scores. The final column shows the inter-annotator agreement measured by Fleiss' kappa. Best results are in bold. We do not bold the Weak Yes and Weak No since it is not clear that higher/lower is better.

WG Model	**WG acc%**	**Acc @100**	**NLE score**	**Yes%**	**Weak Yes%**	**Weak No%**	**No%**	**Fleiss' kappa**
CT–eCT	59.7	63	34.7	17.5	20.1	11.6	50.8	0.11
(CT, eCT)	57.2	**66**	35.9	20.7	15.2	15.2	49.0	0.15
WT5–CT	**60.2**	65	8.7	4.6	4.1	4.1	87.2	0.06
WT5	58.0	55	8.3	4.8	3.0	4.2	87.9	0.16
(PT, CT, eCT)	53.6	49	28.3	14.3	14.3	13.6	57.8	0.12
(PT, CT)–eCT	56.0	63	**44.1**	**25.9**	18.0	18.5	**37.6**	0.1
PT–(CT, eCT)	54.6	54	29.6	15.4	14.8	13.0	56.8	0.08
PT–CT–eCT	58.2	65	41.9	22.6	22.6	12.8	42.1	0.2
ComVE Model	**ComVE acc%**	**Acc @100**	**NLE score**	**Yes%**	**Weak Yes%**	**Weak No%**	**No%**	**Fleiss' kappa**
CT–eCT	**87.8**	**88**	31.4	25.4	7.2	3.8	63.6	0.31
(CT, eCT)	83.1	79	27.7	23.6	4.2	3.8	68.4	0.3
WT5–CT	85.7	85	28.9	20.0	11.8	3.1	65.1	0.06
WT5	76.2	72	23.9	15.3	10.2	5.6	69.0	0.15
(PT, CT, eCT)	82.8	82	40.2	28.5	14.6	6.1	50.8	0.07
(PT, CT)–eCT	80.6	79	40.6	27.4	17.7	4.2	50.6	0.31
PT–(CT, eCT)	85.5	76	38.6	30.3	8.8	7.5	53.5	0.21
PT–CT–eCT	86.5	79	48.5	36.7	14.3	6.8	**42.2**	0.22
Heuristic baseline	n/a	100	**49.3**	**39.0**	13.3	4.3	43.3	0.06

The four few-shot transfer learning models underperform their single-task counterparts in terms of accuracy on WinoGrande. This is likely due to the domain shift between parent and child tasks.

All non-zero-shot models significantly outperform the zero-shot baselines (WT5–CT and WT5) in terms of NLE score across all four categories (Yes, Weak Yes, Weak No, No).

Shortcomings. The annotated shortcomings of the NLEs are informative of the issues that current generated NLEs have. We summarize the findings here. For full results, see Table 7 in Appendix C. Unsurprisingly, worse-performing models in terms of NLE score (such as WT5 and WT5–CT) produce significantly more nonsensical NLEs (71.1% and 60.8%) when compared to the other models (maximum of 43.7%). On the contrary, better-performing models produce more partially correct NLEs ("insufficient justification"), with a minimum of 29.3% for PT–(CT, eCT) compared to 12.8% and 20.3% for the single-task baselines WT5 and WT5–CT, respectively.

Qualitative results. While most of the correct NLEs that are generated by the models are trivial, such as this example from the PT–CT–eCT model:

> Logan ended the one-side relationship they were having with Robert, which made ___ feel relieved.
> **Answer:** Logan. **Explanation:** Logan ending a one-side relationship makes Logan feel relieved.

there are non-trivial NLEs generated by the models, such as the two below generated by (PT, CT)–eCT:

> Christopher declined when asked by Ian to play a game of chess because ___ liked to play intellectual games.
> **Answer:** Ian. **Explanation:** Chess is a type of intellectual game, and Christopher declined when asked by Ian to play a game of chess.
>
> Brett was browsing the internet while he found the information unlike Randy, ___ prefers using books.
> **Answer:** Randy. **Explanation:** If Brett is browsing the internet, then he is more likely to find information than Randy, who prefers to use books.

For more qualitative results, see Appendix D, where six NLEs from all models are presented.

Table 4: Automatic evaluation of the models with ComVE as child task (CT). B-1,2,3,4 stand for BLEU-1,2,3,4.

Model	**ComVE ppl**	**B-1**	**B-2**	**B-3**	**B-4**	**METEOR**	**BERTScore**
CT–eCT	5.21	**45.2**	**29.5**	**19.5**	**13.1**	**21.5**	83.4
(CT, eCT)	9.51	27.4	16.6	10.2	6.4	19.1	81.8
WT5–CT	65.25	24.6	15.1	9.7	6.5	13.5	74.8
WT5	36.15	22.8	12.0	6.4	3.6	12.7	71.5
(PT, CT, eCT)	8.02	34.5	19.2	10.8	6.3	20.3	81.8
(PT, CT)–eCT	**5.11**	43.5	26.3	16.5	10.6	20.0	83.1
PT–(CT, eCT)	8.18	33.6	18.8	10.9	6.2	20.8	82.1
PT–CT–eCT	5.13	44.4	27.5	17.5	10.7	21.2	**83.6**
Heuristic baseline	n/a	40.8	25.8	17.2	12.0	18.7	81.4

3.2 ComVE

Quantitative results. The results in Table 3 show that the PT–CT–eCT model significantly outperforms all single-task and zero-shot baselines, and all three other compared methods, in terms of NLE score, in a statistically significant way (p-values of at most 0.03). PT–CT–eCT performs weaker than the heuristic baseline in terms of NLE score (48.5 vs. 49.3), but not in a statistically significant way.

In terms of ComVE test accuracy, PT–(CT, eCT) and PT–CT–eCT perform the best out of the four main models, which, like in the WinoGrande results, suggests that the parent and child tasks need separate training regimes in order to preserve the classification accuracy on the child tasks.

Shortcomings. While the NLE score does not reflect a significant difference between the Heuristic and the PT–CT–eCT model, the significant difference between them lies in terms of the shortcomings of NLEs that they produce, as provided by the annotators. PT–CT–eCT has significantly more "does not make sense" (18.8% vs 5.2%), whereas the heuristic baseline has significantly more "insufficient justification" (36.8% vs 28.2%). This reflects the fact that the NLEs of the heuristic baseline are just a copy of the correct statement. The full results of the shortcomings are presented in the Table 7 in Appendix C.

Qualitative results. Upon manual inspection, the correct model-generated NLEs are one of two types. The first is a repetition of the correct statement when it explains why the nonsensical sentence is wrong, e.g., from CT–eCT:

> **Statement 1:** The fire will burn you if you touch it.
> **Statement 2:** The fire will cool you if you touch it.
> **Answer:** Statement 2 does not make sense. **Explanation:** The fire will burn you if you touch it.

The second type are negations of the nonsensical statement. E.g., from (PT, CT, eCT):

> **Statement 1:** He inserts his thumb into the car ignition switch.
> **Statement 2:** He inserts the keys into the car ignition switch.
> **Answer:** Statement 1 does not make sense. **Explanation:** One cannot insert his thumb into the car ignition switch.

For more qualitative results, see Appendix D, where six NLEs from all models are presented.

3.3 Automatic Evaluation

In Table 4, we additionally evaluate all models on the full ComVE test set (1,000 examples with three NLEs per example) via automatic metrics. Automatic evaluation provides additional insights to the human evaluation, since the human evaluation was done on ten times fewer examples. We do not compute automatic metrics w.r.t. WinoGrande, since its NLE test set contains only 100 examples for which we already have the human evaluation, the gold standard for NLE evaluation. We report the ComVE test perplexity, along with BLEU [Papineni et al., 2002], METEOR [Banerjee and Lavie, 2005], and BERTScore [Zhang et al., 2020b], with METEOR and BERTScore having been shown to have the best (although still low) correlation with human judgment of NLEs across several datasets [Kayser et al., 2021]. As with the human evaluation, we report all automatic metrics on the correctly classified examples only.

Table 5: Scalability performance. The first line in each dataset section shows the results when replacing T5-Base with T5-Large in the best performing model in terms of NLE score for each dataset (as per Table 3). The other models for each dataset show the scalability w.r.t. the number of NLEs for the child task. The columns Yes, Weak Yes, Weak No, and No present the percentages of NLE validity scores given by the human annotators. Only correctly classified examples are included in these scores. The final column shows the inter-annotator agreement measured by Fleiss' kappa. Best results are in bold. We do not bold the Weak Yes and Weak No since it is not clear that higher/lower is better.

WG Model	**Acc**	**NLE**	**Yes%**	**Weak**	**Weak**	**No%**	**Fleiss'**
	@100	**score**		**Yes%**	**No%**		**kappa**
T5L–(PT, CT)–eCT50	**68**	**49.5**	**27.0**	27.9	11.8	**33.3**	0.11
(PT, CT)	64	10.6	3.6	7.3	6.2	82.8	0.19
(PT, CT)–eCT25	64	40.5	19.3	25.5	12.5	42.7	0.15
(PT, CT)–eCT50	63	44.1	25.9	18.0	18.5	37.6	0.1
(PT, CT)–eCT100	63	40.4	20.6	22.2	14.8	42.3	0.06
ComVE Model	**Acc**	**NLE**	**Yes%**	**Weak**	**Weak**	**No%**	**Fleiss'**
	@100	**score**		**Yes%**	**No%**		**kappa**
T5L–PT–CT–eCT50	**87**	45.8	29.5	22.2	4.6	43.7	0.21
PT–CT	83	25.4	10.4	18.9	7.2	63.5	0.02
PT–CT–eCT25	82	43.2	26.4	22.4	5.7	45.5	0.19
PT–CT–eCT50	79	48.5	36.7	14.3	6.8	42.2	0.22
PT–CT–eCT100	81	46.2	32.5	18.5	4.1	44.9	0.32
PT–CT–eCT200	79	**49.9**	**35.9**	18.6	5.1	**40.5**	0.33

In terms of test perplexity, aligned with the human evaluation, the (PT, CT)–eCT and PT–CT–eCT models outperform all other models (the heuristic baseline is not included here as there is no definition of perplexity for it). On the contrary, although the (PT, CT, eCT) and (PT, CT)–eCT models are similar in terms of human evaluation, they differ significantly on test perplexity (8.02 vs. 5.11). This is similar for PT–(CT, eCT) and PT–CT–eCT (8.18 vs. 5.13), which reflects the fact that fine-tuning on the 50 NLEs (eCT) in a separate training regime yields a better fit to the test NLE distribution, as given by test perplexity. This is confirmed by the results on BLEU, where fine-tuning on the 50 NLEs produces much more low-level features (B-1, B-2, B-3, and B-4) that match with the test dataset.

In terms of BLEU, METEOR, and BERTScore, PT–CT–eCT outperforms all three other main models. The best-performing baseline, CT–eCT, outperforms PT–CT–eCT in terms of BLEU score but is similar in terms of METEOR and BERTScore. This suggests that it produces NLEs that are closer to the test NLEs in terms of low-level features (unigram, bigram, trigram, and four-grams). This can be explained by the fact that many training NLEs resemble one of the two statements in ComVE, because often the correct statement is a trivial NLE for the instance. The BERTScore and METEOR results on the full test dataset confirm that the PT–CT–eCT model performs significantly better than the heuristic baseline.

3.4 Method Scalability

After selecting the best training approaches from the previous experiments, we investigate the model performance over various explanation dataset sizes and by training with a larger language model (T5-Large) [Raffel et al., 2019]. For WinoGrande, we select (PT, CT)–eCT because it is the only model that significantly outperforms all baselines. For ComVE, we select PT–CT–eCT because it significantly outperforms all other models.

First, we investigate the performance of the best models as we increase the size of the NLE training dataset on the child task. For WinoGrande, we train the best model on up to 100 NLEs (as many as we have). For ComVE, we train the best model on up to 200 NLEs, since ComVE has a vastly larger training set with NLEs. For WinoGrande, the results in Table 5 show that the NLE score improves when having up to 50 training NLEs, but drops with 100 NLEs than with 50 (40.4 vs 44.1), which is confirmed by the percentages of Yes and No. However, the improvement in NLE score from 25 to 50, and the drop from 50 to 100 are not statistically significant. This could suggest that for WinoGrande the model quickly (for up to 25 NLEs) learns how to transfer the explainability knowledge from parent to child task, and may require many more training NLEs (> 100) to start producing significantly better NLEs on the child task. For ComVE, the results in Table 5 show a similar trend between 25 and 100 training NLEs, which is not statistically significant, but the jump from 25 to 200 is statistically significant with $p = 0.04$. This trend is also consistent across Yes and No scores, which confirms that the PT–CT–eCT model scales well with the number of training NLEs. Even if ComVE comes with a larger training set of NLEs, we do not go beyond 200 NLEs for a child task because an investigation of high-resource settings falls beyond the scope of this work.

Second, we train the best methods on each dataset by using T5-Large instead of T5-Base, to verify if larger models can lead to better NLE transfer. For WinoGrande, Table 5 shows that the T5-Large model outperforms T5-Base in terms of NLE score (49.5 vs. 44.1), but it is not statistically significant (p-value of 0.1). The four categories show that while T5-Large is relatively close to T5-Base in terms of Yes, Weak No and No scores, it outperforms it significantly on the Weak Yes score (27.9% vs. 18.0%) with a p-value of 0.01. Furthermore, T5-Large obtains positive scores (Yes or Weak Yes) in much more cases (54.9% vs 43.9% for T5-base), which proves that larger models can obtain more convincing NLEs on this task. For ComVE, the results in Table 5 show that T5-Large underperforms T5-Base in terms of the NLE score (45.8 vs. 48.5), but it is not statistically significant. Similarly to WinoGrande, the two models have similar performance in terms of Yes, Weak No and No scores, but T5-Large significantly outperforms T5-Base in terms of Weak Yes score (22.2% vs 14.3%) with a p-value of 0.01. Our experiments conclude that increasing the language model size from T5-Base to T5-Large does not lead to a significant improvement in the overall NLE quality (NLE score) for either task, but significantly improves the number of plausible NLEs (Weak Yes).

4 Related Work

There are three main focuses in NLE generation: perceived quality improvement [Camburu et al., 2018, Narang et al., 2020, Valentino et al., 2020], NLE faithfulness [Kumar and Talukdar, 2020, Wiegreffe et al., 2021, Liu et al., 2019, Latcinnik and Berant, 2020], and transfer learning of NLEs. In this work, we focus on few-shot out-of-domain transfer learning of NLEs, an area that despite being of high practical importance, has been only little investigated so far. Zero-shot in-domain transfer of NLEs (between datasets of the same task) has been done, e.g., by Camburu et al. [2018], Kumar and Talukdar [2020], and Narang et al. [2020]. Narang et al. [2020] additionally consider zero-shot out-of-domain transfer of NLEs, while Erliksson et al. [2021] extend their work by introducing the first vanilla method for few-shot out-of-domain transfer of NLEs. However, they only evaluated the generated NLEs with automatic metrics, which are notoriously low correlated with human evaluation [Kayser et al., 2021, Camburu et al., 2018]. Contemporary with our work, Marasović et al. [2021] use prompt engineering for few-shot out-of-domain transfer of NLEs, but in the scenario where not only the NLEs but also the labels of the child task are scarce.

In the more general area of natural language generation, few-shot learning is a growing topic [Chen et al., 2020], especially in dialog generation [Peng et al., 2020, Shalyminov et al., 2019]. These approaches, however, do not directly apply to transfer learning of NLEs, which is a dual task of predicting both the label and generating an explanation.

For the task of resolving hard cases of pronoun resolution, there is the WinoWhy [Zhang et al., 2020a] diagnostic dataset for assessing commonsense knowledge in generated NLEs. It is based on the Winograd Schema Challenge dataset [Levesque et al., 2012] and is phrased as a zero-shot NLE classification task. We decided not to use it in this work because we are interested in measuring NLE generation rather than classification of predefined NLEs.

5 Summary and Outlook

In this work, we introduced and compared three vanilla methods for few-shot out-of-domain learning of NLEs and adapted a fourth one from an existing work. We introduced small-e-WinoGrande, a dataset of NLEs on top of a small sample of instances from WinoGrande. We showed that out-of-domain few-shot learning can significantly help with NLE generation compared to zero-shot or single-task learning. Amongst the four NLE few-shot learning methods, we found that the most convincing NLEs are generated by the methods that provide separate training regimes for the child task and its few training NLEs. Finally, we investigated how the best methods scale in terms of model size and NLE training data size. While our results indicate that few-shot out-of-domain transfer learning of NLEs is possible, there is clear room for improvement both in the quality of the generated NLEs and in the task-performance. Thus, our work provides an essential foundation for future research into methods for few-shot out-of-domain transfer learning of NLEs.

References

S. Banerjee and A. Lavie. 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, Ann Arbor, Michigan, June 2005. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/W05-0909.

S. R. Bowman, G. Angeli, C. Potts, and C. D. Manning. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642, Lisbon, Portugal, Sept. 2015. Association for Computational Linguistics. doi: 10.18653/v1/D15-1075. URL https://www.aclweb.org/anthology/D15-1075.

O.-M. Camburu, T. Rocktäschel, T. Lukasiewicz, and P. Blunsom. e-SNLI: Natural language inference with natural language explanations. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems 31*, pages 9539–9549. Curran Associates, Inc., 2018. URL http://papers.nips.cc/paper/8163-e-snli-natural-language-inference-with-natural-language-explanations.pdf.

O.-M. Camburu, B. Shillingford, P. Minervini, T. Lukasiewicz, and P. Blunsom. Make up your mind! Adversarial generation of inconsistent natural language explanations. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 4157–4165, July 2020. doi: 10.18653/v1/2020.acl-main.382. URL https://www.aclweb.org/anthology/2020.acl-main.382.

Z. Chen, H. Eavani, W. Chen, Y. Liu, and W. Y. Wang. Few-shot NLG with pre-trained language model. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 183–190, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.18. URL https://www.aclweb.org/anthology/2020.acl-main.18.

I. Dagan, O. Glickman, and B. Magnini. The pascal recognising textual entailment challenge. In J. Quiñonero-Candela, I. Dagan, B. Magnini, and F. d'Alché Buc, editors, *Machine Learning Challenges. Evaluating Predictive Uncertainty, Visual Object Classification, and Recognising Tectual Entailment*, pages 177–190, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg. ISBN 978-3-540-33428-6. doi: 10.1007/11736790_9.

K. F. Erliksson, A. Arpteg, M. Matskin, and A. H. Payberah. Cross-domain transfer of generative explanations using text-to-text models. In E. Métais, F. Meziane, H. Horacek, and E. Kapetanios, editors, *Natural Language Processing and Information Systems*, pages 76–89, Cham, 2021. Springer International Publishing. ISBN 978-3-030-80599-9.

J. Fleiss et al. Measuring Nominal Scale Agreement Among Many Raters. *Psychological Bulletin*, 76(5):378–382, 1971.

L. Hendricks, Z. Akata, M. Rohrbach, J. Donahue, B. Schiele, and T. Darrell. Generating visual explanations. In *Proceedings of the European Conference on Computer Vision (ECCV)*, volume 9908 of *LNCS*, pages 3–19, 10 2016. ISBN 978-3-319-46492-3. doi: 10.1007/978-3-319-46493-0_1.

J. Howard and S. Ruder. Universal language model fine-tuning for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 328–339, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-1031. URL https://www.aclweb.org/anthology/P18-1031.

M. Kayser, O.-M. Camburu, L. Salewski, C. Emde, V. Do, Z. Akata, and T. Lukasiewicz. e-ViL: A dataset and benchmark for natural language explanations in vision-language tasks. *Computing Research Repository*, arXiv:2105.03761, 2021.

J. Kim, A. Rohrbach, T. Darrell, J. Canny, and Z. Akata. Textual explanations for self-driving vehicles. *Lecture Notes in Computer Science*, page 577–593, 2018. ISSN 1611-3349. doi: 10.1007/978-3-030-01216-8_35. URL http://dx.doi.org/10.1007/978-3-030-01216-8_35.

S. Kumar and P. Talukdar. NILE : Natural language inference with faithful natural language explanations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8730–8742, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.771. URL `https://www.aclweb.org/anthology/2020.acl-main.771`.

J. R. Landis and G. G. Koch. The measurement of observer agreement for categorical data. *Biometrics*, 33(1):159–174, 1977. ISSN 0006341X, 15410420. URL `http://www.jstor.org/stable/2529310`.

V. Latcinnik and J. Berant. Explaining question answering models through text generation. *Computing Research Repository*, arXiv:2004.05569, 2020.

H. J. Levesque, E. Davis, and L. Morgenstern. The Winograd Schema Challenge. In *Proceedings of the 13th International Conference on Principles of Knowledge Representation and Reasoning*, pages 552–561. AAAI Press, 2012. ISBN 9781577355601. URL `https://dl.acm.org/doi/10.5555/3031843.3031909`.

W. Ling, D. Yogatama, C. Dyer, and P. Blunsom. Program induction by rationale generation: Learning to solve and explain algebraic word problems. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 158–167, July 2017. doi: 10.18653/v1/P17-1015. URL `https://www.aclweb.org/anthology/P17-1015`.

H. Liu, Q. Yin, and W. Y. Wang. Towards explainable NLP: A generative explanation framework for text classification. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5570–5581, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1560. URL `https://www.aclweb.org/anthology/P19-1560`.

T. Lombrozo. The structure and function of explanations. In *Trends in Cognitive Sciences*, volume 10, pages 464–70. Cell Press, 11 2006. doi: 10.1016/j.tics.2006.08.004.

T. Lombrozo. Explanation and abductive inference. In *Oxford Handbook of Thinking and Reasoning*, pages 260–276. Oxford University Press, 01 2012. doi: 10.1093/oxfordhb/9780199734689.013.0014.

I. Loshchilov and F. Hutter. Decoupled weight decay regularization. In *Proceedings of the 7th International Conference on Learning Representations*, New Orleans, LA, USA, 2019. OpenReview.net. URL `https://openreview.net/forum?id=Bkg6RiCqY7`.

B. P. Majumder, O.-M. Camburu, T. Lukasiewicz, and J. McAuley. Rationale-inspired natural language explanations with commonsense. *Computing Research Repository*, arXiv:2106.13876, 2021.

A. Marasović, I. Beltagy, D. Downey, and M. E. Peters. Few-shot self-rationalization with natural language prompts. *Computing Research Repository*, arXiv:2111.08284, 2021.

S. Narang, C. Raffel, K. Lee, A. Roberts, N. Fiedel, and K. Malkan. WT5?! Training text-to-text models to explain their predictions. *Computing Research Repository*, arXiv:2004.14546, 2020.

K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA, July 2002. Association for Computational Linguistics. doi: 10.3115/1073083.1073135. URL `https://www.aclweb.org/anthology/P02-1040`.

D. H. Park, L. A. Hendricks, Z. Akata, A. Rohrbach, B. Schiele, T. Darrell, and M. Rohrbach. Multimodal explanations: Justifying decisions and pointing to the evidence. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8779–8788, 2018. doi: 10.1109/CVPR.2018.00915.

B. Peng, C. Zhu, C. Li, X. Li, J. Li, M. Zeng, and J. Gao. Few-shot natural language generation for task-oriented dialog. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 172–182, Online, Nov. 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.findings-emnlp.17. URL `https://www.aclweb.org/anthology/2020.findings-emnlp.17`.

C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Computing Research Repository*, arXiv:1910.10683, 2019.

N. F. Rajani, B. McCann, C. Xiong, and R. Socher. Explain yourself! leveraging language models for commonsense reasoning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4932–4942, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1487. URL https://www.aclweb.org/anthology/P19-1487.

J. Randolph. Free-marginal multirater kappa (multirater kfree): An alternative to fleiss fixed-marginal multirater kappa. In *Advances in Data Analysis and Classification*, volume 4, 01 2010.

S. Ravi and H. Larochelle. Optimization as a model for few-shot learning. In *5th International Conference on Learning Representations, ICLR 2017, Conference Track Proceedings*, Toulon, France, 2017. OpenReview.net. URL https://openreview.net/forum?id=rJY0-Kcll.

K. Sakaguchi, R. Le Bras, C. Bhagavatula, and Y. Choi. WinoGrande: An adversarial winograd schema challenge at scale. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05): 8732–8740, Apr. 2020. doi: 10.1609/aaai.v34i05.6399. URL https://ojs.aaai.org/index.php/AAAI/article/view/6399.

I. Shalyminov, S. Lee, A. Eshghi, and O. Lemon. Few-shot dialogue generation without annotated data: A transfer learning approach. In *Proceedings of the 20th Annual SIGdial Meeting on Discourse and Dialogue*, pages 32–39, Stockholm, Sweden, Sept. 2019. Association for Computational Linguistics. doi: 10.18653/v1/W19-5904. URL https://www.aclweb.org/anthology/W19-5904.

S. Thrun. Is learning the n-th thing any easier than learning the first? In *Advances in Neural Information Processing Systems*, volume 8, pages 640–646. MIT Press, 1996. URL https://proceedings.neurips.cc/paper/1995/file/bdb106a0560c4e46ccc488ef010af787-Paper.pdf.

M. Valentino, M. Thayaparan, and A. Freitas. Explainable natural language reasoning via conceptual unification. *Computing Research Repository*, arXiv:2009.14539, 2020.

C. Wang, S. Liang, Y. Jin, Y. Wang, X. Zhu, and Y. Zhang. SemEval-2020 task 4: Commonsense validation and explanation. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 307–321, Barcelona (online), Dec. 2020. International Committee for Computational Linguistics. URL https://www.aclweb.org/anthology/2020.semeval-1.39.

S. Wiegreffe, A. Marasović, and N. A. Smith. Measuring association between labels and free-text rationales. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 10266–10284, Online and Punta Cana, Dominican Republic, Nov. 2021. Association for Computational Linguistics. URL https://aclanthology.org/2021.emnlp-main.804.

K. K. Yuen and W. J. Dixon. The approximate behaviour and performance of the two-sample trimmed t. *Biometrika*, 60(2):369–374, 1973. ISSN 00063444. URL http://www.jstor.org/stable/2334550.

H. Zhang, X. Zhao, and Y. Song. WinoWhy: A deep diagnosis of essential commonsense knowledge for answering Winograd schema challenge. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5736–5745, Online, July 2020a. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.508. URL https://www.aclweb.org/anthology/2020.acl-main.508.

T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, and Y. Artzi. BERTScore: Evaluating text generation with BERT. In *Proceedings of the 8th International Conference on Learning Representations*, Addis Ababa, Ethiopia, 2020b. OpenReview.net. URL https://openreview.net/forum?id=SkeHuCVFDr.

Table 6: Best hyperparameters for all trained models, along with the corresponding criterion used for model selection, and the best dev result value w.r.t. that criterion. *–subject to the dev accuracy being large enough ($> 75\%$).

Models	**Num epochs**	**Learning rate**	**Criterion**	**Best value**
T5B–PT	3	3e-4	e-SNLI dev NLE ppl	2.192
T5L–PT	–	–	same as T5B	–
T5B–(e-SNLI, SNLI)	3	3e-4	e-SNLI dev NLE ppl	2.199
WG Models				
T5B–(PT, CT)	5	1e-4	WG-dev acc	83.2%
T5L–(PT, CT)	–	–	same as T5B	–
T5B–PT–CT	7	3e-4	WG-dev acc	81.0%
T5B–CT	5	1e-4	WG-dev acc	85.1%
T5B–(WG)–(eWG50)	21	3e-4	WG dev NLE ppl	4.665
T5B–(WG, eWG50)	5	1e-4	WG dev NLE ppl	4.945
WT5–CT	11	3e-4	WG-dev acc	80.8%
WT5	5	1e-4	WG-dev acc	83.4%
(PT, CT, eCT50)	3	3e-5	WG dev NLE ppl	4.815
PT–(CT, eCT50)	5	1e-4	WG dev NLE ppl	5.419
(PT, CT)–eCT50	10	3e-4	WG dev NLE ppl	4.401
PT–CT–eCT50	17	3e-4	WG dev NLE ppl	5.022
T5L–(PT, CT)–eCT50	7	3e-4	WG dev NLE ppl	3.974
(PT, CT)–eCT25	10	3e-4	WG dev NLE ppl	4.684
(PT, CT)–eCT100	7	3e-4	WG dev NLE ppl	4.212
ComVE Models				
T5B–(PT, CT)	3	3e-4	ComVE dev acc	82.8%
T5B–PT–CT	7	3e-4	ComVE dev acc	86.8%
T5L–PT–CT	7	3e-4	ComVE dev acc	89.4%
T5B–CT	5	3e-4	ComVE dev acc	88.4%
T5B–CT–eCT50	13	3e-4	ComVE dev NLE ppl	5.170
T5B–(CT, eCT50)	5	1e-4	ComVE dev NLE ppl*	9.294
WT5–CT	10	3e-4	ComVE dev acc	87.0%
WT5	5	1e-4	ComVE dev acc	84.4%
(PT, CT, eCT50)	5	1e-4	ComVE dev NLE ppl	7.886
PT–(CT, eCT50)	1	1e-3	ComVE dev NLE ppl	7.970
(PT, CT)–eCT50	5	1e-3	ComVE dev NLE ppl	4.958
PT–CT–eCT50	5	1e-3	ComVE dev NLE ppl	5.002
T5L–PT–CT–eCT50: PT–CT–eCT50	7	1e-3	ComVE dev NLE ppl	4.654
PT–CT–eCT25	7	1e-3	ComVE dev NLE ppl	5.274
PT–CT–eCT100	3	1e-3	ComVE dev NLE ppl	4.865
PT–CT–eCT200	3	1e-3	ComVE dev NLE ppl	4.688

A Training Details

We use the AdamW optimizer [Loshchilov and Hutter, 2019] and linear learning rate scheduler with warm-up over 10% of the training. For all models, we fix the batch size to 16 and do a grid search over the learning rate values and the number of training epochs. For all WinoGrande models, we search over the learning rate values of 3e-4, 1e-4, and 3e-5, whereas for ComVE we search over 1e-3, 3e-4, 1e-4, and 3e-5. For e-SNLI, we train on 1, 2, 3, and 5 epochs. For WinoGrande, we train on 1, 2, 3, 5, 7, 9, and 11 epochs, and for ComVE, we train on 1, 2, 3, 5, 7, 10, and 13 epochs. When few-shot fine-tuning with NLEs, we train on 1, 2, 3, 5, 7, 10, 13, 17, 21, and 26 epochs. Multi-task learning always uses the hyperparameter range of the larger dataset. No early stopping is needed, because we use a learning rate scheduler and the number of training epochs is a hyperparameter. We do not use gradual unfreezing Howard and Ruder [2018], because it has been shown that it does not help when applied to the T5 language model Raffel et al. [2019].

At each stage of training, the best hyperparameter combinations are selected via grid search by either the perplexity relative to target NLEs on the dev set of the child task (CT), by dev accuracy on CT, or by NLE perplexity on the e-SNLI dev set, whichever is most suitable. The selection criteria for each model, along with the best hyperparameters are given in Table 6. Note that the WG-dev accuracy in Table 6 is much higher than the corresponding WG-test accuracy in Table 3 because WG-dev is sampled from the training dataset of WinoGrande, whereas WG-test is the original WinoGrande development set, which is filtered to increase its difficulty Sakaguchi et al. [2020]. Model-generated explanations are obtained via beam search with a beam width of 5.

Table 7: Shortcomings provided by the human annotators for all model-generated NLEs. The best results are in bold.

WG Model	**Does not make sense%**	**Insufficient justification%**	**Irrelevant to the schema%**	**Too trivial%**	**None%**
CT–eCT50	32.0	37.0	4.0	7.5	19.5
(CT, eCT50)	33.8	32.4	5.5	6.4	21.9
WT5–CT	60.8	20.3	10.6	4.1	4.1
WT5	71.1	**12.8**	9.6	**2.1**	4.3
(PT, CT, eCT50)	28.0	39.5	8.9	4.5	19.1
(PT, CT)–eCT50	28.1	33.2	6.5	4.0	28.1
PT–(CT, eCT50)	43.7	29.3	6.9	2.3	17.8
PT–CT–eCT50	34.3	33.3	**2.5**	6.9	23.0
T5L–(PT, CT)–eCT50	**19.4**	37.9	4.3	7.1	**31.3**
(PT, CT)	67.6	12.6	11.1	4.3	4.3
(PT, CT)–eCT25	28.9	37.1	7.1	3.0	23.9
(PT, CT)–eCT100	29.0	36.2	3.8	7.6	23.3
ComVE Model	**Does not make sense%**	**Insufficient justification%**	**Irrelevant to the schema%**	**Too trivial%**	**None%**
CT–eCT50	26.9	32.3	12.5	3.6	24.7
(CT, eCT50)	39.8	24.6	10.2	2.7	22.7
WT5–CT	30.7	37.9	**8.9**	3.6	18.9
WT5	36.9	31.7	11.9	5.2	14.3
(PT, CT, eCT50)	22.1	29.0	18.1	4.7	26.1
(PT, CT)–eCT50	23.9	33.5	10.4	4.4	27.9
PT–(CT, eCT50)	32.5	**21.7**	12.0	4.4	29.3
PT–CT–eCT50	18.8	28.2	13.1	2.9	37.1
Heuristic baseline	**5.2**	36.8	15.0	2.9	**40.1**
T5L–PT–CT–eCT50	18.1	34.4	10.0	4.4	33.0
PT–CT	38.1	31.1	12.5	4.3	14.0
PT–CT–eCT25	16.9	37.0	13.8	3.1	29.1
PT–CT–eCT100	14.9	37.3	12.0	2.4	33.3
PT–CT–eCT200	18.2	32.2	9.5	**2.1**	38.0

B Data Collection Forms

Below are screenshots of the data collection forms that we used for WinoGrande (Figure 1) and ComVE (Figure 2).

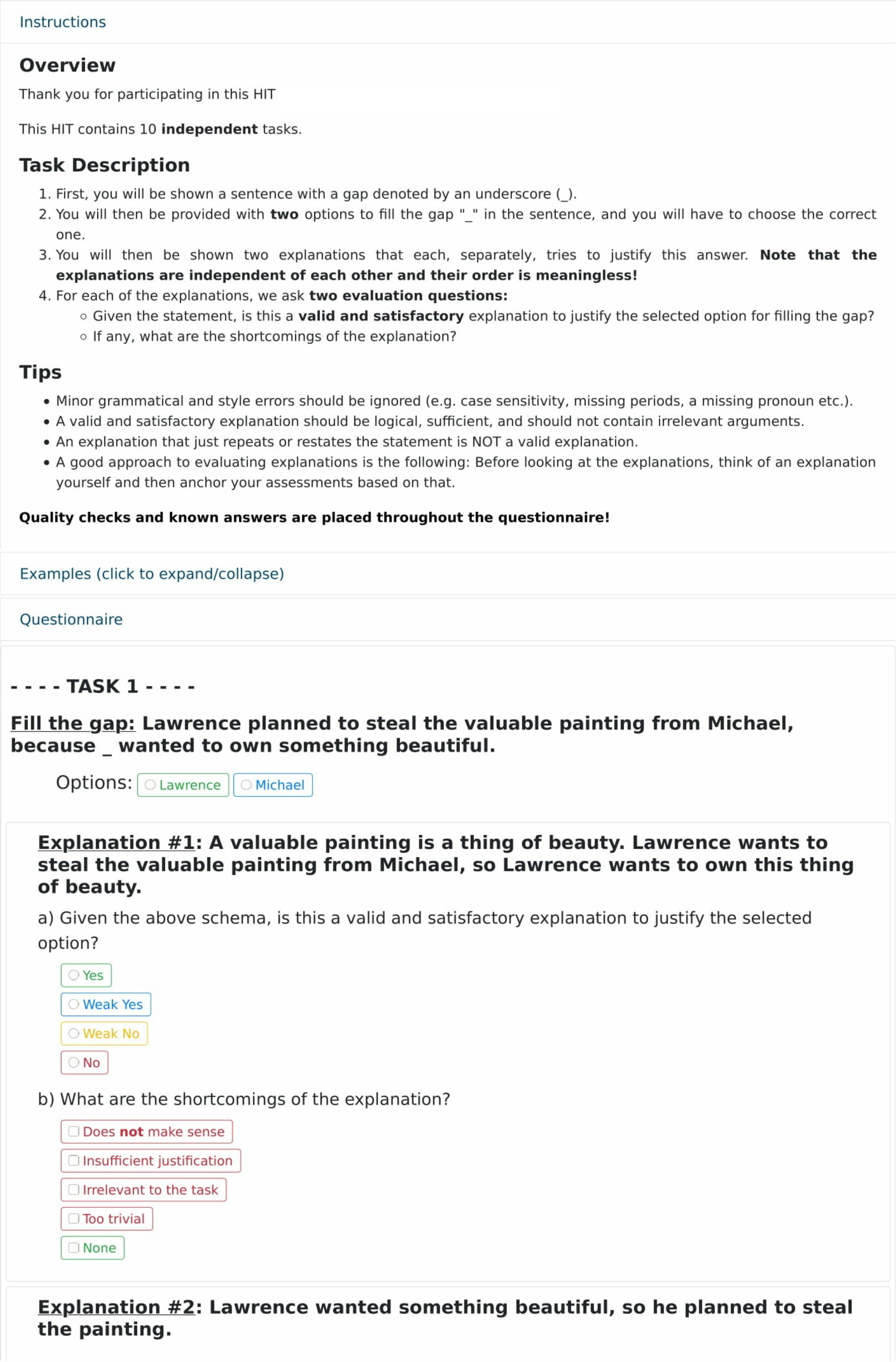
C Additional Results

Table 7 summarizes, for each model, the shortcomings that the human annotators found in the model-generated NLEs.

D Examples of Model-Generated NLEs

In the twelve tables below Table 7 are the answers and NLEs for each child task (WG and ComVE) and for all eight compared models on the first six examples (out of the 100 that were evaluated).

Figure 1: WinoGrande data collection template. There are two explanations per task.



Instructions

Overview

Thank you for participating in this HIT

This HIT contains 10 **independent** tasks.

Task Description

1. First, you will be shown a sentence with a gap denoted by an underscore (_).
2. You will then be provided with **two** options to fill the gap "_" in the sentence, and you will have to choose the correct one.
3. You will then be shown two explanations that each, separately, tries to justify this answer. **Note that the explanations are independent of each other and their order is meaningless!**
4. For each of the explanations, we ask **two evaluation questions:**
   - Given the statement, is this a **valid and satisfactory** explanation to justify the selected option for filling the gap?
   - If any, what are the shortcomings of the explanation?

Tips

- Minor grammatical and style errors should be ignored (e.g. case sensitivity, missing periods, a missing pronoun etc.).
- A valid and satisfactory explanation should be logical, sufficient, and should not contain irrelevant arguments.
- An explanation that just repeats or restates the statement is NOT a valid explanation.
- A good approach to evaluating explanations is the following: Before looking at the explanations, think of an explanation yourself and then anchor your assessments based on that.

Quality checks and known answers are placed throughout the questionnaire!

Examples (click to expand/collapse)

Questionnaire

- - - - TASK 1 - - - -

Fill the gap: Lawrence planned to steal the valuable painting from Michael, because _ wanted to own something beautiful.

Options: ○ Lawrence ○ Michael

Explanation #1: A valuable painting is a thing of beauty. Lawrence wants to steal the valuable painting from Michael, so Lawrence wants to own this thing of beauty.

a) Given the above schema, is this a valid and satisfactory explanation to justify the selected option?

○ Yes

○ Weak Yes

○ Weak No

○ No

b) What are the shortcomings of the explanation?

□ Does **not** make sense

□ Insufficient justification

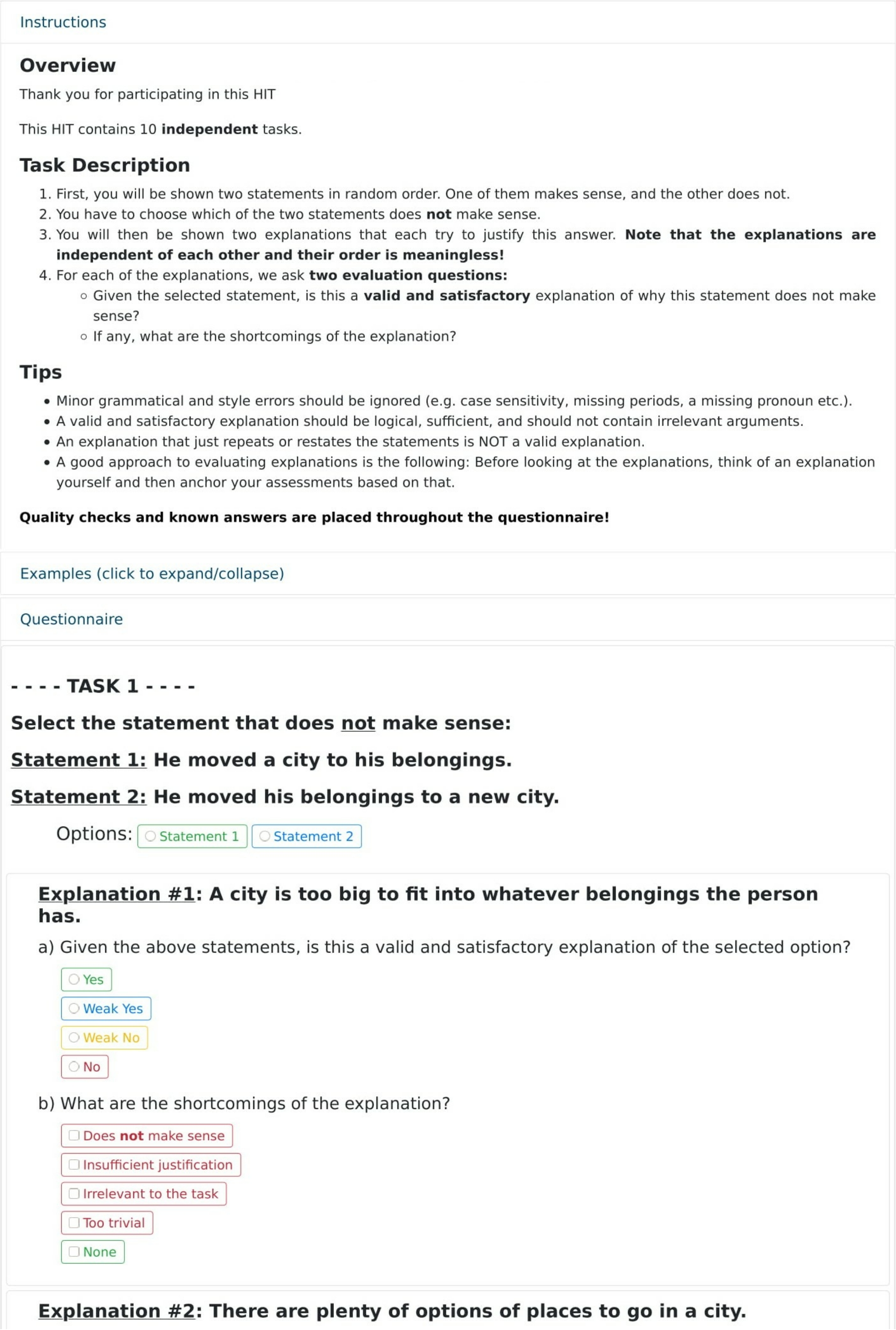
□ Irrelevant to the task

□ Too trivial

□ None

Explanation #2: Lawrence wanted something beautiful, so he planned to steal the painting.

Figure 2: ComVE data collection template. There are two explanations per task.



Instructions

Overview

Thank you for participating in this HIT

This HIT contains 10 **independent** tasks.

Task Description

1. First, you will be shown two statements in random order. One of them makes sense, and the other does not.
2. You have to choose which of the two statements does **not** make sense.
3. You will then be shown two explanations that each try to justify this answer. **Note that the explanations are independent of each other and their order is meaningless!**
4. For each of the explanations, we ask **two evaluation questions:**
   - Given the selected statement, is this a **valid and satisfactory** explanation of why this statement does not make sense?
   - If any, what are the shortcomings of the explanation?

Tips

- Minor grammatical and style errors should be ignored (e.g. case sensitivity, missing periods, a missing pronoun etc.).
- A valid and satisfactory explanation should be logical, sufficient, and should not contain irrelevant arguments.
- An explanation that just repeats or restates the statements is NOT a valid explanation.
- A good approach to evaluating explanations is the following: Before looking at the explanations, think of an explanation yourself and then anchor your assessments based on that.

Quality checks and known answers are placed throughout the questionnaire!

Examples (click to expand/collapse)

Questionnaire

- - - - TASK 1 - - - -

Select the statement that does not make sense:

Statement 1: He moved a city to his belongings.

Statement 2: He moved his belongings to a new city.

Options: ○ Statement 1 ○ Statement 2

Explanation #1: A city is too big to fit into whatever belongings the person has.

a) Given the above statements, is this a valid and satisfactory explanation of the selected option?

○ Yes

○ Weak Yes

○ Weak No

○ No

b) What are the shortcomings of the explanation?

□ Does **not** make sense

□ Insufficient justification

□ Irrelevant to the task

□ Too trivial

□ None

Explanation #2: There are plenty of options of places to go in a city.

Schema: The ___ car was a pleasure to drive but the red car was not as the blue car was very expensive. **Options:** blue, red. **Ground-truth:** blue.	
Model	**Output**
CT–eCT50	**Prediction:** red. **NLE:** The blue car is expensive and the blue car is more expensive.
(CT, eCT50)	**Prediction:** red. **NLE:** The blue car is more expensive than the red car, hence it is more expensive to drive.
WT5–CT	**Prediction:** red. **NLE:** The car is described in more detail than in the second sentence. Blue
WT5	**Prediction:** blue. **NLE:** Red is a different color than blue.
(PT, CT, eCT50)	**Prediction:** red. **NLE:** The car was a pleasure to drive but the blue car was not as the blue car was very expensive.
(PT, CT)–eCT50	**Prediction:** red. **NLE:** The red car is more expensive than the blue car.
PT–(CT, eCT50)	**Prediction:** blue. **NLE:** The blue car is a pleasure to drive and the red car is very expensive.
PT–CT–eCT50	**Prediction:** blue. **NLE:** The car is a pleasure to drive but the red car is red.

Schema: Logan ended the one-side relationship they were having with Robert, which made ___ feel relieved. **Options:** Logan, Robert. **Ground-truth:** Logan.	
Model	**Output**
CT–eCT50	**Prediction:** Robert. **NLE:** Logan ended the one-side relationship they were having with Robert, which made Logan feel relieved.
(CT, eCT50)	**Prediction:** Logan. **NLE:** Robert has a one-sided relationship with Logan, which makes him feel relieved.
WT5–CT	**Prediction:** Logan. **NLE:** The Logan
WT5	**Prediction:** Logan. **NLE:** Robert
(PT, CT, eCT50)	**Prediction:** Robert. **NLE:** Logan ended the one-side relationship they were having with Robert, which made Logan feel relieved
(PT, CT)–eCT50	**Prediction:** Logan. **NLE:** The fact that Logan ended the one-sided relationship with Robert makes him feel relieved.
PT–(CT, eCT50)	**Prediction:** Robert. **NLE:** Logan's feelings are relieved because he ended the relationship with Robert.
PT–CT–eCT50	**Prediction:** Logan. **NLE:** Logan ending a one-side relationship makes Logan feel relieved.

Schema: Fiction books were interesting and easy to read for Logan but not Brett because ___ enjoyed real tales. **Options:** Logan, Brett. **Ground-truth:** Brett.	
Model	**Output**
CT–eCT50	**Prediction:** Brett. **NLE:** Fiction books were interesting and easy to read for Logan, but Brett enjoyed real tales more than fiction books.
(CT, eCT50)	**Prediction:** Brett. **NLE:** Fiction books were interesting and easy to read for Logan, while real tales were more difficult to read for Brett.
WT5–CT	**Prediction:** Logan. **NLE:** There is nothing to indicate that he is enjoying real stories or that he has been referred to as 'Boy'
WT5	**Prediction:** Logan. **NLE:** The Logan may not be Logan. Brett may not be Brett. Logan may not be Logan. Brett may not be Brett. Logan may not be Logan. Brett may not be Brett. Logan may not be Logan. Brett may not be Brett. Loga [...]
(PT, CT, eCT50)	**Prediction:** Brett. **NLE:** Fiction books were interesting and easy to read for Logan but not Brett because Brett enjoyed real tales
(PT, CT)–eCT50	**Prediction:** Brett. **NLE:** Fiction books are not real tales.
PT–(CT, eCT50)	**Prediction:** Logan. **NLE:** Fiction books are interesting and easy to read for Logan, so Logan is more likely to enjoy real tales than Brett.
PT–CT–eCT50	**Prediction:** Logan. **NLE:** Fiction books are not real tales.

Schema: Erin ordered a martini when Tanya only ordered a tonic water with lemon, because ___ was pregnant. **Options:** Erin, Tanya. **Ground-truth:** Tanya.	
Model	**Output**
CT–eCT50	**Prediction:** Tanya. **NLE:** Erin ordered a martini when Tanya ordered tonic water with lemon because she was pregnant.
(CT, eCT50)	**Prediction:** Erin. **NLE:** Erin ordered a martini when Tanya ordered a tonic water with lemon, because Erin was pregnant.
WT5–CT	**Prediction:** Erin. **NLE:** The Tanya is pregnant. Tanya is pregnant. Tanya
WT5	**Prediction:** Tanya. **NLE:** Erin and Tanya are two different people. Erin and Tanya are two different people. Erin and Tanya are two different people. Erin and Tanya are two different people. Erin and Tanya are two different peo [...]
(PT, CT, eCT50)	**Prediction:** Tanya. **NLE:** Erin ordered a martini when Tanya only ordered a tonic water with lemon, because __ was pregnant.
(PT, CT)–eCT50	**Prediction:** Tanya. **NLE:** Erin ordering a martini implies Erin ordering a martini, and Tanya ordering a tonic implies Erin is pregnant.
PT–(CT, eCT50)	**Prediction:** Tanya. **NLE:** Erin ordered a martini
PT–CT–eCT50	**Prediction:** Erin. **NLE:** Erin ordering a martini implies Erin is pregnant. Erin ordering a tonic water implies Erin is pregnant. Erin ordering a martini implies Erin ordering a martini. Erin ordering a tonic water implies Eri [...]

Schema: The game on television was very important for Ian but not Dennis because ___ was into music instead. **Options:** Ian, Dennis. **Ground-truth:** Dennis.	
Model	**Output**
CT–eCT50	**Prediction:** Dennis. **NLE:** The game on television was very important for Ian but not for Dennis because he was into music instead.
(CT, eCT50)	**Prediction:** Dennis. **NLE:** The game on television was very important for Ian, but not for Dennis, because he was into music instead.
WT5–CT	**Prediction:** Dennis. **NLE:** He is playing music instead of playing music instead of playing music instead of playing music instead of playing music instead of playing music instead of playing music instead of playing music inste [...]
WT5	**Prediction:** Dennis. **NLE:** The Ian can't be Dennis. Dennis can't be Ian. Dennis can't be Dennis. Dennis can't be Ian. Dennis can't be Ian. Dennis can't be Dennis. Dennis can't be Ian. Dennis can't be Dennis [...]
(PT, CT, eCT50)	**Prediction:** Dennis. **NLE:** The game on television was very important for Ian but not Dennis because he was into music instead.
(PT, CT)–eCT50	**Prediction:** Dennis. **NLE:** The game on television is more important for Ian than Dennis because Dennis is more likely to be into music instead.
PT–(CT, eCT50)	**Prediction:** Dennis. **NLE:** The game on television is very important for Ian
PT–CT–eCT50	**Prediction:** Dennis. **NLE:** The game on television is very important for Ian, so he is not into music.

Schema: It was easier for the diner to follow their budget than the food truck because the ___ had more money to spend. **Options:** diner, food truck. **Ground-truth:** diner.	
Model	**Output**
CT–eCT50	**Prediction:** diner. **NLE:** The diner has more money to spend than the food truck.
(CT, eCT50)	**Prediction:** diner. **NLE:** The diner has more money to spend than the food truck, so it is easier for them to follow their budget.
WT5–CT	**Prediction:** diner. **NLE:** food truck
WT5	**Prediction:** food truck. **NLE:** diner
(PT, CT, eCT50)	**Prediction:** diner. **NLE:** The diner had more money to spend than the food truck.
(PT, CT)–eCT50	**Prediction:** diner. **NLE:** The diner has more money to spend than the food truck.
PT–(CT, eCT50)	**Prediction:** diner. **NLE:** The diner has more money to spend than the food truck.
PT–CT–eCT50	**Prediction:** diner. **NLE:** More money to spend is a rephrasing of more money to spend.

Statement 1: He inserts his thumb into the car ignition switch. **Statement 2:** He inserts the keys into the car ignition switch. **Nonsensical statement:** Statement 1.

Model	**Output**
CT–eCT50	**Prediction:** Statement 1. **NLE:** A thumb is not inserted into the car ignition switch.
(CT, eCT50)	**Prediction:** Statement 1. **NLE:** It's not a car ignition switch, it's a car ignition switch, it's a car ignition switch, it's a car ignition switch, it's a car ignition switch, it's a car ignition switch, it's a car ignition switch, [. . .]
WT5–CT	**Prediction:** Statement 1. **NLE:** SIDEWALK INSTALLS THE KEYS INTO THE CAR INFUSION SHUT.
WT5	**Prediction:** Statement 1. **NLE:** the keys are not the same as the thumb
(PT, CT, eCT50)	**Prediction:** Statement 1. **NLE:** One cannot insert his thumb into the car ignition switch.
(PT, CT)–eCT50	**Prediction:** Statement 1. **NLE:** One cannot insert his thumb into the car ignition switch.
PT–(CT, eCT50)	**Prediction:** Statement 1. **NLE:** You can't use your thumb to insert keys into the car ignition switch.
PT–CT–eCT50	**Prediction:** Statement 1. **NLE:** A thumb is not used to insert keys into a car ignition switch

Statement 1: She was met with an accident and was rushed to the hospital **Statement 2:** She was met with an accident and was rushed to the library **Nonsensical statement:** Statement 2.

Model	**Output**
CT–eCT50	**Prediction:** Statement 2. **NLE:** The library is closed to the public
(CT, eCT50)	**Prediction:** Statement 2. **NLE:** She was met with an accident and was rushed to the library
WT5–CT	**Prediction:** Statement 2. **NLE:** She was met with an accident and was rushed to the hospital 1 explanation:
WT5	**Prediction:** Statement 2. **NLE:** .
(PT, CT, eCT50)	**Prediction:** Statement 2. **NLE:** The library is not the only place where a person can have an accident.
(PT, CT)–eCT50	**Prediction:** Statement 2. **NLE:** A library is not a hospital.
PT–(CT, eCT50)	**Prediction:** Statement 2. **NLE:** A library is not a hospital.
PT–CT–eCT50	**Prediction:** Statement 2. **NLE:** A library is not a hospital.

Statement 1: She came off sailing across the pacific ocean **Statement 2:** She came off walking across the pacific ocean **Nonsensical statement:** Statement 2.

Model	**Output**
CT–eCT50	**Prediction:** Statement 2. **NLE:** She came off sailing across the pacific ocean
(CT, eCT50)	**Prediction:** Statement 2. **NLE:** She came off sailing across the pacific ocean Sailing across the pacific ocean Sailing across the pacific ocean Sailing across the pacific ocean Sailing across the pacific ocean Sailing across the pac [. . .]
WT5–CT	**Prediction:** Statement 2. **NLE:** She came off sailing across the pacific ocean Sentence 1:
WT5	**Prediction:** Statement 2. **NLE:** .
(PT, CT, eCT50)	**Prediction:** Statement 2. **NLE:** A person can't be sailing and walking at the same time.
(PT, CT)–eCT50	**Prediction:** Statement 2. **NLE:** sailing is not the same as walking
PT–(CT, eCT50)	**Prediction:** Statement 2. **NLE:** Sailing and walking are two different things.
PT–CT–eCT50	**Prediction:** Statement 2. **NLE:** sailing is not the same as walking

Statement 1: Leena daily eats two bananas which are good for health. **Statement 2:** Leena eats almost 1000 bananas daily as his morning breakfast. **Nonsensical statement:** Statement 2.

Model	**Output**
CT–eCT50	**Prediction:** Statement 2. **NLE:** Leena eats two bananas a day which are good for health.
(CT, eCT50)	**Prediction:** Statement 2. **NLE:** Bananas are good for health and you can't eat them in the morning.
WT5–CT	**Prediction:** Statement 2. **NLE:** Leena daily eats two bananas which are good for health 1 explanation:
WT5	**Prediction:** Statement 2. **NLE:** .
(PT, CT, eCT50)	**Prediction:** Statement 2. **NLE:** Leena daily eats two bananas which are good for health. is same as Leena eats almost 1000 bananas daily as his morning breakfast
(PT, CT)–eCT50	**Prediction:** Statement 2. **NLE:** The bananas are good for health.
PT–(CT, eCT50)	**Prediction:** Statement 2. **NLE:** The bananas are not good for health, they are good for health.
PT–CT–eCT50	**Prediction:** Statement 2. **NLE:** The bananas are good for health.

<table>
<tr><td colspan="2"><b>Statement 1:</b> The fire will burn you if you touch it. <b>Statement 2:</b> The fire will cool you if you touch it. <b>Nonsensical statement:</b> Statement 2.</td></tr>
<tr><td><b>Model</b></td><td><b>Output</b></td></tr>
<tr><td>CT–eCT50</td><td><b>Prediction:</b> Statement 2. <b>NLE:</b> The fire will burn you if you touch it.</td></tr>
<tr><td>(CT, eCT50)</td><td><b>Prediction:</b> Statement 2. <b>NLE:</b> The fire will burn you if you touch it and will cool you if you touch it</td></tr>
<tr><td>WT5–CT</td><td><b>Prediction:</b> Statement 2. <b>NLE:</b> The fire will burn you if you touch it.</td></tr>
<tr><td>WT5</td><td><b>Prediction:</b> Statement 2. <b>NLE:</b> the fire will burn you if you touch it.</td></tr>
<tr><td>(PT, CT, eCT50)</td><td><b>Prediction:</b> Statement 2. <b>NLE:</b> The fire will burn you is not the same as the fire that will cool you.</td></tr>
<tr><td>(PT, CT)–eCT50</td><td><b>Prediction:</b> Statement 2. <b>NLE:</b> The fire will burn you if you touch it</td></tr>
<tr><td>PT–(CT, eCT50)</td><td><b>Prediction:</b> Statement 1. <b>NLE:</b> Fire can’t burn you</td></tr>
<tr><td>PT–CT–eCT50</td><td><b>Prediction:</b> Statement 2. <b>NLE:</b> The fire will burn you.</td></tr>
</table>

<table>
<tr><td colspan="2"><b>Statement 1:</b> Salsa is for hair <b>Statement 2:</b> Brushes are for hair <b>Nonsensical statement:</b> Statement 1.</td></tr>
<tr><td><b>Model</b></td><td><b>Output</b></td></tr>
<tr><td>CT–eCT50</td><td><b>Prediction:</b> Statement 1. <b>NLE:</b> Salsa is for hair</td></tr>
<tr><td>(CT, eCT50)</td><td><b>Prediction:</b> Statement 1. <b>NLE:</b> Salsa is for hair</td></tr>
<tr><td>WT5–CT</td><td><b>Prediction:</b> Statement 1. <b>NLE:</b> SENTENCE 1:</td></tr>
<tr><td>WT5</td><td><b>Prediction:</b> Statement 1. <b>NLE:</b> :</td></tr>
<tr><td>(PT, CT, eCT50)</td><td><b>Prediction:</b> Statement 1. <b>NLE:</b> Brushes are for hair is not the same as a Salsa</td></tr>
<tr><td>(PT, CT)–eCT50</td><td><b>Prediction:</b> Statement 1. <b>NLE:</b> Salsa is for hair</td></tr>
<tr><td>PT–(CT, eCT50)</td><td><b>Prediction:</b> Statement 1. <b>NLE:</b> Brushes are for hair is a rephrasing of brushes are for hair.</td></tr>
<tr><td>PT–CT–eCT50</td><td><b>Prediction:</b> Statement 1. <b>NLE:</b> Salsa is for hair</td></tr>
</table>