Few-Shot Out-of-Domain Transfer Learning of Natural Language Explanations in a Label-Abundant Setup

Yordan Yordanov[1], Vid Kocijan[2], Thomas Lukasiewicz[3,1], Oana-Maria Camburu[4]
[1] University of Oxford [2] Kumo.ai [3] TU Wien
[4] University College London
yordan.yordanov@cs.ox.ac.uk, thomas.lukasiewicz@tuwien.ac.at,
vid@kumo.ai, o.camburu@cs.ucl.ac.uk

Abstract

Training a model to provide natural language explanations (NLEs) for its predictions usually requires the acquisition of task-specific NLEs, which is time- and resource-consuming. A potential solution is the few-shot out-of-domain transfer of NLEs from a parent task with many NLEs to a child task. In this work, we examine the setup in which the child task has few NLEs but abundant labels. We establish four few-shot transfer learning methods that cover the possible fine-tuning combinations of the labels and NLEs for the parent and child tasks. We transfer explainability from a large natural language inference dataset (e-SNLI) separately to two child tasks: (1) hard cases of pronoun resolution, where we introduce the small-e-WinoGrande dataset of NLEs on top of the WinoGrande dataset, and (2) commonsense validation (ComVE). Our results demonstrate that the parent task helps with NLE generation and we establish the best methods for this setup.

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 (Hendricks et al., 2016; Ling et al., 2017; Park et al., 2018; Camburu et al., 2018; Kim et al., 2018; Rajani et al., 2019; Camburu et al., 2020; Narang et al., 2020; Kumar and Talukdar, 2020; Marasović et al., 2022). However, large datasets of NLEs, such as e-SNLI (Camburu et al., 2018), are time-consuming and expensive to gather. One approach is to transfer explanations from a different domain, via few-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 this work, we assume that the child task has few training NLEs but abundant labels. Given the advent of deep learning in the last years, this scenario may be quite frequent, as one may already have a large dataset with labels on which they aim to train NLEs-generating models without annotating the entire dataset with NLEs. To our knowledge, there is only one existing work in this setup, that of Erliksson et al. (2021), which introduces a vanilla fine-tuning method on top of the zero-shot WT5 model (Narang et al., 2020). However, their work is limited by: (1) they only test one of the four possible scenarios we identify for this setup, and (2) they use only automatic evaluation metrics, which do not necessarily align with human judgment (Camburu et al., 2018; 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 the parent and child task, and we adapt for computational efficiency the method from Erliksson et al. (2021). Together, these four methods are combinations of multi-task learning and fine-tuning between a parent and a child task 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 contain NLEs, we introduce small-e-WinoGrande, which provides 100/50/100 NLEs for the training, development, and test sets, respectively. 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 human evaluation and compare against child-task-only and zero-shot baselines.[1]

[1] The code and the datasets are publicly available at: https://github.com/YDYordanov/Few-shot-NLEs.

Findings of the Association for Computational Linguistics: EMNLP 2022, pages 3486 - 3501

Task	Input Format	Target Format
e-SNLI	explain nli premise: [premise] hypothesis: [hypothesis]	[relation] explanation: [explanation]
W.G.	explain WinoGrande 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 1: T5 input/target formats for each task, used for all models. When training on examples without NLEs, "explain" and "explanation:" are not included in the input/target format.

2 Experimental Setup

2.1 Datasets

e-SNLI. Natural language inference (NLI) (Dagan et al., 2006) is the task of assigning a relation of *entailment*, *contradiction*, or *neutrality* between a *premise* and a *hypothesis*. The e-SNLI dataset (Camburu et al., 2018) consists of human-written NLEs on top of the Stanford Natural Language Inference (SNLI) (Bowman et al., 2015). We select e-SNLI as parent dataset due to its large size (∼570K) and high-quality NLEs.

WinoGrande. The WinoGrande dataset (Sakaguchi et al., 2020) consists of 40,398 binary fill-in-the-gap instances of pronoun resolution that follow the Winograd Schema format (Levesque et al., 2012). We select WinoGrande as a child task, since it requires implicit knowledge, which we want to capture in the NLEs. We construct the *small-e-WinoGrande* dataset by manually creating NLEs for 100/50/100 training/dev/test instances.

ComVE. Commonsense Validation and Explanation (ComVE) (Wang et al., 2020), as reformulated by Majumder et al. (2022), is the task of jointly identifying which one of two statements contradicts commonsense and explaining why. The dataset consists of 10,000 training, 1,000 validation, and 1,000 test instances. We select ComVE as a child task, because it is a commonsense reasoning task for which there are good-quality human-written NLEs. For more dataset details, see Appendix A.

2.2 Base Model

Similarly to Narang et al. (2020), we use the T5 (Raffel et al., 2020) generative language model, in particular, the "Base" model with 220M parameters, 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 by Narang et al. (2020): *premise: [premise] hypothesis: [hypothesis] / [relation] explanation: [explanation]*. We obtain the input formats for WinoGrande and ComVE in a similar manner (see Table 1). We observed in early experiments that the exact choice of input/target formats does not significantly affect performance.

We choose the best practice for multi-task learning with T5, namely, via training on the union of the datasets in question (Raffel et al., 2020).

2.3 Few-Shot Transfer Learning Methods

Table 2 shows all the models that we use. M1 to M4 are the four few-shot transfer learning methods for NLE generation, which we obtain by combining the parent dataset with NLEs, the child dataset, and a few NLEs (we use 50 in this work) in all reasonable multi-task and fine-tuning combinations. M3 is similar to the method by Erliksson et al. (2021), but the latter uses the union of the parent dataset with and without explanations, mimicking WT5. We choose against this, because in the few-shot NLE case, this is unnecessary and doubles the computation cost.

We also consider four baseline methods. The two child-task baselines CD–fine-tune and CD-union serve to measure the contribution of the parent in NLE transfer. Two zero-shot NLE transfer learning baselines, WT5 (Narang et al., 2020) and WT5–fine-tune, serve to measure the contribution of the 50 training NLEs in the child task. The training details are given in Appendix B.

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 instance is in three steps and follows existing works (Kayser et al., 2021; Majumder et al., 2022; Marasović et al., 2022). First, annotators have to predict the classification label for the example. Second, they have to select one of four options for whether the NLE is a valid and satisfactory explanation for the selected label: Yes, Weak Yes, Weak No, or No. Third, they have to select shortcomings of the explanation from the following: "does not make sense", "insufficient justification", "irrelevant to the task", "too trivial", and "none".

Model name	Meaning
CD–fine-tune	fine-tune T5 on the child dataset, and then fine-tune on 50 NLEs
CD–union	fine-tune T5 on the union of the child dataset and 50 NLEs
WT5–fine-tune	fine-tune T5 on the union of e-SNLI and SNLI, and then fine-tune on the child dataset
WT5	fine-tune T5 on the union of e-SNLI, SNLI, and the child dataset
M1	fine-tune T5 on the union of e-SNLI, the child dataset, and 50 NLEs
M2	fine-tune T5 on the union of e-SNLI and the child dataset, and then fine-tune on 50 NLEs
M3	fine-tune T5 on e-SNLI, and then fine-tune on the union of the child dataset and 50 NLEs
M4	fine-tune T5 on e-SNLI, and then fine-tune on the child dataset, and, finally, on 50 NLEs

Table 2: Legend of the model names. The child dataset excludes the NLEs, unless specified. The 50 NLEs refer to the few (50) instances of the child task with NLEs.

Model	WinoGrande		ComVE		ComVE Automatic NLE Metrics					
	Task acc%	NLE score	Task acc%	NLE score	B-1	B-2	B-3	B-4	METEOR	BERTScore
CD–fine-tune	59.7	34.7	**87.8**	31.4	**45.2**	**29.5**	**19.5**	**13.1**	**21.5**	83.4
CD–union	57.2	35.9	83.1	27.7	27.4	16.6	10.2	6.4	19.1	81.8
WT5–fine-tune	**60.2**	8.7	85.7	28.9	24.6	15.1	9.7	6.5	13.5	74.8
WT5	58.0	8.3	76.2	23.9	22.8	12.0	6.4	3.6	12.7	71.5
M1	53.6	28.3	82.8	40.2	34.5	19.2	10.8	6.3	20.3	81.8
M2	56.0	**44.1***	80.6	40.6	43.5	26.3	16.5	10.6	20.0	83.1
M3	54.6	29.6	85.5	38.6	33.6	18.8	10.9	6.2	20.8	82.1
M4	58.2	41.9*	86.5	**48.5***	44.4	27.5	17.5	10.7	21.2	**83.6**

Table 3: Performance of models on WinoGrande and ComVE as child tasks. From the 100 test examples, only the correctly classified are given NLE scores. B-1,2,3,4 stand for BLEU-1,2,3,4. Best results are in bold; * denotes the statistically significant best results.

All models are evaluated on 100 examples from the test dataset of each child task. Similarly to previous works (Camburu et al., 2018; Kayser et al., 2021; Majumder et al., 2022), the NLE evaluation is only done on correctly labeled (by the model) examples, as it is expected that an incorrect label is not supported by the model with a correct NLE. See Appendix C for more details and for screenshots of the forms used to collect the annotations.

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 the weights 1, $2/3$, $1/3$, and 0, respectively. This aggregation has two goals: (1) to provide a single metric to compare the methods, and (2) to account for the subjective nature of choosing between close labels such as Yes and Weak Yes. A summary of the Yes, Weak Yes, Weak No, and No scores and the shortcomings are presented in Appendix D.

For every model comparison, we report if it is statistically significant 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.

The results are given in Table 3. We only report automatic metrics (BLEU (Papineni et al., 2002), METEOR (Banerjee and Lavie, 2005), and BERTScore (Zhang et al., 2020)) for NLE quality for ComVE, since WinoGrande has only a small number of test NLE instances (which have been used for grounding in our human evaluation – see Appendix C). We notice that the automatic metrics are not well aligned with the human evaluation (NLE score). This has also been previously observed in other studies (Camburu et al., 2018; Kayser et al., 2021). Therefore, we will base our conclusions only on the human evaluation (NLE score).

First, we notice that all methods (M1–M4) significantly outperform the zero-shot baselines (WT5–fine-tune and WT5) in terms of NLE quality for both datasets, which proves the utility of the 50 child-task NLEs.

Second, we see that not all methods outperform the child-task baselines. For example, on WinoGrande, both CD–fine-tune and CD-union outper-

form M1 and M3 in terms of the NLE quality. This shows that it is sometimes possible that fine-tuning on a large parent task of out-of-domain NLEs hurts NLE quality of a child task. However, for both datasets, the best performing method is among the M1–M4 methods (and for ComVE, all M1–M4 methods outperform the child-task baselines), suggesting that it is generally useful to use a large dataset of NLEs as a parent task even when out-of-domain.

Third, we see that the M1–M4 methods rank differently on different datasets, in particular, M2 and M4 are the significantly best methods on WinoGrande, and M4 is the significantly best method on ComVE. We believe that the main difference in method ranking is that the methods obtain much closer-to-chance accuracy on WinoGrande than on ComVE. This is important, because poor task understanding can lead to poor NLE generation. In particular, M2 and M4 obtain significantly better WinoGrande accuracy than M1 and M3, which might explain the significant gap in NLE score. M2 and M4 are the best NLE generation methods on both datasets, which could be because M2 and M4 use the 50 NLEs of the child task in a separate training regime, whereas M1 and M3 use a combination of the much larger child dataset with the 50 NLEs. This suggests that the 50 NLEs require their own training regime to obtain good NLE generation.

Finally, we see that the best task performances on both datasets are obtained by a baseline method: WT5–fine-tune (no child NLEs) for WinoGrande and CD–fine-tune (child-task only) for ComVE. Moreover, among the M1-M4 methods, there is no clear best method in terms of task performance, nor a correlation with the NLE quality. Hence, a trade-off between task performance and NLE quality is needed, which we observe is achieved by M4.

3.1 Qualitative Examples

WinoGrande. While many of the correct NLEs that are generated by the models are trivial, e.g.:

> Fiction books were interesting and easy to read for Logan but not Brett because ___ enjoyed real tales. **A:** Brett.
> **NLE:** Fiction books were interesting and easy to read for Logan but not Brett because Brett enjoyed real tales

there are some non-trivial generations, e.g.:

> Brett was browsing the internet while he found the information unlike Randy, ___ prefers using books. **A:** Randy.
> **NLE:** If Brett is browsing the internet, then he is more likely to find information than Randy, who prefers to use books.

ComVE. By manual inspection, the correct model-generated NLEs are either a repetition of the correct statement, or a negation of the nonsensical statement. Examples from each:

> **Statements:** The fire will [burn / cool] you if you touch it. **A:** burn. **NLE:** The fire will burn you if you touch it.

> **Statements:** He inserts [his thumb / the keys] into the car ignition switch.
> **A:** the keys. **NLE:** One cannot insert his thumb into the car ignition switch.

For more qualitative results, see Appendix E, where six non-cherry-picked NLEs from each model are presented.

4 Related Work

There are three main focuses in NLE generation: quality improvement (Camburu et al., 2018; Narang et al., 2020; Valentino et al., 2022), NLE faithfulness (Kumar and Talukdar, 2020; Wiegreffe et al., 2021; Liu et al., 2019; Latcinnik and Berant, 2020), and transfer learning of NLEs. 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 showing that few-shot out-of-domain transfer of NLEs is possible in the abundant-label setup. Marasović et al. (2022) use prompt engineering for few-shot out-of-domain transfer of NLEs for the scarce-label setup. The prompt choice is less relevant in our abundant-label setup, because task adaptation can be done via the abundant training labels. In the more general area of natural language generation, few-shot learning is a growing topic (Chen et al., 2020), e.g., 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.

5 Summary and Outlook

In this work, we investigated four methods for few-shot out-of-domain transfer learning of NLEs for the abundant-label setting. 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 child-task-only 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. While our results indicate that few-shot out-of-domain transfer learning of NLEs is helpful, there is room for improvement both in the quality of the generated NLEs and in task-performance. Thus, our work provides an essential foundation for future research into few-shot out-of-domain transfer learning of NLEs where label abundance is available.

6 Limitations

The training methods in this work can apply to any language other than English, but a large parent task with NLEs is needed and a high-performance pre-trained generative language model may be needed for that language. Training one of our methods takes approximately three hours per e-SNLI epoch on one NVIDIA TITAN Xp GPU, which should be multiplied by the number of epochs and the hyperparameter combinations used. In practice, we observed that the results are sensitive to the number of epochs and to the choice of the learning rate, so a comprehensive hyperparameter search may be needed. This significantly increases the computational requirements and can be an obstacle for researchers on a limited budget. In total, the required time to reproduce all our results is approximately 45 GPU days. Scaling our methods to larger language models can also be challenging from a computational requirements standpoint.

7 Acknowledgments

This work was supported by an Early Career Leverhulme Fellowship, by the Alan Turing Institute under the EPSRC grant EP/N510129/1, by the AXA Research Fund, and by the EPSRC Studentship OUCS/EPSRC-NPIF/VK/1123106. We also acknowledge the use of the EPSRC-funded Tier 2 facility JADE (EP/P020275/1) and GPU computing support by Scan Computers International Ltd.

References

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72. Association for Computational Linguistics.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. 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. Association for Computational Linguistics.

Oana-Maria Camburu, Tim Rocktäschel, Thomas Lukasiewicz, and Phil Blunsom. 2018. e-SNLI: Natural language inference with natural language explanations. In *Advances in Neural Information Processing Systems 31*, pages 9539–9549. Curran Associates, Inc.

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 *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 4157–4165.

Zhiyu Chen, Harini Eavani, Wenhu Chen, Yinyin Liu, and William Yang Wang. 2020. Few-shot NLG with pre-trained language model. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 183–190. Association for Computational Linguistics.

Ido Dagan, Oren Glickman, and Bernardo Magnini. 2006. The PASCAL recognising textual entailment challenge. In *Machine Learning Challenges. Evaluating Predictive Uncertainty, Visual Object Classification, and Recognising Tectual Entailment*, pages 177–190. Springer.

Karl Fredrik Erliksson, Anders Arpteg, Mihhail Matskin, and Amir H. Payberah. 2021. Cross-domain transfer of generative explanations using text-to-text models. In *Natural Language Processing and Information Systems: 26th International Conference on Applications of Natural Language to Information Systems, NLDB 2021, June 23–25, 2021, Proceedings*, page 76–89. Springer-Verlag.

Lisa Hendricks, Zeynep Akata, Marcus Rohrbach, Jeff Donahue, Bernt Schiele, and Trevor Darrell. 2016. Generating visual explanations. In *Proceedings of the European Conference on Computer Vision (ECCV)*, volume 9908 of *LNCS*, pages 3–19.

Jeremy Howard and Sebastian Ruder. 2018. 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. Association for Computational Linguistics.

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 *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1224–1234.

Jinkyu Kim, Anna Rohrbach, Trevor Darrell, John Canny, and Zeynep Akata. 2018. Textual explanations for self-driving vehicles. *Lecture Notes in Computer Science*, page 577–593.

Sawan Kumar and Partha Talukdar. 2020. 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. Association for Computational Linguistics.

Veronica Latcinnik and Jonathan Berant. 2020. Explaining question answering models through text generation. *CoRR*, arXiv:2004.05569.

Hector J. Levesque, Ernest Davis, and Leora Morgenstern. 2012. The Winograd Schema Challenge. In *Proceedings of the 13th International Conference on Principles of Knowledge Representation and Reasoning*, pages 552–561. AAAI Press.

Wang Ling, Dani Yogatama, Chris Dyer, and Phil Blunsom. 2017. 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.

Hui Liu, Qingyu Yin, and William Yang Wang. 2019. 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. Association for Computational Linguistics.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *Proceedings of the 7th International Conference on Learning Representations*. OpenReview.net.

Bodhisattwa Prasad Majumder, Oana-Maria Camburu, Thomas Lukasiewicz, and Julian McAuley. 2022. Knowledge-grounded self-rationalization via extractive and natural language explanations. *Proceedings of 39th International Conference on Machine Learning (ICML)*.

Ana Marasović, Iz Beltagy, Doug Downey, and Matthew E. Peters. 2022. Few-shot self-rationalization with natural language prompts. In *Findings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics.

Sharan Narang, Colin Raffel, Katherine Lee, A. Roberts, Noah Fiedel, and Karishma Malkan. 2020. WT5?! Training text-to-text models to explain their predictions. *CoRR*, arXiv:2004.14546.

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

Dong Huk Park, Lisa Anne Hendricks, Zeynep Akata, Anna Rohrbach, Bernt Schiele, Trevor Darrell, and Marcus Rohrbach. 2018. Multimodal explanations: Justifying decisions and pointing to the evidence. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8779–8788.

Baolin Peng, Chenguang Zhu, Chunyuan Li, Xiujun Li, Jinchao Li, Michael Zeng, and Jianfeng Gao. 2020. Few-shot natural language generation for task-oriented dialog. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 172–182. Association for Computational Linguistics.

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. *Journal of Machine Learning Research*, 21(140):1–67.

Nazneen Fatema Rajani, Bryan McCann, Caiming Xiong, and Richard Socher. 2019. Explain yourself! leveraging language models for commonsense reasoning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4932–4942. Association for Computational Linguistics.

Sachin Ravi and Hugo Larochelle. 2017. Optimization as a model for few-shot learning. In *5th International Conference on Learning Representations, ICLR 2017, Conference Track Proceedings*. OpenReview.net.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2020. WinoGrande: An adversarial winograd schema challenge at scale. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):8732–8740.

Igor Shalyminov, Sungjin Lee, Arash Eshghi, and Oliver Lemon. 2019. 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. Association for Computational Linguistics.

Sebastian Thrun. 1996. 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.

Marco Valentino, Mokanarangan Thayaparan, and André Freitas. 2022. Case-based abductive natural language inference. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 1556–1568. International Committee on Computational Linguistics.

Cunxiang Wang, Shuailong Liang, Yili Jin, Yilong Wang, Xiaodan Zhu, and Yue Zhang. 2020. SemEval-2020 Task 4: Commonsense validation and explanation. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 307–321. International Committee for Computational Linguistics.

Sarah Wiegreffe, Ana Marasović, and Noah A. Smith. 2021. Measuring association between labels and free-text rationales. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 10266–10284. Association for Computational Linguistics.

Karen K. Yuen and W. J. Dixon. 1973. The approximate behaviour and performance of the two-sample trimmed t. *Biometrika*, 60(2):369–374.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. BERTScore: Evaluating text generation with BERT. In *Proceedings of the 8th International Conference on Learning Representations*. OpenReview.net.

A Datasets

WinoGrande. 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 WinoGrande 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 created the small-e-WinoGrande dataset by manually constructing NLEs for 100 examples from WG-train, 50 examples from WG-dev, and 100 examples from WG-test. Example:

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

ComVE. Originally, ComVE (Wang et al., 2020) 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. To form a classification task with explanations, we merge tasks A and C by matching the nonsensical statements, as done by Majumder et al. (2022). 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. The resulting 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] https://leaderboard.allenai.org/winogrande/submissions/public

B Training Details

The training objective is given by cross-entropy loss with targets as described in Table 1. 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., 2020).

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, by dev accuracy on the child dataset, 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 4. Note that the WG-dev accuracy in Table 4 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.

C Human Evaluation

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

Models	Num epochs	Learning rate	Criterion	Best value
e-SNLI	3	3e-4	e-SNLI dev NLE ppl	2.192
(e-SNLI, SNLI)	3	3e-4	e-SNLI dev NLE ppl	2.199
WinoGrande Models				
(e-SNLI, WinoGrande)	5	1e-4	WG-dev acc	83.2%
e-SNLI–WinoGrande	7	3e-4	WG-dev acc	81.0%
WinoGrande	5	1e-4	WG-dev acc	85.1%
CD–fine-tune	21	3e-4	WG-dev NLE ppl	4.665
CD–union	5	1e-4	WG-dev NLE ppl	4.945
WT5–fine-tune	11	3e-4	WG-dev acc	80.8%
WT5	5	1e-4	WG-dev acc	83.4%
M1	3	3e-5	WG-dev NLE ppl	4.815
M2	5	1e-4	WG-dev NLE ppl	5.419
M3	10	3e-4	WG-dev NLE ppl	4.401
M4	17	3e-4	WG-dev NLE ppl	5.022
ComVE Models				
(e-SNLI, ComVE)	3	3e-4	ComVE dev acc	82.8%
e-SNLI–ComVE	7	3e-4	ComVE dev acc	86.8%
ComVE	5	3e-4	ComVE dev acc	88.4%
CD–fine-tune	13	3e-4	ComVE dev NLE ppl	5.170
CD–union	5	1e-4	ComVE dev NLE ppl*	9.294
WT5–fine-tune	10	3e-4	ComVE dev acc	87.0%
WT5	5	1e-4	ComVE dev acc	84.4%
M1	5	1e-4	ComVE dev NLE ppl	7.886
M2	1	1e-3	ComVE dev NLE ppl	7.970
M3	5	1e-3	ComVE dev NLE ppl	4.958
M4	5	1e-3	ComVE dev NLE ppl	5.002

Table 4: Best hyperparameters for all trained models (including the intermediary models), along with the corresponding criterion used for model selection, and the best dev result value w.r.t. that criterion. The datasets in brackets denotes the model obtained by fine-tuning T5 on the union of those datasets; dataset1–dataset2 denotes subsequent fine-tuning on dataset1, then on dataset2. *–subject to the dev accuracy being large enough ($> 75\%$).

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.

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.

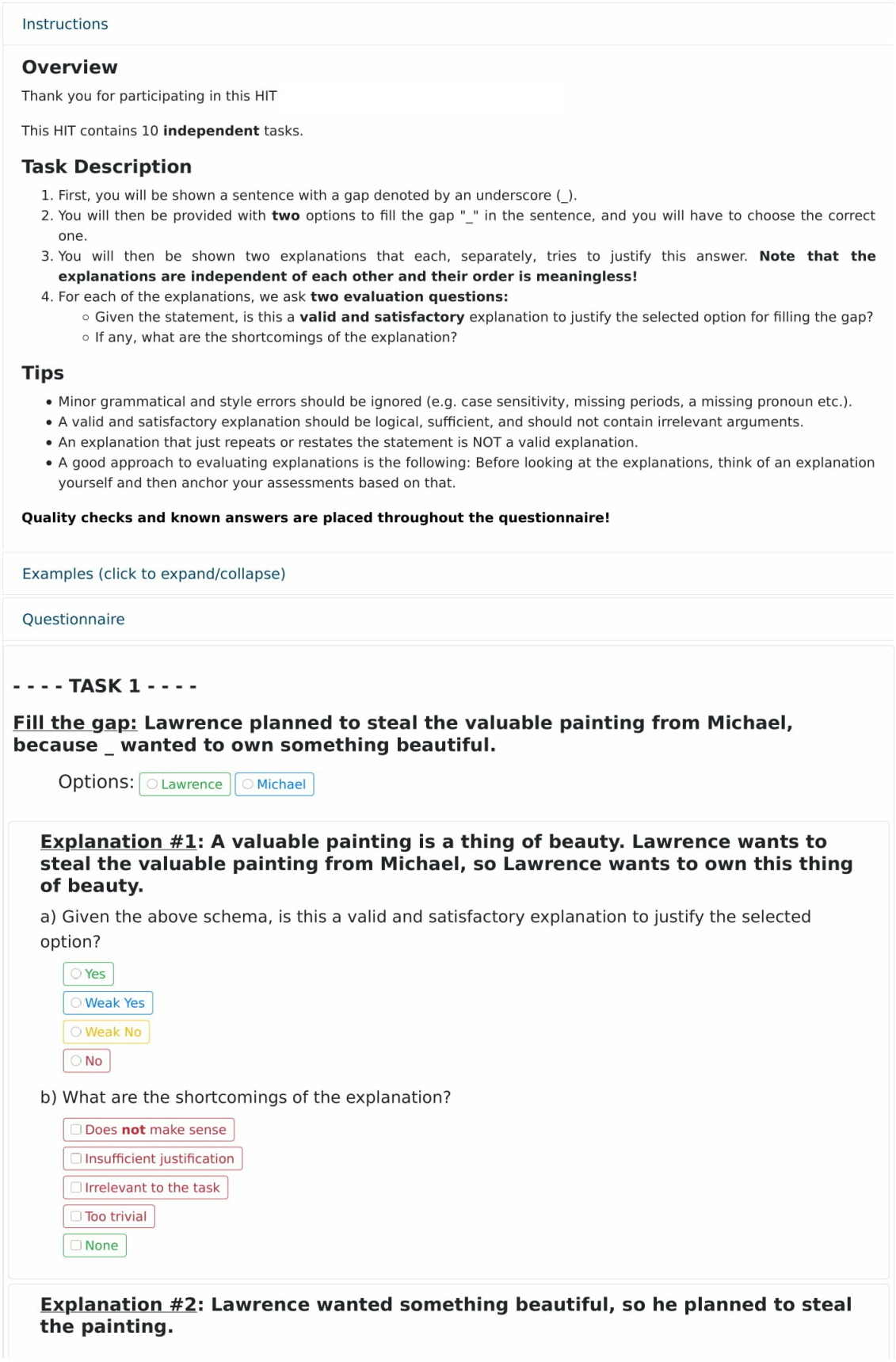
Below are full-page screenshots of the data collection forms that we used for WinoGrande (Figure 1) and ComVE (Figure 2).

D Additional Results

Table 5 presents the full human evaluation results table for all models which includes the separate Yes, Weak Yes, Weak No, and No scores. Table 5 also summarizes, for each model, the shortcomings that the human annotators found in the model-generated NLEs. The annotated shortcomings of the NLEs are informative of the issues that current generated NLEs have.

E Examples of Model-Generated NLEs

In the twelve tables below Figure 2 are the answers and NLEs for each child task (WinoGrande and ComVE) and for all eight compared models on the first six examples (out of the 100 that were evaluated). The first six tables present six examples for WinoGrande, whereas the second six tables are for ComVE.



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

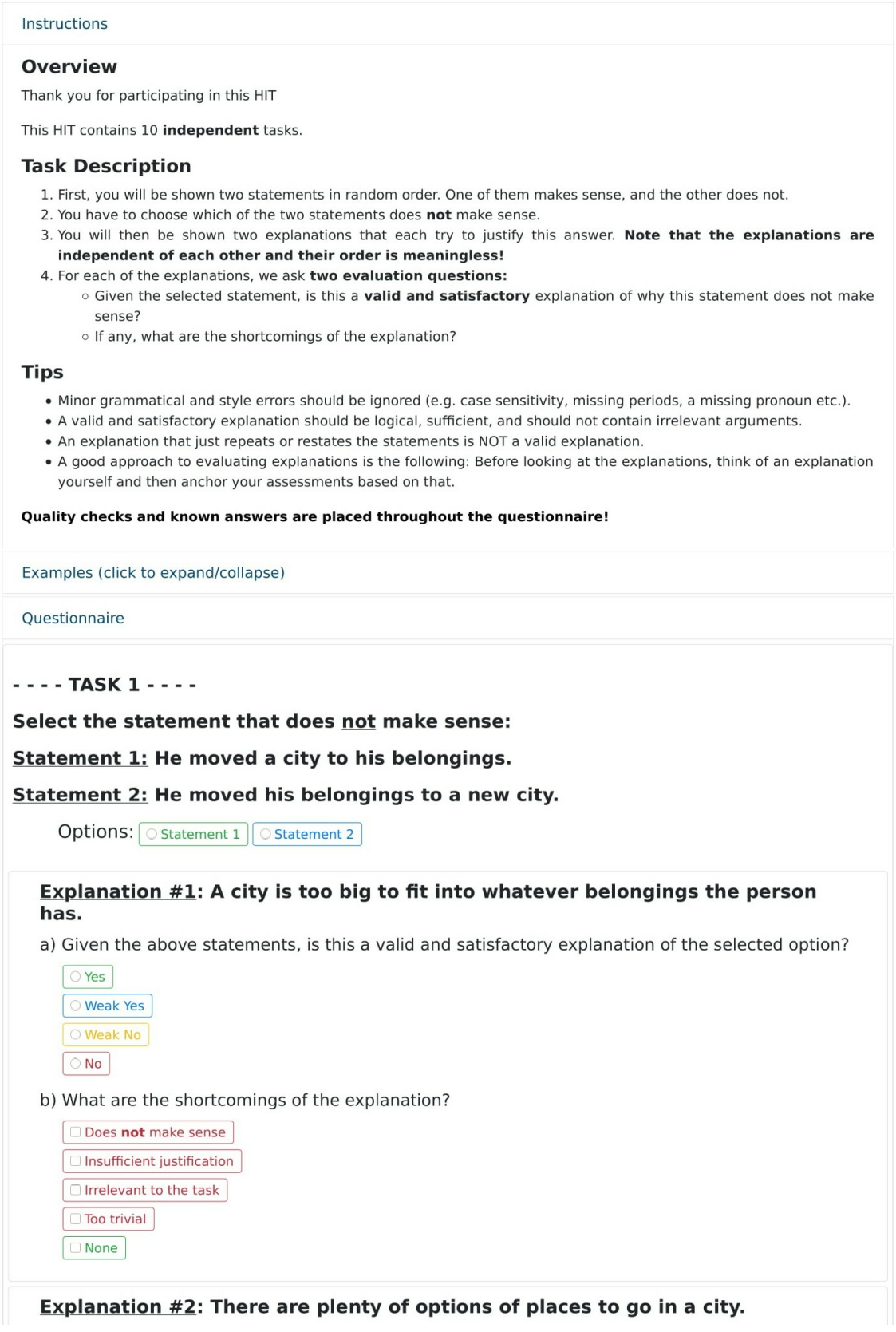


Figure 2: ComVE data collection template. There are two explanations per task.

WinoGrande Model	**NLE score**	**Yes%**	**Weak Yes%**	**Weak No%**	**No%**	**Does not make sense%**	**Insufficient justification%**	**Irrelevant to the schema%**	**Too trivial%**	**None%**
CD–fine-tune	34.7	17.5	20.1	11.6	50.8	32.0	37.0	4.0	7.5	19.5
CD–union	35.9	20.7	15.2	15.2	49.0	33.8	32.4	5.5	6.4	21.9
WT5–fine-tune	8.7	4.6	4.1	4.1	87.2	60.8	20.3	10.6	4.1	4.1
WT5	8.3	4.8	3.0	4.2	87.9	71.1	**12.8**	9.6	**2.1**	4.3
M1	28.3	14.3	14.3	13.6	57.8	**28.0**	39.5	8.9	4.5	19.1
M2	**44.1**	**25.9**	18.0	18.5	**37.6**	**28.1**	33.2	6.5	4.0	**28.1**
M3	29.6	15.4	14.8	13.0	56.8	43.7	29.3	6.9	2.3	17.8
M4	41.9	22.6	22.6	12.8	42.1	34.3	33.3	**2.5**	6.9	23.0
ComVE Model	**NLE score**	**Yes%**	**Weak Yes%**	**Weak No%**	**No%**	**Does not make sense%**	**Insufficient justification%**	**Irrelevant to the schema%**	**Too trivial%**	**None%**
CD–fine-tune	31.4	25.4	7.2	3.8	63.6	26.9	32.3	12.5	3.6	24.7
CD–union	27.7	23.6	4.2	3.8	68.4	39.8	24.6	10.2	**2.7**	22.7
WT5–fine-tune	28.9	20.0	11.8	3.1	65.1	30.7	37.9	**8.9**	3.6	18.9
WT5	23.9	15.3	10.2	5.6	69.0	36.9	31.7	11.9	5.2	14.3
M1	40.2	28.5	14.6	6.1	50.8	22.1	29.0	18.1	4.7	26.1
M2	40.6	27.4	17.7	4.2	50.6	23.9	33.5	10.4	4.4	27.9
M3	38.6	30.3	8.8	7.5	53.5	32.5	**21.7**	12.0	4.4	29.3
M4	**48.5**	**36.7**	14.3	6.8	**42.2**	**18.8**	28.2	13.1	2.9	**37.1**

Table 5: Human annotations of the correctly-classified NLEs generated by models with 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. The last five columns present the shortcomings provided by the human annotators. 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.

WinoGrande 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**
CD–fine-tune	**Prediction:** red. **NLE:** The blue car is expensive and the blue car is more expensive.
CD–union	**Prediction:** red. **NLE:** The blue car is more expensive than the red car, hence it is more expensive to drive.
WT5–fine-tune	**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.
M1	**Prediction:** red. **NLE:** The car was a pleasure to drive but the blue car was not as the blue car was very expensive.
M2	**Prediction:** red. **NLE:** The red car is more expensive than the blue car.
M3	**Prediction:** blue. **NLE:** The blue car is a pleasure to drive and the red car is very expensive.
M4	**Prediction:** blue. **NLE:** The car is a pleasure to drive but the red car is red.

WinoGrande schema: Logan ended the one-side relationship they were having with Robert, which made ___ feel relieved. **Options:** Logan, Robert. **Ground-truth:** Logan.	
Model	**Output**
CD–fine-tune	**Prediction:** Robert. **NLE:** Logan ended the one-side relationship they were having with Robert, which made Logan feel relieved.
CD–union	**Prediction:** Logan. **NLE:** Robert has a one-sided relationship with Logan, which makes him feel relieved.
WT5–fine-tune	**Prediction:** Logan. **NLE:** The Logan
WT5	**Prediction:** Logan. **NLE:** Robert
M1	**Prediction:** Robert. **NLE:** Logan ended the one-side relationship they were having with Robert, which made Logan feel relieved
M2	**Prediction:** Logan. **NLE:** The fact that Logan ended the one-sided relationship with Robert makes him feel relieved.
M3	**Prediction:** Robert. **NLE:** Logan's feelings are relieved because he ended the relationship with Robert.
M4	**Prediction:** Logan. **NLE:** Logan ending a one-side relationship makes Logan feel relieved.

WinoGrande 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**
CD–fine-tune	**Prediction:** Brett. **NLE:** Fiction books were interesting and easy to read for Logan, but Brett enjoyed real tales more than fiction books.
CD–union	**Prediction:** Brett. **NLE:** Fiction books were interesting and easy to read for Logan, while real tales were more difficult to read for Brett.
WT5–fine-tune	**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 [. . .]
M1	**Prediction:** Brett. **NLE:** Fiction books were interesting and easy to read for Logan but not Brett because Brett enjoyed real tales
M2	**Prediction:** Brett. **NLE:** Fiction books are not real tales.
M3	**Prediction:** Logan. **NLE:** Fiction books are interesting and easy to read for Logan, so Logan is more likely to enjoy real tales than Brett.
M4	**Prediction:** Logan. **NLE:** Fiction books are not real tales.

WinoGrande 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**
CD–fine-tune	**Prediction:** Tanya. **NLE:** Erin ordered a martini when Tanya ordered tonic water with lemon because she was pregnant.
CD–union	**Prediction:** Erin. **NLE:** Erin ordered a martini when Tanya ordered a tonic water with lemon, because Erin was pregnant.
WT5–fine-tune	**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 [. . .]
M1	**Prediction:** Tanya. **NLE:** Erin ordered a martini when Tanya only ordered a tonic water with lemon, because _ was pregnant.
M2	**Prediction:** Tanya. **NLE:** Erin ordering a martini implies Erin ordering a martini, and Tanya ordering a tonic implies Erin is pregnant.
M3	**Prediction:** Tanya. **NLE:** Erin ordered a martini
M4	**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 [. . .]

WinoGrande 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**
CD–fine-tune	**Prediction:** Dennis. **NLE:** The game on television was very important for Ian but not for Dennis because he was into music instead.
CD–union	**Prediction:** Dennis. **NLE:** The game on television was very important for Ian, but not for Dennis, because he was into music instead.
WT5–fine-tune	**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 inste [. . .]
WT5	**Prediction:** Dennis. **NLE:** The Ian 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 Ian. Dennis can't be Dennis. Dennis can't be Ian. Dennis can't be Dennis [. . .]
M1	**Prediction:** Dennis. **NLE:** The game on television was very important for Ian but not Dennis because he was into music instead.
M2	**Prediction:** Dennis. **NLE:** The game on television is more important for Ian than Dennis because Dennis is more likely to be into music instead.
M3	**Prediction:** Dennis. **NLE:** The game on television is very important for Ian
M4	**Prediction:** Dennis. **NLE:** The game on television is very important for Ian, so he is not into music.

WinoGrande 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**
CD–fine-tune	**Prediction:** diner. **NLE:** The diner has more money to spend than the food truck.
CD–union	**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–fine-tune	**Prediction:** diner. **NLE:** food truck
WT5	**Prediction:** food truck. **NLE:** diner
M1	**Prediction:** diner. **NLE:** The diner had more money to spend than the food truck.
M2	**Prediction:** diner. **NLE:** The diner has more money to spend than the food truck.
M3	**Prediction:** diner. **NLE:** The diner has more money to spend than the food truck.
M4	**Prediction:** diner. **NLE:** More money to spend is a rephrasing of more money to spend.

ComVE 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**
CD–fine-tune	**Prediction:** Statement 1. **NLE:** A thumb is not inserted into the car ignition switch.
CD–union	**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–fine-tune	**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
M1	**Prediction:** Statement 1. **NLE:** One cannot insert his thumb into the car ignition switch.
M2	**Prediction:** Statement 1. **NLE:** One cannot insert his thumb into the car ignition switch.
M3	**Prediction:** Statement 1. **NLE:** You can't use your thumb to insert keys into the car ignition switch.
M4	**Prediction:** Statement 1. **NLE:** A thumb is not used to insert keys into a car ignition switch

ComVE 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**
CD–fine-tune	**Prediction:** Statement 2. **NLE:** The library is closed to the public
CD–union	**Prediction:** Statement 2. **NLE:** She was met with an accident and was rushed to the library
WT5–fine-tune	**Prediction:** Statement 2. **NLE:** She was met with an accident and was rushed to the hospital 1 explanation:
WT5	**Prediction:** Statement 2. **NLE:** .
M1	**Prediction:** Statement 2. **NLE:** The library is not the only place where a person can have an accident.
M2	**Prediction:** Statement 2. **NLE:** A library is not a hospital.
M3	**Prediction:** Statement 2. **NLE:** A library is not a hospital.
M4	**Prediction:** Statement 2. **NLE:** A library is not a hospital.

ComVE 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**
CD–fine-tune	**Prediction:** Statement 2. **NLE:** She came off sailing across the pacific ocean
CD–union	**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–fine-tune	**Prediction:** Statement 2. **NLE:** She came off sailing across the pacific ocean Sentence 1:
WT5	**Prediction:** Statement 2. **NLE:** .
M1	**Prediction:** Statement 2. **NLE:** A person can't be sailing and walking at the same time.
M2	**Prediction:** Statement 2. **NLE:** sailing is not the same as walking
M3	**Prediction:** Statement 2. **NLE:** Sailing and walking are two different things.
M4	**Prediction:** Statement 2. **NLE:** sailing is not the same as walking

ComVE 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**
CD–fine-tune	**Prediction:** Statement 2. **NLE:** Leena eats two bananas a day which are good for health.
CD–union	**Prediction:** Statement 2. **NLE:** Bananas are good for health and you can't eat them in the morning.
WT5–fine-tune	**Prediction:** Statement 2. **NLE:** Leena daily eats two bananas which are good for health 1 explanation:
WT5	**Prediction:** Statement 2. **NLE:** .
M1	**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
M2	**Prediction:** Statement 2. **NLE:** The bananas are good for health.
M3	**Prediction:** Statement 2. **NLE:** The bananas are not good for health, they are good for health.
M4	**Prediction:** Statement 2. **NLE:** The bananas are good for health.

ComVE Statement 1: The fire will burn you if you touch it. **Statement 2:** The fire will cool you if you touch it. **Nonsensical statement:** Statement 2.	
Model	**Output**
CD–fine-tune	**Prediction:** Statement 2. **NLE:** The fire will burn you if you touch it.
CD–union	**Prediction:** Statement 2. **NLE:** The fire will burn you if you touch it and will cool you if you touch it
WT5–fine-tune	**Prediction:** Statement 2. **NLE:** The fire will burn you if you touch it.
WT5	**Prediction:** Statement 2. **NLE:** the fire will burn you if you touch it.
M1	**Prediction:** Statement 2. **NLE:** The fire will burn you is not the same as the fire that will cool you.
M2	**Prediction:** Statement 2. **NLE:** The fire will burn you if you touch it
M3	**Prediction:** Statement 1. **NLE:** Fire can't burn you
M4	**Prediction:** Statement 2. **NLE:** The fire will burn you.

ComVE Statement 1: Salsa is for hair **Statement 2:** Brushes are for hair **Nonsensical statement:** Statement 1.	
Model	**Output**
CD–fine-tune	**Prediction:** Statement 1. **NLE:** Salsa is for hair
CD–union	**Prediction:** Statement 1. **NLE:** Salsa is for hair
WT5–fine-tune	**Prediction:** Statement 1. **NLE:** SENTENCE 1:
WT5	**Prediction:** Statement 1. **NLE:** :
M1	**Prediction:** Statement 1. **NLE:** Brushes are for hair is not the same as a Salsa
M2	**Prediction:** Statement 1. **NLE:** Salsa is for hair
M3	**Prediction:** Statement 1. **NLE:** Brushes are for hair is a rephrasing of brushes are for hair.
M4	**Prediction:** Statement 1. **NLE:** Salsa is for hair