Evaluating Prompts Across Multiple Choice Tasks In a Zero-Shot Setting

Gabriel Orlanski
go533@nyu.edu

Abstract

Large language models have shown that impressive zero-shot performance can be achieved through natural language prompts (Radford et al., 2019; Brown et al., 2020; Sanh et al., 2021). Creating an effective prompt, however, requires significant trial and error. That prompts the question: how do the qualities of a prompt affect its performance? To this end, we collect and standardize prompts from a diverse range of tasks for use with tasks they were not designed for. We then evaluate these prompts across fixed multiple choice datasets for a quantitative analysis of how certain attributes of a prompt affect performance. We find that including the choices and using prompts not used during pre-training provide significant improvements. All experiments and code can be found https://github.com/gabeorlanski/zero-shot-cross-task.

1 Introduction

Recent work has shown that using a natural language (NL) prompt with pre-trained language models (LM) significantly improves performance in few-shot and zero-shot settings (Brown et al., 2020; Schick and Schütze, 2021b), to the point where they can be worth 100s of data points (Scao and Rush, 2021). Further, T5 (Raffel et al., 2020) showed that simple prompts and reformulating NLP tasks as text-to-text performs well on a wide range of tasks. Recent models such as FLAN (Wei et al., 2021b) and T0 (Sanh et al., 2021) demonstrate that multi-task training a large LM with prompts results in improved zero-shot performance on a wide range of tasks. However, manually designing a prompt is a non-trivial task due to the trial-and-error nature of the task (Jiang et al., 2020; Shin et al., 2021). Some works focus on “prompt programming”—best practices for designing prompts (Reynolds and McDonell, 2021; Liu et al., 2021a). While others have looked towards continuous “soft-prompts”—random vectors added to the input sequence, which are then fine-tuned (Zhong et al., 2021b; Qin and Eisner, 2021). However, recent work has revealed that these prompts are susceptible to minor perturbations (Mishra et al., 2021).

Motivated by this, we aim to conduct a quantitative analysis of what affects a prompt’s performance. We evaluate T0-3B (Sanh et al., 2021) on generalized prompts from a wide range of tasks with eight datasets to provide a quantitative analysis of how the qualitative aspects of a prompt affect its performance.

We collected 95 prompts across 20 tasks and evaluated each on eight datasets. We find that using a prompt performs better for every evaluation task than not using a prompt. Further, we find that the set of prompts with the highest performance is not the ones designed for the specific task for seven of the eight datasets. In our ablations, we find that adding a small amount of task-specific NL to the generalized prompt increases performance by a median of 4.65%. Finally, we find that prompts with the choices present outperform those that do not and that presenting the options as multiple distinct choices further improves the results. Additionally, the longer a prompt is, the worse it performs.

2 Methodology

Our overall approach is detailed in Figure 1. Given a downstream multiple-choice task $T_o = \{(x_1, y_1), \ldots, (x_n, y_n)\}$, evaluate how well T0 (Sanh et al., 2021) performs when using a prompt $P_d$ designed for a different task $T_d$.

2.1 Fixed Choice Tasks

For the purposes of this paper, we limit the scope of the tasks we look at to be only multiple choice tasks whose choices are constant across all examples. Thus, for
\( \mathcal{T}_o = \{(x_1, y_1), \ldots, (x_n, y_n)\} \), every \( y_i \in C_o \) where \( C_o = \{C_1, \ldots, C_\ell\} \) with lengths \( \ell = \{\ell_1, \ldots, \ell_\ell\} \). To make a prediction for the example point \( x_i \), we follow Holtzman et al. (2021) and use rank scoring: taking the choice with the highest probability as defined by

\[
\arg\max_{c_j \in C_o} \frac{\sum_{k=1}^{\ell_j} \log[P(C_j^k | x_i, c_j^1 \ldots c_j^{k-1})]}{\ell_j}
\]

However, the lengths in \( \ell \) are not guaranteed to be the same and thus Equation 1 will unintentionally penalize longer choices (Brown et al., 2020; Holtzman et al., 2021). We thus follow prior work and take the choice with the highest Average Log-Likelihood

\[
\arg\max_{c_j \in C_o} \prod_{k=1}^{\ell_j} P(C_j^k | x_i, c_j^1 \ldots c_j^{k-1})
\]

### 2.2 Generalized Prompts

We use the PromptSource\(^1\) framework proposed by Sanh et al. (2021) for templates as it provides a standardized format for managing prompts. We standardize the input fields and answer formats across all tasks such that they fell into three general categories: CLASSIFICATION, ENTAILMENT, and QUESTIONANSWERING (QA). Table 1 displays an example of what the generalization would look like for an ENTAILMENT prompt. To use a prompt with a task that does not have the same number of inputs, we add additional task specific NL to better align the inputs. To use the example prompt from Table 1 with a sentiment classification task, we would map the input text from the task to the premise field and pass "what is the sentiment" as the hypothesis. In prompts where answers choices are present, we replace them with an additional input field for the choice string (i.e. "yes, no, or maybe") to hold how the choices are presented constant across all prompts. A detailed breakdown of the tasks used for the generalized prompts can be found in Table 3.

### 3 Experimental Setup

#### 3.1 Datasets

For all datasets, we use the most recent version on HuggingFace (Wolf et al., 2020). Every evaluation is done using the validation split as per Sanh et al. (2021). For the evaluation datasets, we again follow Sanh et al. (2021) and use: Adversarial NLI (ANLI) (Nie et al., 2020), CommitmentBank (CB) (DE Marneff et al., 2019), Recognizing Textual Entailment (RTE) (Dagan et al., 2005; Bar-Haim et al., 2006; Giampiccolo et al., 2007; Bentivogli et al., 2009), and Words In Context (WiC) (Pilehvar and osé Camacho-Collados, 2018).

Two additional tasks are used to evaluate T0’s performance on complex tasks in unseen domains: Algebra Question Answering (AQuA) (Ling et al., 2017) and CraigslistBargains (He et al., 2018). Descriptions for these two tasks can be found in Appendix B.

For the generalized prompts, we collect 86 prompts across 19 distinct tasks. In addition to these, we also include 12 prompts with no additional NL for each of the three categories for 4 different ablations. More details can be found in Appendix A.

### 3.2 Model and Metrics

We evaluate the performance of the 3B parameter T0, and T5 (Raffel et al., 2020) models with the HuggingFace implementation (Wolf et al., 2020) as we were limited to a single RTX Titan 24GB card. Following prior works (Sanh et al., 2021; Brown et al., 2020; Holtzman et al., 2021), we report the accuracy and F1 scores on each of the eight datasets.

As each task will have different mean metrics, we cannot compare the raw accuracy and F1 scores across tasks. For a given prompt, we calculate the median accuracy and F1 ranks compared to the 95 other prompts for all evaluation tasks. As the rank is ascending, lower values for median accuracy rank (MAR) and median F1 rank (MFR) indicate a better performing prompt.

### 4 Results

#### 4.1 Baselines On New Tasks

As we evaluate T0-3B on two new tasks, we first want to gauge how the model performs as shown in both Figure 2 and Table 4. We follow Sanh et al. (2021) in using rank scoring without length normalization as defined by Equation 1 and find that for both AQuA and CraigslistBargains, the base T5 model performs better than T0. However, T0’s unweighted multi-class F1 is better than that of T5 on both tasks, indicating that T5 achieves a higher score due to only predicting a subset of the choices. Figure 3 provides further evidence for this hypothesis. In both AQuA and CraigslistBargains, T0 more evenly distributes its predictions across the possible choices where as T5 heavily favors a subset of the choices. Thus, the disparity in the accuracy is likely a result of class imbalance in the evaluation datasets in which T5 is ‘lucky’ in heavily predicting the class that was more populous. We consider this to be ‘luck’ as T5 outperforming T0 only occurs in only two of the datasets we examined.

#### 4.2 Generalized Prompts

We report the results of the cross-task evaluation in Table 2. We find that the only task in which the original\(^2\) prompt performs best is ANLI R2 with an accuracy of 34.70. Conversely, the worst performing prompts for the AQuA task were its original prompts with an accuracy of 17.32. Furthermore, we find that there is no task out of the eight used for evaluation where not using a prompt has the best performance.

\(^1\)https://github.com/bigscience-workshop/promptsourc

\(^2\)By original we are referring to the prompts specifically designed for a task.
We report how the added NL discussed in subsection 2.2 affects the zero-shot performance in Table 5. Across the eight evaluation tasks, adding some task specific text leads to an average increase of 4.65% in the accuracy and a 2.34% increase to the unweighted multiclass F1. However, there was also a decrease of 4.21% and 13.90% to minimum accuracy and unweighted multiclass F1 respectively. This implies that the added extra text helps to better amplify the negative and positive elements of a prompt.

### 4.3 Qualitative Analysis of Prompts

Table 5 displays the rank statistics across multiple ablations and Table 7 displays the correlations of a prompt’s qualities with their rank. In prompts which have choices, the MAR is 33.12 compared to 52.25 when the choices are left out. However, the range as indicated by the Q1 and Q3 MARs is significantly larger when the choices are included, implying that adding the choice string causes high variance.

We also find that when the choices are presented as multiple distinct choices the MAR is further improved to 22.25. In comparison, the prompts with choices that are not in this format have a MAR of 36.00. These results provide further evidence to the findings from Wei et al. (2021a) that clearly distinguishing the options in a prompt improves performance.

Next, we find that the median rank of prompts used in training is 50.25 compared to 42.00 of the unseen prompts. The F1 scores display a similar pattern with training prompts having a MFR of 55.75 while unseen prompts have a median of 36.50. Although this implies that prompts that share tokens with those used for training will perform worse, we find that there is no significant correlation between the number of tokens a prompt \( P \) shares with those used in training and its rank.

Finally, we find that longer prompts have a slightly negative impact on the performance of a prompt. Figure 4 shows that, with the 95 prompts used across eight tasks, the best performing prompts are those whose length is in the range \([14, 21]\) as their MAR is 28.50 and MFR is 36.50. The Q1 values are 18.00 and 15.38, respectively. In comparison, we find that prompts with lengths with lengths \( \geq 25 \) have a median MAR of 50.25 and MFR of 72.00, indicating a negative impact on per-
performance. Surprisingly, we find that prompts whose lengths are < 14 have a median MAR of 47.75 and MFR of 49.00. While this is a negative impact on performance, it is not as large as that in longer prompts.

5 Related Works

Pre-trained Language Models In the past few years, large pre-trained models have risen to prominence due to their strong performance on a wide range of NLP tasks (Radford et al., 2019; Brown et al., 2020; Lewis et al., 2020). In particular, T5 (Raffel et al., 2020) explored transferred learning for large LMs by transforming all NLP problems to a text-to-text format. Beyond natural language, these models have also excelled at tasks in other modalities such as code generation (Austin et al., 2021; Chen et al., 2021; Orlando and Gittens, 2021), in-context learning (Min et al., 2021; Zhong et al., 2021a), and semantic parsing (Rongali et al., 2020). One aspect of large LMs is that they perform well in zero-shot settings (Radford et al., 2019; Brown et al., 2020; Vu et al., 2020).

NL Prompting A drawback of these large LMs is that their size makes it costly to fine-tune them. This lead to the rise of the “pre-train, prompt, and predict” paradigm in which a downstream task is modified to resemble those used in training through the use of NL prompts (Liu et al., 2021b). These prompts have improved few-shot and zero-shot performance across a vast number of models and tasks (Brown et al., 2020; Schick and Schütze, 2021b,a; Mishra et al., 2021; Scao and Rush, 2021; Shin et al., 2020). Recent models such as FLAN (Wei et al., 2021b) and T0 (Sanh et al., 2021) have shown that even better zero-shot performance can be achieved through using a multi-task pre-training objectives with a diverse set of prompts.

6 Conclusion

In this paper, we examined T0’s performance on a range of fixed multiple-choice tasks. We find that T0 does worse than T5 on two unseen complex tasks. Next, we evaluated how the performance of a prompt transfers between tasks. Our results show that using a prompt performs consistently better than not using any prompt. We conclude with a quantitative analysis of what aspects were. Finally, we find that prompts whose length are between 14 and 24 perform better than both longer and shorter prompts. Further work should examine prompt transfer with larger models while also expanding the number of prompts and tasks used.

References

Aida Amini, Saadia Gabriel, Shanchuan Lin, Rik Koncel-Kedziorski, Yejin Choi, and Hannaneh Hajishirzi. 2019. MathQA: Towards interpretable math word problem solving with operation-based formalisms. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pages 2357–2367, Minneapolis, Minnesota. Association for Computational Linguistics.

Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie J. Cai, Michael Terry, Quoc V. Le, and Charles Sutton. 2021. Program synthesis with large language models. ArXiv, abs/2108.07732.

Roy Bar-Haim, Ido Dagan, Bill Dolan, Lisa Ferro, Danilo Giampiccolo, Bernardo Magnini, and Idan Szpektor. 2006. The second pascal recognising textual entailment challenge. In Proceedings of the second PASCAL challenges workshop on recognising textual entailment, volume 6, pages 6–4. Venice.

Max Bartolo, Alastair Roberts, Johannes Welbl, Sebastian Riedel, and Pontus Stenetorp. 2020. Beat the ai: Investigating adversarial human annotation for reading comprehension. Transactions of the Association for Computational Linguistics, 8:662–678.

Luisa Bentivogli, Peter Clark, Ido Dagan, and Danilo Giampiccolo. 2009. The fifth pascal recognizing textual entailment challenge. In TAC.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde, Jared Kaplan, Harrison Edwards, Yura Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder, Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mohammad Bavarian, Clemens Winter, Philippe Tillet, Felipe Petroski Such, David W. Cummings, Matthias Plappert, Fotios Chantziis, Elizabeth Barnes, Ariel Herbert-Voss, William H. Guss, Alex Nichol, Igor Babuschkin, S. Arun Balaji, Shantanu Jain, Andrew Carr, Jan Leike, Joshua Achiam, Vedant Misra, Evan Morikawa, Alec Radford, Matthew M. Knight, Miles Brundage, Mira Murati, Katie Mayer, Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba. 2021. Evaluating large language models trained on code. ArXiv, abs/2107.03374.
Bill Yuchen Lin, Wangchunshu Zhou, Ming Shen, Pei Zhou, Chandra Bhagavatula, Yejin Choi, and Xiang Ren. 2020b. CommonGen: A constrained text generation challenge for generative commonsense reasoning. In Findings of the Association for Computational Linguistics: EMNLP 2020, pages 1823–1840, Online. Association for Computational Linguistics.

Wang Ling, Dani Yogatama, Chris Dyer, and Phil Blunsom. 2017. Program induction by rationale generation: Learning to solve and explain algebraic word problems. ACL.

Jiachang Liu, Dinghan Shen, Yizhe Zhang, Bill Dolan, Lawrence Carin, and Weizhu Chen. 2021a. What makes good in-context examples for gpt-3?

Pengfei Liu, Weiiche Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2021b. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. ArXiv preprint arXiv:2010.14235.

Yao Lu, Yue Dong, and Laurent Charlin. 2020. Multixscience: A large-scale dataset for extreme multi-document summarization of scientific articles. arXiv.

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, pages 142–150, Portland, Oregon, USA. Association for Computational Linguistics.

P. Malo, A. Sinha, P. Korhonen, J. Wallenius, and P. Takala. 2014. Good debt or bad debt: Detecting semantic orientations in economic texts. Journal of the Association for Information Science and Technology, 65.

Sewon Min, Mike Lewis, Luke Zettlemoyer, and Hannah Hajishirzi. 2021. Metaicl: Learning to learn in context. ArXiv, abs/2110.15943.

Swaroop Mishra, Daniel Khashabi, Chitta Baral, and Hanna Hajishirzi. 2021. Cross-task generalization via natural language crowdsourcing instructions.

Shashi Narayan, Shay B. Cohen, and Mirella Lapata. 2018. Don’t give me the details, just the summary! topica-convolu-tional neural networks for extreme summarization. ArXiv, abs/1808.08745.

Yixin Nie, Adina Williams, Emily Dinan, Mohit Bansal, Jason Weston, and Douwe Kiela. 2020. Adversarial nli: A new benchmark for natural language understanding. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics.

Gabriel Orlanski and Alex Gittens. 2021. Reading StackOverflow encourages cheating: Adding question text improves extractive code generation. In Proceedings of the 1st Workshop on Natural

Bill Yuchen Lin, Seyeon Lee, Rahul Khanna, and Xiang Ren. 2020a. Birds have four legs?! numersense: Probing numerical commonsense knowledge of pretrained language models. In Proceedings of EMNLP. To appear.

He He, Derek Chen, Anusha Balakrishnan, and Percy Liang. 2018. Decoupling strategy and generation in negotiation dialogues. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 2333–2343, Brussels, Belgium. Association for Computational Linguistics.

Giovanni Grano, Andrea Di Sorbo, Francesco Mercaldo, Corrado A. Visaggio, Gerardo Canfora, and Sebastiano Panichella. 2017. Android apps and user feedback: A dataset for software evolution and quality improvement. In Proceedings of the 2nd ACM SIGSOFT International Workshop on App Market Analytics, WAMA 2017, pages 8–11, New York, NY, USA. Association for Computing Machinery.

Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Dan Huang, Andrew Y. Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, pages 142–150, Portland, Oregon, USA. Association for Computational Linguistics.

Iris Hendrickx, Su Nam Kim, Zornitsa Kozareva, Preslav Nakov, Diarmuid O’Searghda, Sebastian Padó, Marco Pennacchiotti, Lorenza Romano, and Stan Szpakowicz. 2010. SemEval-2010 task 8: Multi-way classification of semantic relations between pairs of nominals. In Proceedings of the 5th International Workshop on Semantic Evaluation, pages 33–38, Uppsala, Sweden. Association for Computational Linguistics.

Ari Holtzman, Peter West, Vered Schwartz, Yejin Choi, and Luke Zettlemoyer. 2021. Surface form competition: Why the highest probability answer isn’t always right. In EMNLP.

Jiachang Liu, Dinghan Shen, Yizhe Zhang, Bill Dolan, Lawrence Carin, and Weizhu Chen. 2021a. What makes good in-context examples for gpt-3?

Pengfei Liu, Weiiche Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2021b. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. ArXiv preprint arXiv:2010.14235.

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, pages 142–150, Portland, Oregon, USA. Association for Computational Linguistics.

P. Malo, A. Sinha, P. Korhonen, J. Wallenius, and P. Takala. 2014. Good debt or bad debt: Detecting semantic orientations in economic texts. Journal of the Association for Information Science and Technology, 65.

Sewon Min, Mike Lewis, Luke Zettlemoyer, and Hannah Hajishirzi. 2021. Metaicl: Learning to learn in context. ArXiv, abs/2110.15943.

Swaroop Mishra, Daniel Khashabi, Chitta Baral, and Hanna Hajishirzi. 2021. Cross-task generalization via natural language crowdsourcing instructions.

Shashi Narayan, Shay B. Cohen, and Mirella Lapata. 2018. Don’t give me the details, just the summary! topica-convolu-tional neural networks for extreme summarization. ArXiv, abs/1808.08745.

Yixin Nie, Adina Williams, Emily Dinan, Mohit Bansal, Jason Weston, and Douwe Kiela. 2020. Adversarial nli: A new benchmark for natural language understanding. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics.

Gabriel Orlanski and Alex Gittens. 2021. Reading StackOverflow encourages cheating: Adding question text improves extractive code generation. In Proceedings of the 1st Workshop on Natural
Language Processing for Programming (NLP4Prog 2021), pages 65–76, Online. Association for Computational Linguistics.

Denis Paperno, Germán Kruszewski, Angeliki Lazari-dou, Ngoc Quan Pham, Raffaella Bernardi, Sandro Pezzelle, Marco Baroni, Gemma Boleda, and Raquel Fernandez. 2016. The LAMBADA dataset: Word prediction requiring a broad discourse context. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 1525–1534, Berlin, Germany. Association for Computational Linguistics.

Mohammad Taher Pilehvar and osé Camacho-Collados. 2018. Wic: 10, 000 example pairs for evaluating context-sensitive representations. CoRR, abs/1808.09121.

Guanghui Qin and Jason Eisner. 2021. Learning how to ask: Querying lms with mixtures of soft prompts.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

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.

Laria Reynolds and Kyle McDonell. 2021. Prompt programming for large language models: Beyond the few-shot paradigm.

Melissa Roemmele, Cosmin Adrian Bejan, and Andrew S Gordon. 2011. Choice of plausible alternatives: An evaluation of commonsense causal reasoning. In 2011 AAAI Spring Symposium Series.

Subendhu Rongali, Luca Soldaini, Emilio Monti, and Wael Hamza. 2020. Don’t parse, generate! a sequence to sequence architecture for task-oriented semantic parsing. Proceedings of The Web Conference 2020.

Victor Sanh, Albert Webson, Colin Raffel, Stephen H. Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Teven Le Scao, Arun Raja, Manan Dey, M Saiful Bari, Canwen Xu, Urnish Thakker, Shanya Sharma Sharma, Eliza Szczesha, Taewoon Kim, Gujun Chhablani, Nihal Nayak, Debajyoti Datta, Jonathan Chang, Mike Tian-Jian Jiang, Han Wang, Matteo Manica, Sheng Shen, Zheng Xin Yong, Harshit Pandey, Rachel Bawden, Thomas Wang, Trishala Neeraj, Jos Rozen, Abheesht Sharma, Andrea Santilli, Thibault Fevry, Jason Alan Fries, Ryan Teehan, Stella Biderman, Leo Gao, Tali Bers, Thomas Wolf, and Alexander M. Rush. 2021. Multi-task prompted training enables zero-shot task generalization.

Teven Le Scao and Alexander M. Rush. 2021. How many data points is a prompt worth? In NAACL.

Timo Schick and Hinrich Schütze. 2021a. Exploiting cloze-questions for few-shot text classification and natural language inference. In Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, pages 255–269, Online. Association for Computational Linguistics.

Timo Schick and Hinrich Schütze. 2021b. It’s not just size that matters: Small language models are also few-shot learners. In Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 2339–2352, Online. Association for Computational Linguistics.

Richard Shin, Christopher H. Lin, Sam Thomson, Charles Chen, Subhro Roy, Emmanouil Antonios Platanios, Adam Pauls, Dan Klein, Jason Eisner, and Benjamin Van Durme. 2021. Constrained language models yield few-shot semantic parsers.

Taylor Shin, Yasaman Razeghi, Robert L. Logan IV, Eric Wallace, and Sameer Singh. 2020. AutoPrompt: Eliciting Knowledge from Language Models with Automatically Generated Prompts. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 4222–4235, Online. Association for Computational Linguistics.

Tu Vu, Tong Wang, Tsenduren Munkhdalai, Alessandro Sordoni, Adam Trischler, Andrew Mattarella-Micke, Subhransu Maji, and Mohit Iyyer. 2020. Exploring and predicting transferability across NLP tasks. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 7882–7926, Online. Association for Computational Linguistics.

Jason Wei, Maarten Bosma, Vincent Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V. Le. 2021a. Finetuned language models are zero-shot learners. ArXiv, abs/2109.01652.

Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. 2021b. Finetuned language models are zero-shot learners. arXiv preprint arXiv:2109.01652.

Orion Weller, Nicholas Lourie, Matt Gardner, and Matthew E. Peters. 2020. Learning from task descriptions. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 1361–1375, Online. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierre Cistac, Tim Rault, Remi Louf, Morgan Fontowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing.
Ruiqi Zhong, Kristy Lee, Zheng Zhang, and Dan Klein. 2021a. Adapting language models for zero-shot learning by meta-tuning on dataset and prompt collections. In EMNLP.

Zexuan Zhong, Dan Friedman, and Danqi Chen. 2021b. Factual probing is [mask]: Learning vs. learning to recall.

A Generalized Prompts

The tasks we took prompts from that were not used to train T0 are:

- COPA (Roemmele et al., 2011)
- FinancialNews (Malo et al., 2014)
- LAMBADA (Paperno et al., 2016)
- MathQA (Amini et al., 2019)
- MultiXSci (Lu et al., 2020)
- NumerSense (Lin et al., 2020a)
- SemEval2010 (Hendrickx et al., 2010)
- ZEST (Weller et al., 2020)

The tasks we took prompts from that were used to train T0 are:

- AppReviews (Grano et al., 2017)
- Adversarial QA (Bartolo et al., 2020)
- CommonGen (Lin et al., 2020b)
- IMDB (Maas et al., 2011)
- XSum (Narayan et al., 2018)

| Task Name       | #  | Type         | MCQ |
|-----------------|----|--------------|-----|
| Text            | 6  | All Three   | ✓   |
| Text+Choices    | 6  | All Three   | ✓   |

Table 3: Number of prompts used by task for the generalized prompts. # is the number of prompts used from this task. MCQ indicates if the task had prompts that are formatted as a multiple choice question with choice letters.

B Complex Task Datasets

The two complex task used to evaluate T0’s performance are:

Algebra Question Answering (AQuA) Dataset of multiple choice algebraic word problems. The choices for this task are \{A,B,C,D,E\} and each letter maps to a potential mathematical answer (Ling et al., 2017).

CraigslistBargains A collection of dialogues involving two-parties negotiating the price of an item for sale on Craigslist. For the scope of this paper, we use the task of classifying who won the negotiation (He et al., 2018).

C Additional Results

This section is for results that could not be included in the main body of the paper due to the page limits.
Figure 2: Median Accuracy with interquartile range for three models: GPT-3, T5, and T0. Darker indicates larger model. Results for GPT-3 Model are from Brown et al. (2020). Results for the 11B T0 and T5 models are taken from Sanh et al. (2021).

| Task            | Model | Accuracy | F1    |
|-----------------|-------|----------|-------|
| ANLI R1         | T0    | 34.30    | 26.17 |
|                 | T5    | 33.40    | 20.28 |
| ANLI R2         | T0    | 33.40    | 23.70 |
|                 | T5    | 33.50    | 21.25 |
| ANLI R3         | T0    | 33.42    | 21.82 |
|                 | T5    | 33.33    | 24.84 |
| AQuA            | T0    | 18.90    | 15.04 |
|                 | T5    | 24.41    | 12.74 |
| CB              | T0    | 55.36    | 38.62 |
|                 | T5    | 21.43    | 19.41 |
| CraigslistBargains | T0  | 33.42    | 18.45 |
|                 | T5    | 44.89    | 16.47 |
| RTE             | T0    | 59.21    | 69.56 |
|                 | T5    | 52.89    | 36.08 |
| WiC             | T0    | 50.86    | 8.81  |
|                 | T5    | 50.16    | 5.44  |

Table 4: Median accuracy and F1 for the corresponding Figure 2.

|                  | Accuracy | F1     |
|------------------|----------|--------|
| Has Choices      | -0.14    | -0.27  |
| Is MCQ           | -0.16    | -0.25  |
| Training Prompt  | 0.07     | 0.13   |
| Length           | 0.14     | 0.18   |

Table 7: Correlations with metric rank for a given prompt quality. Per the definition of rank, a lower score is better and therefore a negative correlation indicates a quality improves performance. Length is measured as the raw number of tokens in a prompt.

Figure 3: Distribution of choices for T0 and T5 on the AQuA and CraigslistBargains.
### Table 5: Accuracy and F1 ranks for different ablations. It is calculated by taking the median rank of a given prompt across all 8 tasks then taking the Mean, Median, Q1, and Q3 of that. Lower is better. Q1 and Q3 are the first and third quartile.

| Ablation               | Accuracy | F1       |
|------------------------|----------|----------|
|                        | Mean     | Median   | Q1   | Q3   | Mean     | Median   | Q1   | Q3   |
| Training Prompts       | 51.46    | 50.25    | 45.25 | 63.00| 54.18    | 55.75    | 38.00| 66.00|
| Unseen Prompts         | 42.72    | 42.00    | 23.50 | 60.75| 42.46    | 36.50    | 22.00| 62.50|
| With Choices           | 39.44    | 33.12    | 20.19 | 58.62| 39.37    | 31.00    | 19.12| 61.75|
| No Choices             | 51.73    | 52.25    | 44.75 | 60.50| 55.93    | 53.50    | 43.00| 66.00|
| Is MCQ                 | 25.80    | 22.25    | 16.50 | 26.00| 23.14    | 16.50    | 13.75| 25.25|
| Not MCQ                | 43.28    | 36.00    | 26.38 | 62.62| 43.95    | 36.50    | 22.00| 65.50|
| Extra Text             | 44.99    | 46.75    | 28.81 | 60.31| 46.44    | 46.50    | 27.75| 66.00|
| No Extra Text          | 44.41    | 48.00    | 32.75 | 57.62| 45.43    | 44.50    | 26.62| 62.25|

Table 6: Median Accuracy when using modified prompts for cross task zero-shot evaluation. **Bolded** entries are prompts for the original task. *Green Cells* and *Red Cells* are the best and worst performing tasks for a column respectively. Rank is the median rank of prompts from this task out of 95 total prompts. ANLI and CB both use the same prompts for their original task prompts per PromptSource.
Figure 4: Accuracy and F1 rank compared with the number of tokens in the prompt. The tick value is the lower bound of the range. p=The number of prompts that fall into that respective range.