Continuation is a Sub-Task of Fill in the Blank: Why Not Train for Both?

Anonymous ACL-IJCNLP submission

Abstract

The task of inserting text into a specified position in a passage, known as fill in the blank, is useful for a variety of applications where writers interact with a natural language generation (NLG) system to craft text. However, NLG research has mostly focused on continuation models that append text to the end of a passage. Since continuation is in fact a subtask of fill in the blank, one where the blank is placed at the sequence's end, we propose the training of a single model which can effectively handle both these tasks. The result is improved efficiency—as only one model needs to be maintained—without any negative impact on performance at either task.

1 Introduction

Natural language generation systems are increasingly being incorporated into applications where a human writer and an AI jointly collaborate to construct text. These range from creative domains such as collaborative story writing (Coenen et al., 2021; Akoury et al., 2020) to more practical ones such as email composition (Buschek et al., 2021; Wu, 2018). Currently, these applications are mostly limited to proposing continuations that come at the end of what has been written so far. This is because because both historical language models (LMs) and state-of-the-art neural LMs have typically been designed to generate text by repeatedly predicting the next word in a sequence given the previous words. However, a more powerful interactive tool would enable writers to insert text into any arbitrary position within the existing text. This task is known as infilling or fill in the blank.

Filling in the blank (FitB) is actually a more general task than continuation. Any model that can do FitB can be made to do continuation by placing the blank at the end of the input. In this paper we make a case for training models that prioritize performance on FitB. We show through human evaluation that training for FitB does not harm performance on continuation. In addition, having a single model that can handle a variety of tasks is better for downstream applications where maintaining multiple neural networks can be prohibitive.

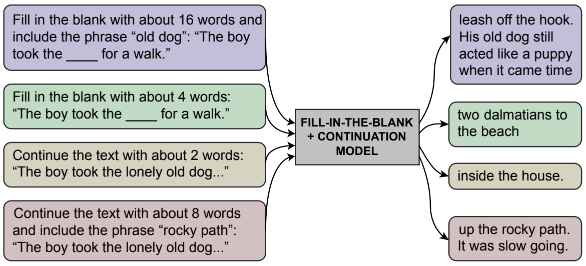

Fill in the blank with about 16 words and include the phrase "old dog": "The boy took the ____ for a walk."
Fill in the blank with about 4 words: "The boy took the ____ for a walk."
Continue the text with about 2 words: "The boy took the lonely old dog..."
Continue the text with about 8 words and include the phrase "rocky path": "The boy took the lonely old dog..."
FILL-IN-THE-BLANK + CONTINUATION MODEL
leash off the hook. His old dog still acted like a puppy when it came time
two dalmatians to the beach
inside the house.
up the rocky path. It was slow going.

Figure 1: A single model that can handle a variety of related writing tasks is more efficient than separate models per task.

In particular, we finetune T5, an encoder-decoder model whose pre-training objective resembles FitB (Raffel et al., 2019). During finetuning, we add two extra conditioning signals likely to be useful in interactive settings: the desired length of the generated text and a goal subsequence that should be included in it. We compare our approach against few-shot learning (Brown et al., 2020), a method by which a sufficiently large language model trained to do continuation is made to perform other tasks through manual construction of a prompt containing several demonstrations of the task. We find that few-shot learning via prompt engineering may not be well-suited to the FitB task.

2 Related Work

FitB is a form of Cloze task (Taylor, 1953). Prior deep-learning approaches to this task include training a Transformer model from scratch to predict the blank text given the context, a target length bucket, and a list of tokens which should be included in the prediction (Ippolito et al., 2019), as well as

Example Type	Input	Target
C4FillBlank no goal	fill: I love avocados. I ate a sandwich covered in them. _8_ I talked to my doctor about it later. It turned out I was allergic to avocados.	After I ate it, my mouth was itchy and tingly.
C4FillBlank with goal	fill: I love avocados. I ate a sandwich covered in them. _8_ I talked to my doctor about it later. It turned out I was allergic to avocados. **Goal: mouth was itchy**	After I ate it, my mouth was itchy and tingly.
C4FillBlank no goal	fill: I love avocados. I ate a sandwich covered in them. After I ate it, my mouth was itchy and tingly. I talked to my doctor about it later. _8_	It turned out I was allergic to avocados.
C4FillEnd with goal	fill: I love avocados. I ate a sandwich covered in them. After I ate it, my mouth was itchy and tingly. I talked to my doctor about it later. _8_ **Goal: allergic to**	It turned out I was allergic to avocados.

Table 1: Example finetuning objectives. The “8” between the two underscores is the approximate length in words of the target sequence. During finetuning, half of examples contained a “goal” subsequence and half did not.

finetuning GPT-2 (Radford et al., 2019) to perform FitB (Donahue et al., 2020). Our FitB training objective is similar to the “infilling by language modeling” objective described in Donahue et al. (2020), except since we use an encoder-decoder model instead of a decoder-only model, the attention layers encoding the context in our approach support attending to future tokens positions, not just prior ones. Related to FitB, Mori et al. (2020) investigate a setting where a sentence is randomly deleted from the input, and the model must both predict the location of the deletion as well as its contents. In addition, Huang et al. (2020) tackle the sentence infilling task using a combination of BERT to encode the context sentences and GPT-2 to generate the missing sentence given the context's BERT embeddings. Lastly, many LM pre-training objectives involve masking out parts of the input then predicting the masked values, which is similar to FitB (Devlin et al., 2018; Raffel et al., 2019).

3 Method

Training Language models trained to do continuation are typically decoder-only; i.e. they are trained to predict the next token in a target sequence given the previous tokens, and the Transformer attention mechanism is masked so that token positions can only attend to preceding positions. To support both continuation and FitB, we instead suggest an encoder-decoder model. An input sequence is encoded with an encoder network, and a decoder network predicts the tokens of the target sequence given both the preceding tokens of the target and the encoder's output embeddings (Vaswani et al., 2017). When training for FitB, the input sequence is the context text surrounding the chosen blank concatenated with textual representations of any additional conditioning signals. The target sequence is the true text for the blank. This formulation easily supports continuation by placing the blank at the end (that is, no right context).

We design FitB training examples (Table 1) the following way. Text sequences are drawn from C4, a dataset containing 350M cleaned web documents, then split into words sequences. After filtering to examples between 256-512 words long, a subsequence of between 1 and 64 words is selected to be blanked out; this becomes the target the model is trying to predict. Following Roberts and Raffel (2020), the input to the model is the original text sequence but with the target replaced with “_X_” where X is a discretized version of the target's length in words. For half of training examples, we perform “goal conditioning,” where a random subsequence of up to half the words of the target is appended to the input.

Rather than train from scratch, we finetune pre-trained T5 models. T5 was pre-trained with a span corruption task where the model had to reconstruct the missing text after random sub-sequences of the input were replaced with special identifiers. This objective is reminiscent of FitB. We finetune in one of three settings: either the blank location is sampled uniform randomly across the sequence (FillBlank), the blank is always placed at the end of the sequence (FillEnd), or for half of examples the blank is randomly selected and for the other half it is always at the end (FillBlankOrEnd). For each of FillBlank, FillEnd, and FillBlankOrEnd, we finetune a 770 million parameter “Large” pre-trained T5 model. For FillBlankOrEnd, we additionally finetune the 3B parameter “XL” T5 model (see Appendix for training details).

Evaluation Datasets Though we finetune on C4, Common Crawl dataset is variable in quality (Dodge et al., 2021). Therefore, we additionally evaluate on two story writing datasets as these

XL Model	Context	Length
C4FillBlank	0.867	0.810
RwpFillBlank	0.800	0.830
C4FillEnd	0.864	0.826
RwpFillEnd	0.830	0.820
Large Model	Context	Length
C4FillBlank	0.860	0.877
RwpFillBlank	0.797	0.881
C4FillEnd	0.858	0.775
RwpFillEnd	0.791	0.746

Table 2: Accuracy of models finetuned on FillBlankOrEnd at correctly using provided length and goal conditioning signals.

	FillBlankOrEnd		**FillBlank**	**FillEnd**
	XL	**Large**	**Large**	**Large**
C4FillBlank	9.53	11.79	**11.64**	16.10
RocFillMiddle	5.34	6.43	**6.41**	37.08
RwpFillBlank	13.05	16.15	**16.11**	21.35
RwpFillSent	11.98	**14.84**	14.89	27.73
C4FillEnd	11.15	13.47	13.88	**13.26**
RocFillEnd-T	5.79	**6.73**	6.84	6.79
RocFillEnd-F (↑)	9.58	10.09	10.09	**10.14**
RwpFillEnd	16.57	**19.89**	20.16	19.9

Table 3: The perplexity of finetuned T5 models on each validation set. Except for RocFillEnd-S5-F, lower is better.

more closely match likely use cases for FitB models. Reddit Writing Prompts (Rwp) is a corpus of stories from the 'r/WritingPrompts' sub-Reddit (Fan et al., 2018). ROC Stories (Roc) is a crowdsourced dataset of five-sentence commonsense stories (Mostafazadeh et al., 2016). From the Reddit Writing Prompts validation set, we produce RwpFillBlankand RwpFillEnd, which are processed identically to the C4 training data, as well as RwpFillSent, where gaps are randomly chosen but always exactly one sentence long. For ROC Stories, the 2018 validation set is used to construct RocFillMiddle, where the middle sentence is blanked out, and RocFillEnd, where the last sentence is blanked out. Unless otherwise noted, evaluation is done without goal conditioning.

Baseline Few-shot learning as introduced by Brown et al. (2020) involves constructing a natural language prompt that includes several demonstrations of the desired task. The prompt is passed as input to a large LM which generates a continuation. Choosing appropriate examples for the prompt can be challenging as task performance is often sensitive to minor changes in prompt design (Zhao et al., 2021). We experiment with a GPT-3-sized model and prompts randomly selected from the C4, Reddit Writing Prompts, and ROC Stories training sets, as well as prompts consisting of examples handwritten by the authors with the goal of story-writing in mind. For each prompt source, we randomly generate five possible prompts, each with three examples (more details in Appendix). To simplify the task, we condition on desired length but not goal text.

4 Results

Qualitative examples from our method and baselines can be found at `https://bit.ly/2U0Ixxa`.

Performance on Continuation Table 3 shows perplexity of the target text for each evaluation dataset for each finetuned model. Interestingly, it may not be necessary to explicitly train on fill-in-the-end examples to achieve a model that can do continuation. The FillBlank model, with blanks randomly placed during training, has about as low perplexity as the FillBlankOrEnd and FillEnd models, for which 50% and 100% of blanks were placed at the end respectively. On ROC Stories, we report perplexity for both the true 5th sentence in each story and a semantically similar but incorrect 5th sentence. As expected, the false endings have higher perplexity.

Performance on FitB Both the FillBlankOrEnd and FillEnd models are about equally effective at filling in the blank, while the FillEnd model, unsurprisingly, does not handle randomly-laced blanks well at all (Table 3).

Domain Transfer Through Table 3, we see that despite training exclusively on C4, the models have decent transferability to more targeted domains. For example, the models all have lower perplexity on ROC Stories than they do C4, and all models correctly give the ROC Stories validation set's false story endings higher perplexity overall than its true story ending sentences. For the Reddit Writing Prompts, perplexity is slightly lower when evaluating with sentence level blanks, even though the model was trained with randomly chosen gaps, which suggests that sentence-level FitB is an easier task that can be performed well even by models not explicitly trained for it.

Extra Conditioning Table 2 shows how well the finetuned FillBlankOrEnd models abide by the length and goal text conditioning signals when generating outputs for versions of the eval sets that contain both signals. Surprisingly, for the Fill-

	C4Fill	RocFill	RwpFill	RwpFill
Few-shot source:	Blank	Middle	Blank	Blank-Sent
C4FillBlank	15.67	19.72	**19.65**	**16.82**
RocFillMiddle	**14.14**	19.61	**19.48**	**16.36**
RwpFillBlank	24.39	20.29	32.33	28.13
RwpFillBlank-Sent	18.91	**18.21**	24.44	19.87
FS Custom	17.98	19.80	21.72	18.38
FillBlankOrEnd	10.33	20.47	14.08	10.37

Table 4: Perplexity of evaluation sets according to a standard LM when blank has been filled in with prediction Large model (bottom). For few-shot, perplexities are averaged over 5 prompts, and the best method for each eval set is bolded, as well as other methods within one standard error.

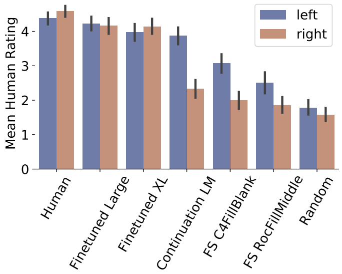


Figure 2: Human ratings of generations on 35 different prompts. Error bars are 95% confidence intervals.

Blank eval sets, the XL model is worse at length conditioning than the Large one. It is slightly better at using the goal text than the Large model. For few-shot learning, the prompt included a desired length but no goal text. Across all tested prefix tuning prompts and all eval sets, only 22.4%. of generations fell into their target length bucket, with the best prompt achieving 30% accuracy.

Generation Perplexity To directly compare the results from few-shot learning with our finetuned models, we use an alternative measurement of perplexity. Using an off-the-shelf LM trained for continuation, we evaluate perplexity of the eval set examples when the predicted text is placed in the blank (Donahue et al., 2020). The results are shown in Table 4. Our finetuning approach outperforms few-shot learning on all eval sets except for ROC Stories. Moreover, we observe high variance in the performance of the different few-shot prompt sources. Surprisingly, choosing few-shot examples from the same data source as an eval dataset did not result in the best performance on that eval set. ROC Stories, with its simplistic 5-sentence examples, tended to make the best few-shot prompts overall. Finally, few-shot learning tended be unreliable; for 4.2% of examples, it was not possible to parse a FitB output from the predicted continuation.

Human Evaluation We conduct human evaluation on 35 examples chosen from RwpFillBlank, with examples about evenly distributed across length buckets. In each annotation task, the rater was presented an input context and several possible sequences that could go in the blank. They were asked to rate each sequence first, on how well it fit the text before it, and second, on how well it fit with the text following it, according to a 5-point slider (more details in the Appendix). The sequences shown included (1) the ground-truth text that was originally in the blank, (2) outputs from FillBlankOrEnd Large; (3) outputs from FillBlankOrEnd XL; (4 and 5) outputs from the best two few-shot prompts; which were sourced from RocFillMiddle and RwpFillBlank; (6) a sequence chosen randomly from all method outputs over all datasets; and (7) the continuation from a C4-trained decoder-only LM with the same number of parameters as FillBlankOrEndthat only had the left context as input.

As shown in Figure 2, the Large and XL models performed about equivalently, which is not too surprising since once models become sufficiently big, it is difficult for human raters to distinguish between them, especially for the relatively short generation lengths we are using here. The LM trained only for continuation was, as expected, about as good as matching with the left context as the FillBlankOrEnd methods, but much worse at matching with the right context. The outputs from few-shot learning were rated to be significantly worse than the finetuned models.

5 Conclusion

In this work, we show that a model trained for fill-in-the-blank is perfectly capable of doing continuation. Additional conditioning signals such as desired length and goal text can be successfully incorporated into fine-tuning in order to support an even greater diversity of model interactions. Multi-task models like the ones we propose require less total training and are more efficient to store and use at inference time. While few-shot learning is a promising method for supporting multi-task inference (and requires no finetuning), it is challenging to make work for the FitB task.

References

Nader Akoury, Shufan Wang, Josh Whiting, Stephen Hood, Nanyun Peng, and Mohit Iyyer. 2020. Storium: A dataset and evaluation platform for machine-in-the-loop story generation. *arXiv preprint arXiv:2010.01717*.

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.

Daniel Buschek, Martin Zürn, and Malin Eiband. 2021. The impact of multiple parallel phrase suggestions on email input and composition behaviour of native and non-native english writers. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, pages 1–13.

Andy Coenen, Luke Davis, Daphne Ippolito, Ann Yuan, and Emily Reif. 2021. Wordcraft: a human-ai collaborative editor for story writing.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Jesse Dodge, Maarten Sap, Ana Marasovic, William Agnew, Gabriel Ilharco, Dirk Groeneveld, and Matt Gardner. 2021. Documenting the english colossal clean crawled corpus. *arXiv preprint arXiv:2104.08758*.

Chris Donahue, Mina Lee, and Percy Liang. 2020. Enabling language models to fill in the blanks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2492–2501.

Angela Fan, Mike Lewis, and Yann Dauphin. 2018. Hierarchical neural story generation. *arXiv preprint arXiv:1805.04833*.

Yichen Huang, Yizhe Zhang, Oussama Elachqar, and Yu Cheng. 2020. Inset: Sentence infilling with inter-sentential transformer. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2502–2515.

Daphne Ippolito, David Grangier, Chris Callison-Burch, and Douglas Eck. 2019. Unsupervised hierarchical story infilling. In *Proceedings of the First Workshop on Narrative Understanding*, pages 37–43.

Xiang Lisa Li and Percy Liang. 2021. Prefix-tuning: Optimizing continuous prompts for generation. *arXiv preprint arXiv:2101.00190*.

Yusuke Mori, Hiroaki Yamane, Yusuke Mukuta, and Tatsuya Harada. 2020. Finding and generating a missing part for story completion. In *Proceedings of the The 4th Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature*, pages 156–166.

Nasrin Mostafazadeh, Nathanael Chambers, Xiaodong He, Devi Parikh, Dhruv Batra, Lucy Vanderwende, Pushmeet Kohli, and James Allen. 2016. A corpus and cloze evaluation for deeper understanding of commonsense stories. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 839–849, San Diego, California. Association for Computational Linguistics.

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

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv preprint arXiv:1910.10683*.

A Roberts and C Raffel. 2020. Exploring transfer learning with t5: The text-to-text transfer transformer. *Google AI Blog*.

Wilson L Taylor. 1953. “cloze procedure”: A new tool for measuring readability. *Journalism Bulletin*, 30(4):415–433.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *arXiv preprint arXiv:1706.03762*.

Yonghui Wu. 2018. Smart compose: Using neural networks to help write emails. *Google AI Blog*.

Tony Z Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. 2021. Calibrate before use: Improving few-shot performance of language models. *arXiv preprint arXiv:2102.09690*.

A Appendix

A.1 Implementation Details

For length conditioning, when discretizing the target sequence's length to a length bucket, we choose the closest value in {1, 2, 4, 8, 16, 32, 64} to the target's length in words.

All training was done in the Mesh Tensorflow T4 codebase.[1] Each T5 model was finetuned for about 50,000 steps with a batch size of 128 examples (i.e., ∼6.4M examples were seen during finetuning.) A constant learning rate of 0.0008 was used, and no overfitting was observed. Code to reproduce our finetuning objectives on arbitrary datasets is included in the downloadable ".zip" and will be made available on Github upon paper acceptance.

All inference is done with random sampling with top-k set to 50 and temperature at 1.0.

A.2 Few-Shot Learning Details

An example prompt is shown in Figure 3. When choosing random few-shot prompts from the dataset train sets, in order to keep the few-shot prompt text within the 512-token context length limit of the 128B parameter model we used for inference, we only considered examples that contained 100 or fewer tokens, so that the max length of the few-shot prompt was no more than 300 tokens. This left 212 tokens for the text of the actual example we were interested in performing the FITB task on. For each evaluation set, examples with inputs longer than 212 tokens were excluded from analysis. For our hand-written prompt, we wrote the 7 examples shown in Table 5. We generated 5 possible prompts by randomly subsampling 3 examples out of these 7.

Our analysis of few-shot learning prompts was not sufficiently exhaustive to rule out the possibility there might exist a prompt for which this technique would be effective. For example, we did not conduct formal experiments to systematically vary the prompt wording/formatting shown in Figure 3. We can conclude that the process of finding an ideal prompt requires time-consuming trial-and-error and is quite difficult!

Finally, even leaving room for 212 tokens worth of context text, some eval examples did not fit in the prompt length, and these examples were skipped when doing the few-shot perplexity analysis for Table 4. All evaluation datasets started with 5,0000 examples. Figure 4 shows a histogram of how many examples remained for each few-shot prompt after the too-long examples were filtered out.

A.3 Human Evaluation

A screenshot of the Human Intelligence Task (HIT) used for annotations is shown in Figure 5. Workers were paid originally paid $1.85 per HIT, but since the average HIT duration ended up being 15 minutes, we awarded each rater a bonus to raise their pay to an average of $10 per hour.

Each example was shown to three raters, and annotations were rejected if the rater gave a lower overall score to the random output than to the ground-truth one. A total of 3 annotations were rejected. Overall, the Fleiss' kappa agreement of pairs of annotators giving the same numerical score to the same question was 0.26.

A.4 Experiments with Prefix Tuning

During the course of this study, we experimented with the usage of Prefix Tuning (Li and Liang, 2021) for the FITB task. In this method, a fixed-length continuous space prefix is appended to the input sequences and this prefix is directly optimized to maximize performance on a given task. This can be used to estimate an upper bound for the performance of few-shot learning on a given task. We trained two prefixes, both of length 5, on pre-trained GPT-2 of size medium (345M) and large (774M) (Radford et al., 2019). While our results showed that the prefix successfully instructed the pre-trained model to perform the FITB task, neither of these models outperformed our few-shot prompts during Human Evaluation, showing only marginally better performance than our random baseline. Due to the discrepancy in size between the prefix tuned GPT-2 models and the models we tested for few-shot prompting, we left these results out of the final analysis. Future work should seek to explore the limitations of this technique and the ways in which it and few-shot learning can be compared.

[1] https://github.com/google-research/text-to-text-transfer-transformer

Context	Taget
An elderly man was sitting alone on a dark path. The man looked down at his feet, and realized ____ . It was a plain pine box and looked as if it had been there for a long time. The man was afraid to look inside the box.	he was holding a bright red box made of pine
The mantle was cluttered with objects: ____ and more than one vase of dried flowers. The bejeweled lamp was at the very back, nearly invisible.	picture frames showing grandchildren and long-ago weddings, knickknacks collected from all over the world,
”We have to leave now!” Sarah shouted. ____ The only way out was up. We climbed flight after flight. The sound of the monsters banging on the door below became more distant but no less threatening.	”The zombies are going to break through any moment, and then we’ll all be goners.”
The sun was shining, and little gusts of wind brought through the window ____ shocking contrast from the stale city smells she had grown used to.	the faint scents of honeysuckle and freshly turned soil. It was a
I was minding my business at the park, when I was approached by a little girl who was crying because she had lost ____ so of course I helped search.	her cat, which she had just received for her birthday. She did not want her parents to know she’d already lost him. I’m a good person
It was a cold night, and a storm was raging out at sea. A lightning bolt lit up the sky, briefly illuminating the lighthouse ____ plummeted but just before reaching the churning water, he disappeared in a poof of purple flame!	and the young man peering hesitantly over the sheer cliff. Before the next peal of thunder he jumped. At first he
The magician pulled out of his pocket ____ and then a second one and a third. He didn't stop until soon the ground was covered with them.	a scarlet handkerchief

Table 5: Hand-written fill-in-the-blank examples used for “custom” prompt during few-shot learning.

Prompt

```
Fill in the blank with about 16 words.
Text: "We have to leave now!" Sarah shouted. ____ The
only way out was up. We climbed flight after flight. The
sound of the monsters banging on the door below became
more distant but no less threatening.
Answer: "The zombies are going to break through any
moment, and then we'll all be goners."

Fill in the blank with about 32 words.
Text: I was minding my business at the park, when I was
approached by a little girl who was crying because she
had lost ____ so of course I helped search.
Answer: her cat, which she had just received for her
birthday. She did not want her parents to know she'd al-
ready lost him. I'm a good person

Fill in the blank with about 8 words.
Text: The sun was shining, and little gusts of wind
brought through the window ____ shocking contrast from
the stale city smells she had grown used to.
Answer: the faint scents of honeysuckle and freshly
turned soil. It was a

Fill in the blank with about 8 words.
Lina went to see how candy canes were made. She watched
as the workers added dye to the hot candy. ____ Finally,
they shaped it into a cane and let it cool. Lina felt a
new appreciation for candy canes.
Answer:
```

Target Continuation

```
Then, they stretched it out to make it shiny.
```

Figure 3: In blue, one of the few-shot prompts that was derived from handwritten examples, and in green, the target example we would like to perform infilling on.

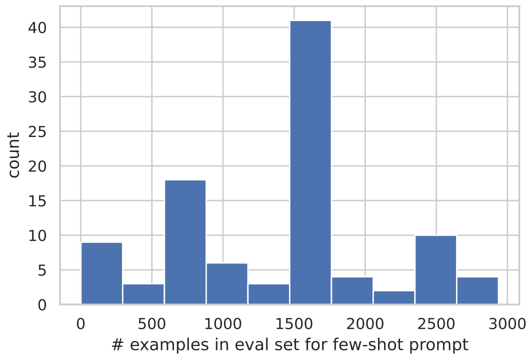
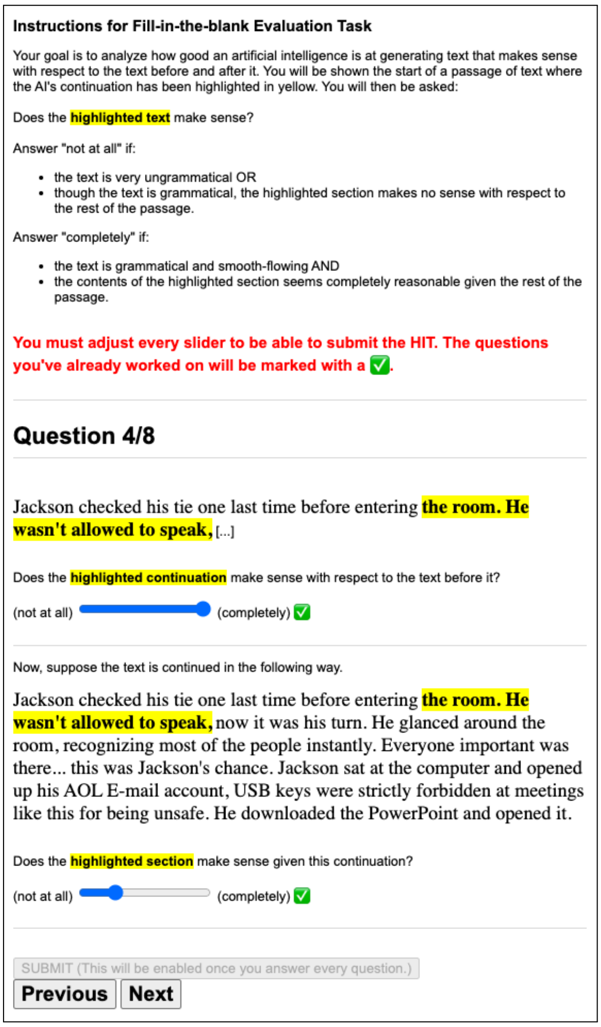


Figure 4: For each of the 4 eval sets, generation was done for 5 few-shot prompts from each of 4 possible train set sources. This histogram shows the distribution of sizes of the 80 few-shot eval sets after examples were removed that did not fit into the max context length. All 4 eval sets started off at 5,000 examlpes.



Instructions for Fill-in-the-blank Evaluation Task

Your goal is to analyze how good an artificial intelligence is at generating text that makes sense with respect to the text before and after it. You will be shown the start of a passage of text where the AI's continuation has been highlighted in yellow. You will then be asked:

Does the **highlighted text** make sense?

Answer "not at all" if:

- the text is very ungrammatical OR
- though the text is grammatical, the highlighted section makes no sense with respect to the rest of the passage.

Answer "completely" if:

- the text is grammatical and smooth-flowing AND
- the contents of the highlighted section seems completely reasonable given the rest of the passage.

You must adjust every slider to be able to submit the HIT. The questions you've already worked on will be marked with a ✅.

Question 4/8

Jackson checked his tie one last time before entering **the room. He wasn't allowed to speak,** [...]

Does the **highlighted continuation** make sense with respect to the text before it?

(not at all) (completely) ✅

Now, suppose the text is continued in the following way.

Jackson checked his tie one last time before entering **the room. He wasn't allowed to speak,** now it was his turn. He glanced around the room, recognizing most of the people instantly. Everyone important was there... this was Jackson's chance. Jackson sat at the computer and opened up his AOL E-mail account, USB keys were strictly forbidden at meetings like this for being unsafe. He downloaded the PowerPoint and opened it.

Does the **highlighted section** make sense given this continuation?

(not at all) (completely) ✅

SUBMIT (This will be enabled once you answer every question.)
Previous
Next

Figure 5: A screenshot of the question structure for human evaluation.