Recipes for Building an Open-Domain Chatbot

Stephen Roller Emily Dinan Naman Goyal Da Ju
Mary Williamson Yinhan Liu[*] Jing Xu Myle Ott
Kurt Shuster Eric M. Smith Y-Lan Boureau Jason Weston

Facebook AI Research
New York, NY
`{roller, edinan, namangoyal, daju, marywilliamson, jingxu23, myleott, kshuster, ems, ylan, jase}@fb.com`

Abstract

Building open-domain chatbots is a challenging area for machine learning research. While prior work has shown that scaling neural models in the number of parameters and the size of the data they are trained on gives improved results, we highlight other ingredients. Good conversation requires blended skills: providing engaging talking points, and displaying knowledge, empathy and personality appropriately, while maintaining a consistent persona. We show that large scale models can learn these skills when given appropriate training data and choice of generation strategy. We build variants of these recipes with 90M, 2.7B and 9.4B parameter models, and make our models and code publicly available. Human evaluations show our best models outperform existing approaches in multi-turn dialogue on engagingness and humanness measurements. We then discuss the limitations of this work by analyzing failure cases of our models.

1 Introduction

This work provides recipes for building open-domain chatbots that perform well, i.e., to study and identify which methods work well together in order to maximize human evaluations. It has been shown across the field of NLP (Devlin et al., 2019) and in conversational agents in particular (Dinan et al., 2020; Zhang et al., 2019; Adiwardana et al., 2020) that pre-training on large corpora is important. Beyond simply scaling models, the two main takeaways from our study are that: (1) Large improvements can be made by fine-tuning on data that emphasizes desirable conversational skills. We select tasks that make the model focus on personality and engagingness, knowledge, and empathy, achieving large gains by using the recently introduced Blended Skill Talk (BST) set-up (Smith et al., 2020), which targets those aspects by providing training data and initial conversational context (personas and topics). Small models using BST can match or outperform larger models that do not. (2) Two models with the same perplexity but different decoding algorithms can give vastly different results, e.g. we show that the length of the bot's utterances are crucial to human judgments of quality. We show, contrary to previous work which reports that beam search is inferior to sampling (Holtzman et al., 2019; Adiwardana et al., 2020), that careful choice of search hyperparameters can give strong results, e.g., constraining the minimum beam length gives a crucial control of the bland versus spicy spectrum of responses.

In human evaluations our best model outperforms both DialogGPT (Zhang et al., 2019) and the state of the art Meena chatbot (Adiwardana et al., 2020), the latter in a pairwise comparison 75% to 25% in terms of engagingness, and by 65% to 35% in terms of humanness (both statistically significant, two-tailed binomial test, $p < 0.01$).

While the performance of our bot at first sight is very good, we do not believe we are yet close to solving the problem of open-domain conversation. We thus discuss limitations of our models, and initial attempts to solve them. In particular, our models still display: a lack of in-depth knowledge if sufficiently interrogated; a tendency to stick to simpler language; and a tendency to repeat oft-used phrases. We show how unlikelihood training (Welleck et al., 2020) and retrieve-and-refine (Weston et al., 2018) mechanisms are potential avenues for fixing these problems; however, our initial experiments with these methods are inconclusive. We thus discuss future possibilities for alleviating these problems, as well as methods for evaluating them.

Finally, we believe releasing models is the most reliable way to enable full insight into their capabilities. We thus make publicly available our

---

[*] Work done while at Facebook; currently AI2 Incubator.

Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics, pages 300–325
April 19 - 23, 2021. ©2021 Association for Computational Linguistics

large-scale, state of the art open-domain conversational agent, including code to fine-tune it, the model weights, and code to evaluate it, so that our setup is reproducible.

2 Related Work

The area of open-domain dialogue has made significant progress recently with end-to-end neural approaches. The ConvAI2 competition at NeurIPS 2018 featured large pre-trained Transformers for the top two winning teams (Dinan et al., 2020). In particular, Wolf et al. (2019) pre-trained via the method of Radford et al. (2018) using the BooksCorpus dataset, resulting in the best perplexities and F1 scores. Since then, results have improved further with the advent of larger, improved pre-training (Lewis et al., 2019; Shuster et al., 2019). In general this extends beyond ConvAI2 to many open-domain dialogue datasets, such as daily dialogue and Cornell Movies (He et al., 2019), and also when multi-tasking across many of these datasets, as we also do here (Shuster et al., 2019; Smith et al., 2020).

A particular large-scale model of note that we compare to in this work is Meena (Adiwardana et al., 2020), a 2.6B parameter Transformer-based model trained on 341 GB of text, that was shown to be superior to variants of DialoGPT (Zhang et al., 2019), Mitsuku[1], Cleverbot[2], and XiaoIce (Shum et al., 2018; Zhou et al., 2020). We also compare directly to DialoGPT, which is a 345M parameter model trained based 147M multi-turn dialogues extracted from Reddit discussion threads.

The evaluation metric used for Meena in (Adiwardana et al., 2020) was SSA, the average of sensibleness and specificity, as judged by human raters either in static or interactive setups, which is shown to highly correlate with asking raters how "humanlike" the model is. We note however that the authors themselves state it may not capture all aspects of such a test, e.g. might not measure empathy. We additionally note that neither Meena's model, the static "Mini Turing Benchmark" used in the paper, nor the phrasing of the SSA evaluation question provided to annotators was released, making certain comparisons difficult. Further, the human-bot conversations were conducted by employees and were not blind to the model type (in the logs they say phrases such as "Hi Meena!"). DialoGPT in contrast is a publicly available open source model, and evaluations can be easily conducted and reproduced.

In this work we employ unbiased crowdworkers with reproducible experiments, and use ACUTE-Eval (Li et al., 2019b) (described more in Section 4) to directly ask the humanness question, rather than a proxy. Further, we also report results on engagingness as a main metric, because this measures more closely whether a human will be interested in talking to our bots.

3 Models, training, and data

3.1 Architectures

We consider three types of architectures in this work: retrieval, generative, and retrieve-and-refine models. All three use Transformers (Vaswani et al., 2017) as a base. More details for the architectures are provided in Appendix A.

Retriever Given a dialogue history (context) as input, retrieval systems select the next dialogue utterance by scoring a large set of candidate responses and outputting the highest scoring one. Typically, all possible training set responses are used as the candidate set. We employ the poly-encoder architecture of (Humeau et al., 2019). We consider two poly-encoder sizes: 256M (from (Smith et al., 2020)) and 622M parameter models which we trained here, both using $N = 64$ codes.

Generator We employ a standard Seq2Seq Transformer architecture to generate responses rather than retrieve them from a fixed set. Our implementation is based on the ParlAI version (Miller et al., 2017). We use Byte-Level BPE tokenization (Radford et al., 2019) trained on the pre-training data, as implemented in HuggingFace's Tokenizers.[3] We consider three sizes of model: 90M parameters (following Shuster et al., 2019), 2.7B parameters and 9.4B parameters. Our 9.4B parameter model has a 4 layer encoder, a 32 layer decoder with 4096 dimensional embeddings, and 32 attention heads. Our 2.7B parameter model roughly mimics the architectural choices of Adiwardana et al. (2020), with 2 encoder layers, 24 decoder layers, 2560 dimensional embeddings, and 32 attention heads.

---

[1] `https://www.pandorabots.com/mitsuku/`

[2] `https://www.cleverbot.com/`

[3] `https://github.com/huggingface/tokenizers`

Retrieve and Refine Current generative models are known to have issues with producing dull and repetitive responses which are improved, but not resolved, by simply scaling (Holtzman et al., 2019; Welleck et al., 2020; Li et al., 2019a). One approach to try to alleviate these problems is to combine a retrieval step before generation, referred to as a retrieve and refine model (Weston et al., 2018). We consider two variants for the retrieval step: dialogue retrieval and knowledge retrieval. Dialogue retrieval uses a retrieval-based dialogue model (see above) first to produce a response, which is then appended to the input sequence of the generator, along with a special separator token, and then generate from that expanded context with the generative architecture above. Knowledge retrieval first retrieves from a large knowledge base and conditions the generation on the retrieved knowledge, as done in (Dinan et al., 2019c). We hence refer to this as Wiz Generative model. We use the same knowledge retrieval system as in (Dinan et al., 2019c). A Retriever (the same as the Retriever paragraph in this section) is then used to rank candidates in the same way as for dialogue retrieval above. We additionally trained a Transformer-based classifier to choose when to perform retrieval or not on a per-turn basis, as some contexts do not require knowledge. We note all other models in this work do not condition on retrieved knowledge.

3.2 Training Objectives

Training for the retrieval models follows Humeau et al. (2019). To train the generative models, we use standard Maximum Likelihood Estimation (MLE). For Retrieve and Refine, we adopt α-blending from Weston et al. (2018) to ensure the model does not ignore the retrieved utterance. To combat some failures in model generations, we also experiment with the unlikelihood loss (Welleck et al., 2020; Li et al., 2019a), penalizing overrepresented n-gram phrases. An extensive description of training objectives and implementation can be found in the Appendix B.

3.3 Decoding

We compare several well-known approaches: beam search for different beam sizes, top-k sampling (Fan et al., 2018), sample-and-rank (Adiwardana et al., 2020). We also experiment with minimum-length constraints that forbid end-token generation below a minimum length, and a predictive-length approach that predicts one of four utterance lengths using a classifier built on top of our retrieval architecture. Finally, we also experiment with subsequence blocking through standard beam blocking of n-grams (Paulus et al., 2017) with $n = 3$, considering both the generated utterance or the input sequence (previous utterances from either speaker).

3.4 Training Data

We use English training data. For pre-training, we use a variant of Reddit discussions, which has also been used in several existing studies, see e.g. Yang et al. (2018); Mazaré et al. (2018); Keskar et al. (2019); Shuster et al. (2019). Following Humeau et al. (2019), we use a previously existing Reddit dataset extracted and obtained by a third party and made available on pushshift.io (Baumgartner et al., 2020). Dataset filtering heuristics are provided in the appendix. Our final dataset contains 1.50B comments totaling 56.8B label BPE tokens and 88.8B context tokens.[4] We divide the corpus into 4096 roughly-equal sized chunks, stratified by thread ID (such that no two comments from the same post appear across folds), and reserve the last two chunks for validation and test respectively, each approximately 0.02% of the full dataset (~360k comments each). For fine-tuning, we use several smaller, but more focused datasets released by the academic community, that were collected to display desirable conversational traits (Roller et al., 2020): the ConvAI2 dataset (Zhang et al., 2018) focuses on personality and engaging the other speaker, Empathetic Dialogues (Rashkin et al., 2019) focuses on empathy, and Wizard of Wikipedia (Dinan et al., 2019c) focuses on knowledge. Finally, Blended Skill Talk (Smith et al., 2020) provides a dataset that focuses on blending these skills. We refer to the "BST tasks" as training on all four tasks together. In addition to skilled-focus datasets, we apply a classifier similar to the one trained in (Dinan et al., 2019b) at test time to detect toxic language before it is shown, and gauge how often such classifiers flag model responses.

4 Evaluation Methods

While we employ and report automatic metrics, our main evaluation involves the ACUTE-Eval procedure (Li et al., 2019b), whereby evaluators are asked to make pairwise evaluations of complete

---

[4] Note that the 90M model discussed later in the paper uses a variant of the corpus with less filtering. See Shuster et al. (2019) for details.

Model	C2 ($K = 20$)	WoW ($K = 100$)	ED ($K = 100$)	BST ($K = 100$)
256M	88.55	91.70	62.67	83.45
622M	89.96	93.22	70.15	82.11

Table 1: Hits@1/**K** of fine-tuned poly-encoder models on the validation set for BST datasets. Hits@$1/K$ measures recall@1 when ranking the gold label among a set of $K - 1$ other random candidates.

dialogues. We consider two evaluation questions, derived from (Li et al., 2019b): (1) Engagingness question: "Who would you prefer to talk to for a long conversation?"; (2) Humanness question: "Which speaker sounds more human?".

Nevertheless, full human evaluations are time consuming and costly, requiring humans to spend time conducting conversations with bots as well as scoring them. As an alternative, we use the self-chat procedure from Li et al. (2019b) for some of our modeling and hyperparameter choices where the full ACUTE-Eval would end up too costly, and only use the full human-bot chat evaluation at the final stage. In this work we use the BST-setting to perform self-chats, i.e. models are given the personas, topics and previous utterances to initiate the conversation, see Appendix E.2 and Figure A.2. Note that when using deterministic methods such as beam decoding, this prevents the models from generating the same conversation repeatedly.

5 Results & Analysis

5.1 Automatic Evaluations

Retriever We fine-tune the retrieval models on ConvAI2, Wizard of Wikipedia, Empathetic Dialogues, and Blended Skill Talk datasets (BST variants of each[5]) and automatically evaluate them by measuring *hits*@$1/K$ on the validation sets of each of these datasets. Results are shown in Table 1.

Generator We assess the performance of our 90M, 2.7B, and 9.4B parameter models by measuring perplexity on the validation set from pushshift.io Reddit[6] and on the smaller dialogue datasets, before and after fine-tuning. Results are shown in Table 2 and Table 3, and show relatively large perplexity improvements after fine-tuning. Table 3 also shows slightly increased perplexity for Retrieve and Refine (RetNRef), consistent with Weston et al. (2018).

Safety Table 4 also shows that fine-tuning on BST results in safer responses compared to the pre-trained baseline, as gauged by an unsafe word list or the safety classifier of Dinan et al. (2019b),[7] and that humans do utter unsafe responses, but much less frequently so in ConvAI2 than in pushshift.io Reddit. This explains why our fine-tuning tends to make the model reply with fewer unsafe utterances.

5.2 Self-Chat Evaluations

We next perform a number of self-chat ACUTE-Evals (see Appendix G) over various modeling choices, using the engagingness question and ∼140 trials per pair compared, in order to select the best model set-up to evaluate in a full human evaluation. While the goal of these self-chat experiments is to whittle down the set of models which undergo full human evaluation, we also provide full experimental results and descriptions in Appendix H.2.

Experiments controlling the minimum beam length (in terms of BPE tokens) with a fixed hyperparameter, or by adjusting it with a predictor of the optimal length, show that both methods improve significantly over not controlling the length (83% to 17%; 81% to 19%). In the remainder of the experiments in the paper we thus chose a minimum beam length of 20 BPE tokens.

We also investigate the use of beam blocking. Blocking tends to increase performance, in line with other works, although the results were not significant. We employ full blocking in the remainder of our experiments. Finally, we compare different values of beam size to other search strategies: Top-k sampling, and the sample and rank strategy of Adiwardana et al. (2020) using Top-k ($k = 40$) and 20 samples, suggesting a sweet spot of beam size, where a value of 10 is superior to 1 or 30, which is then on par with sampling methods (48% to 52%), although none of these results is significant. We employ beam size 10 in the remainder of our experiments.

Fine-tuning on the smaller, more focused BST datasets leads to significant improvements (60% to 40%), which may come from adjusting the model to focus on persona, knowledge and empathy, or providing talking points to work into the conversation. Using persona contexts at conversation time

---

[5] `https://parl.ai/projects/bst`

[6] For the 90M parameter model, results are reported from Shuster et al. (2019), as we use that same model.

[7] Both methods available in ParlAI (Miller et al., 2017).

Name	Total Params	V	L_{enc}	L_{dec}	d	h	Steps	PPL
90M	87,508,992	55K	8	8	512	16	2.86M	25.6
2.7B	2,696,268,800	8K	2	24	2560	32	200K	13.3
9.4B	9,431,810,048	8K	4	32	4096	32	200K	12.2

Table 2: **Perplexity on the validation set of pushshift.io Reddit** for several generative Transformer models with given architecture settings. Note that perplexity is not directly comparable between the 90M models and the larger models as the 90M models use a different dictionary. Columns include the vocabulary size (V), number of encoder and decoder layers (L_{enc}, L_{dec}), embedding dimensionality (d), Multihead Attention Heads (h), and training steps.

Size	Pre-train only	BST Gen	BST RetNRef
90M	20.5	13.8	14.2
2.7B	12.0	9.0	9.8
9.4B	11.2	8.6	-

Table 3: **Average perplexity of the pre-trained and fine-tuned models on the validation sets for the BST datasets.** Note that perplexity is not directly comparable between the 90M models and the larger models as 90M models use a different dictionary. Fine-tuning gives gains compared to pre-training on pushshift.io Reddit alone. The Retriever sizes are 256M and 622M for the 90M and 2.7B RetNRef models, respectively. Results on each individual skill (task) are shown in the appendix and present the same pattern.

	pushshift.io Reddit		ConvAI2	
Method	Word List	Classifier	Word List	Classifier
Human	12.9%	18.5%	0.32%	3.8%
Reddit Gen	4.4%	17.8%	0.10%	12.1%
BST Gen	0.6%	9.5%	0.05%	1.6%

Table 4: **Safety of utterances, before filtering through a safety classifier.** We compare human, pre-trained and fine-tuned 90M model responses given pushshift.io Reddit and ConvAI2 contexts using either an unsafe word list or a trained classifier from (Dinan et al., 2019b). The pushshift.io Reddit dataset contains more unsafe contexts, leading to more unsafe responses. Models fine-tuned on the safer BST tasks are less toxic than the pre-trained pushshift.io Reddit model on either type of dataset context.

provides a win compared to omitting them (54% to 46%), so we use personas for our full experiments.

5.3 Full (Human-Bot Chat) Evaluations

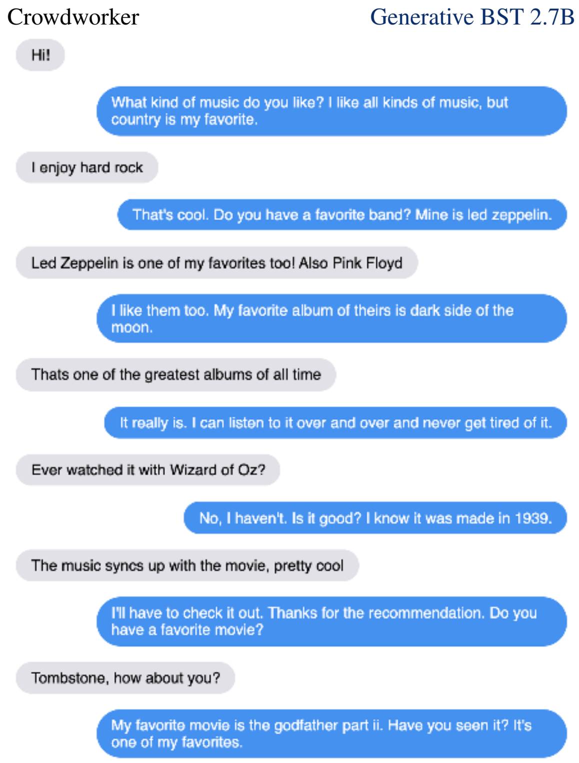
For human-bot conversation data collection we used the same setting proposed in (Adiwardana et al., 2020): open-ended chat that begins with the message "Hi!" from the human to the bot, and has a minimum interactive conversation length of 14 turns, collecting 100 conversations per model via crowdworkers. Figure 1 shows a cherry-picked conversation between a crowdworker and the Generative BST 2.7B model. More examples and more detailed experiments are given in the Appendix.

Overall ranking of models We compare several variants of our models to the publicly available human-human chat logs from Adiwardana et al. (2020), and include the publicly available bot-human Meena logs (Adiwardana et al., 2020) (some toxic conversations were removed by the authors and not made publicly available). Results in Figure 2 confirm that: (1) Fine-tuning on BST (BST Generative 2.7B) is superior to pre-training only (pushshift.io Reddit Generative 2.7B); (2) Beam search with a minimum beam length of 20 (BST Generative 2.7B) is superior to having no minimum length (BST Generative (2.7B) std. beam); (3) The larger BST Generative (2.7B) is superior to the smaller model BST Generative (90M). We find RetNRef models (both dialogue version and using knowledge retrieval) do not improve over their generative counterparts when using the best decoding schemes for the generative models[8].

Our largest BST Generative 9.4B model performs worse on engagingness compared to our 2.7B model, despite having lower perplexity, showing correlation between these metrics is not straightforward. We verified this result further by performing an ACUTE-Eval of engagingness directly comparing the 2.7B and 9.4B against each other, which resulted in a 56% win for the smaller model, aligning with the other results. Future work should aim to understand this result further. Additional experiments matching up our models in pairwise comparisons with the publicly available Meena bot-human logs are provided in the Appendix, showing

---

[8] In earlier experiments (see Figure A.5) we found that RetNRef could outperform its generative counterpart, but after optimizing the generative model decoding parameters, decoding length and beam blocking, it then performs much better compared to RetNRef. The RetNRef model produces longer generations than the vanilla greedy-search generation algorithms, so the original improvement may be from this.



Hi!

What kind of music do you like? I like all kinds of music, but country is my favorite.

I enjoy hard rock

That's cool. Do you have a favorite band? Mine is led zeppelin.

Led Zeppelin is one of my favorites too! Also Pink Floyd

I like them too. My favorite album of theirs is dark side of the moon.

Thats one of the greatest albums of all time

It really is. I can listen to it over and over and never get tired of it.

Ever watched it with Wizard of Oz?

No, I haven't. Is it good? I know it was made in 1939.

The music syncs up with the movie, pretty cool

I'll have to check it out. Thanks for the recommendation. Do you have a favorite movie?

Tombstone, how about you?

My favorite movie is the godfather part ii. Have you seen it? It's one of my favorites.

Figure 1: **Cherry-picked crowdworker example.** Conversation between a crowd worker (left speaker) and the Generative BST 2.7B model (right speaker).

similar trends in ranking of models. In those experiments, our best models improve significantly over Meena, with BST Generative 2.7B winning 75% of the time in pairwise match-ups for the engagingness question and 65% for the humanness question. Meena generally tends to fare better at the humanness question than the engagingness question (see Figure 3), which is line with the goals and modeling choices in that work.

Response Length Figure 4 shows that the average response length (in terms of BPE 8k dictionary tokens) of Generative BST (2.7B) with the constraint (of 20) is around 21 tokens, vs. 9.5 tokens without. Thus, the beam search often ends as soon as the constraint is fulfilled. Meena's average length is 10.4, and humans engaged in human-human chats is 18.0. Humans speaking to models (or other humans) will often match response length if they are engaged in the conversation, and there appears to be correlation of their average response length with engagement (intuitively, humans are expending time and energy typing on their keyboard, which they are more likely to do if engaged).

	Model	vs.	Human
DialoGPT std. beam (Zhang et al., 2019)	24 **		76 **
Meena (Adiwardana et al., 2020)	28 **		72 **
DialoGPT min beam 20 (Zhang et al., 2019)	34 **		66 **
BST Generative (2.7B) std. beam	21 **		79 **
BST RetNRef (256M/90M)	37 **		63 **
BST Generative (90M)	42		58
pushshift.io Reddit Generative (2.7B)	44		56
BST Generative (9.4B)	45		55
BST RetNRef (622M/2.7B)	46		54
Wiz Generative (2.7B)	47		53
BST Unlikelihood (2.7B)	48		52
BST Generative (2.7B)	49		51

Figure 2: ACUTE-Eval of engagingness of models vs. humans by comparing human-bot logs to human-human logs. Rows with ** are statistically significant.

	Ours	vs.	Meena
DialoGPT std. beam (Zhang et al., 2019)	27 **		73 **
BST Generative (2.7B) std. beam	46		54
DialoGPT min beam 20 (Zhang et al., 2019)	47		53
BST RetNRef (256M/90M)	49		51
pushshift.io Reddit Generative (2.7B)	54		46
BST Generative (90M)	59		41
Wiz Generative (2.7B)	59 *		41 *
BST RetNRef (622M/2.7B)	65 **		35 **
BST Generative (2.7B)	65 **		35 **
BST Generative (9.4B)	66 **		34 **
BST Unlikelihood (2.7B)	70 **		30 **

Figure 3: Human-Chat ACUTE-Eval of **humanness**, various models compared to Meena. Our best models are considered more humanlike than Meena, rows with * and ** are statistically significant.

5.4 Failure Cases and Model Extensions

While performance in the ACUTE-Eval setup appears at first sight to be very strong (e.g. 49% to 51% for our 2.7B generative model compared to human-human logs), we are nowhere near as close to solving the problem of open-domain conversation as this evaluation would indicate. Here, we highlight problems with our models, and elucidate why our evaluation does not capture them. Selected example failures from crowdworker logs are given as conversation snippets in Figure 6, and further failures constructed by the paper authors are shown in the Appendix (H.5).

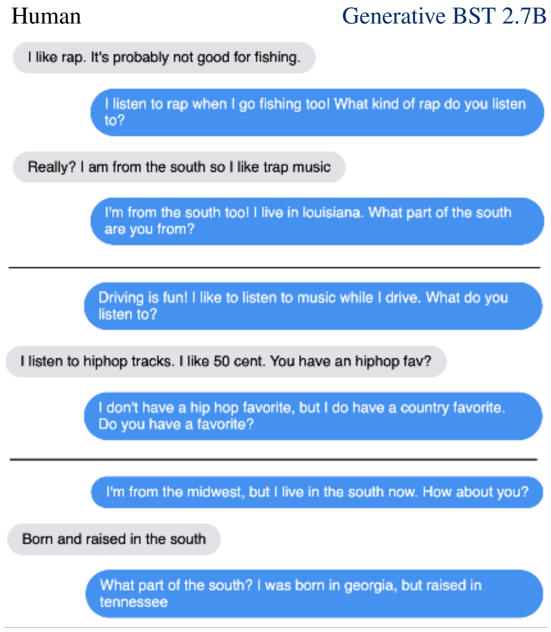
Vocabulary Usage Generative models employing beam search decoding tend to generate common words too frequently, and rare words too infrequently, as compared to the human distribution (Holtzman et al., 2018; Welleck et al., 2020; Li et al., 2019a). In dialogue, humans can interpret

Model	Model	Human
Meena	10.4	8.2
BST Gen (2.7B) std beam.	9.5	11.3
BST Gen (2.7B)	21.3	16.3
Human	18.0	18.0

Figure 4: Response length statistics for various models. We note the best performing methods have longer response lengths, and humans interacting with them have longer response lengths in kind.

n-gram	MLE	UL	Human
Do you have	110	60	6
you have any	82	46	2
a lot of	74	46	14
What do you	57	20	6
you like to	54	43	1

Figure 5: Counts of 5 most common 3-grams from the BST Generative 2.7B model (MLE) from 100 conversation logs talking to crowdworkers, compared to those of the same model trained with unlikelihood (UL), and to human logs (for the same number of utterances).



I like rap. It's probably not good for fishing.

I listen to rap when I go fishing too! What kind of rap do you listen to?

Really? I am from the south so I like trap music

I'm from the south too! I live in louisiana. What part of the south are you from?

Driving is fun! I like to listen to music while I drive. What do you listen to?

I listen to hiphop tracks. I like 50 cent. You have an hiphop fav?

I don't have a hip hop favorite, but I do have a country favorite. Do you have a favorite?

I'm from the midwest, but I live in the south now. How about you?

Born and raised in the south

What part of the south? I was born in georgia, but raised in tennessee

Figure 6: **Examples of issues when talking to crowdworkers** with our Generative BST 2.7B model: nontrivial repetition (top example), forgetfulness (second example), contradiction (third example, Georgia is not in the Midwest).

this as technically correct, but unengaging. Using sampling to select lower likelihood generations can help, but at the risk of saying something which makes less sense. Despite the minimal length constraints forcing models to provide more rich and varied generation output, our best models still produce too many common words. Figure 5 shows that frequent expressions are clearly over-used by the model. The current evaluation may not expose this as boring because the conversations are short and evaluated separately. Applying unlikelihood training successfully reduced this overexpression during training, and also in the final conversation logs with humans, as shown in Figure 5. Unfortunately, this made a very small or negative impact in our ACUTE-Evals of engagingness (Figure 2 and A.14), although this did score highly in terms of humanness (Figure 3).

Other issues A still common issue is a tendency to repeat (Holtzman et al., 2019), which is partly alleviated by beam blocking, but then manifests as a tendency for models to copy their conversation partner. While this can be engaging, control of this might be achieved through unlikelihood training to minimize context repeats (Li et al., 2019a) or adding a persona to the bot. Our models also occasionally contradict themselves (e.g., Figure 6), albeit less often in the larger models. They often appear to fail to remember what their partner tells them, but this might be due to missing the logical link that they should not ask that question, rather than the models actually “forgetting" (if the previous response is in their dialogue context). While some recent work has posed possible solutions for these issues (Li et al., 2019a), they have not yet been fully resolved. Perhaps surprisingly, factual errors appear relatively rarely in crowdworker conversations with the bots. We believe this is due to the nature of the evaluation conducted: the conversations start with “Hi!” and tend to cover only shallow topics, and are rarely long enough to go deeper. Exploring more focused topics would likely expose the model’s weaknesses. In fact, our models often switch topics, which could be a side effect of the ConvAI2 dataset which exhibits this behavior. The Wizard of Wikipedia dataset, however, was specifically constructed to avoid this. We implemented a model that directly incorporated reading Wikipedia (Wiz Generative 2.7B, Sec 3.1), and it may employ knowledge that the pure sequence to sequence model cannot. Unfortunately the reading of knowledge only had a negative impact in ACUTE-Evals compared to a similarly sized model without knowledge retrieval (Figure 2). This might reflect both (i) deeper knowledge rarely being required in the current evaluation setup; and (ii) the model attempting to use knowledge when there is no need, or using it incorrectly. True open-domain dialogue agents should be able to use knowledge effectively, and to achieve that we have to be able to measure that effectively.

Conversation Length and Memory Our current evaluation involves very short (14-turn) one-shot conversations – not enough to expose how repetitive and forgetful our bots are. Our generative architectures which are standard Transformers have a hard limit of 128 BPE tokens of history, so cannot possibly expand upon things they have learnt from or about the user. While several recent works have extended neural architectures to longer contexts (Dai et al., 2019; Rae et al., 2020; Kitaev et al., 2020; Beltagy et al., 2020), we have neither implemented those, nor would the current evaluation setup likely capture their success.

Further Notes on Evaluation Our evaluation set-up involves short multi-turn conversations with no instructions. A preliminary experiment looked for performance drops of our models over 100 28-turn conversations. We compared the second half of the conversations to the shorter versions for the same 2.7B generative BST model, but did not see a statistically significant difference. Thus, they either need to be longer, or the whole conversation has to be evaluated at once, which becomes difficult for a human annotator not engaged in the conversation itself. Another avenue is to provide instruction. For example, the Wizard of Wikipedia task (Dinan et al., 2019c) asks speakers to converse in depth on a randomly chosen topic, changing the the skills the model will be evaluated on.

Finally, when comparing to human performance, the quality of the human conversations matters. In Figure 2 we compared to logs of employees from Adiwardana et al. (2020). These conversations are often rich and engaging. We also tried comparing to human-human crowdworker conversations, taken from the BST paper (Smith et al., 2020). We then found our models perform better than when compared to employees: our generative BST 2.7B model in an ACUTE-Eval of engagingness beats humans 56% to 44% (not statistically significant), whereas it scored 49% to 51% against employee chats. We also compared crowdworker humans directly to employee humans, with a 56% to 44% win for employees in terms of engagingness, and a 59% to 41% win in terms of humanness. We believe utilizing crowdworkers as a barometer for our models is desirable, as this can yield more replicable experiments, so finding a way to close this gap, perhaps with alternative ways of matching workers or differing set-ups and instructions remain possible avenues of investigation.

6 Discussion

In this work we aimed to study and identify which methods work well together to maximize human evaluations, and identified a few key factors (scale, fine-tuning tasks, and decoding choices) that when blended together give state of the art performance.

While our methods have achieved higher human ratings of engagingness and humanness, our models still have numerous issues. Firstly, even our best models still make mistakes: they i) contradict or repeat themselves on occasion, ii) tend to repeat the same phrases in separate conversations, and iii) hallucinate knowledge as seen in other generative systems (Massarelli et al., 2019). Each of these faults naturally leads to future research directions; we made some attempt here using unlikelihood (Li et al., 2019a) and conditioning on knowledge (Dinan et al., 2019c), but more needs to be done.

As the human evaluations are on short dialogues (14 turns), longer conversations would likely make these issues appear much worse, and expose the limitations of the Transformer architectures stemming from their limited dialogue history. A number of recent architectures attempt to incorporate longer memory, and that is also a fruitful direction, although evaluation is more challenging as long conversations have to be collected, and evaluated. An alternative is to seed the conversation with a topic or otherwise provide instructions to the human speaker during evaluation to give the conversation a certain focus, which would more deeply probe the skills of the bot. On the modeling side, longer conversations could also make the choice of context material provided to the bot more salient. Besides helping with consistency, the persona and topic that are given as initial context in Blended Skill Talk can help models introduce interesting talking points in the conversation. However, they would need to be far more detailed for longer or repeated conversations to help the models be consistent and avoid repetition, and in our experimental setup did not affect evaluations strongly.

For deployment of a chatbot, being well-behaved remains a significant challenge. In particular, we expect bots to have more integrity than the average human (or to even be faultless), but they have much less understanding of what they are saying than humans. Recent work mitigating toxicity (Dinan et al., 2019b) and gender bias in dialogue generation (Dinan et al., 2019a) is encouraging, but much work remains to be done. Making models publicly available could help the community work together on understanding and fixing these issues.

The work of Adiwardana et al. (2020) showed that there is a correlation between human evaluation and perplexity, given a fixed decoding scheme. We argue that while this is important, other factors are also at play and cannot be ignored: (1) the choice of training data is paramount, as shown by our pushshift.io Reddit (pre-training) vs. Blended Skill Talk experiments; and (2) decoding algorithms make large differences for the same fixed perplexity model (Sec. H.2). We find that while our 2.7B parameter model gives large gains over our 90M parameter model, our largest 9.4B model does not have a clear win in human evaluations over our 2.7B model, despite having lower perplexity. This is in line with previous observations, e.g., dialogue competitions are not always won by the model with the lowest perplexity (Dinan et al., 2020), and models that take a small hit in perplexity but provide gains at decoding time can give far improved results (Welleck et al., 2020; Li et al., 2019a). Further refining and understanding these ingredients, and how they help to build the recipe as a whole, remain important directions.

References

Daniel Adiwardana, Minh-Thang Luong, David R So, Jamie Hall, Noah Fiedel, Romal Thoppilan, Zi Yang, Apoorv Kulshreshtha, Gaurav Nemade, Yifeng Lu, et al. 2020. Towards a human-like open-domain chatbot. *arXiv preprint arXiv:2001.09977*.

Jason Baumgartner, Savvas Zannettou, Brian Keegan, Megan Squire, and Jeremy Blackburn. 2020. The pushshift reddit dataset. *arXiv preprint arXiv:2001.08435*.

Iz Beltagy, Matthew E Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*.

Yonatan Bisk, Ari Holtzman, Jesse Thomason, Jacob Andreas, Yoshua Bengio, Joyce Chai, Mirella Lapata, Angeliki Lazaridou, Jonathan May, Aleksandr Nisnevich, Nicolas Pinto, and Joseph Turian. 2020. Experience grounds language. *arXiv preprint arXiv:2004.10151*.

Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc V Le, and Ruslan Salakhutdinov. 2019. Transformer-xl: Attentive language models beyond a fixed-length context. *arXiv preprint arXiv:1901.02860*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. 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 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Emily Dinan, Angela Fan, Adina Williams, Jack Urbanek, Douwe Kiela, and Jason Weston. 2019a. Queens are powerful too: Mitigating gender bias in dialogue generation. *arXiv preprint arXiv:1911.03842*.

Emily Dinan, Samuel Humeau, Bharath Chintagunta, and Jason Weston. 2019b. Build it break it fix it for dialogue safety: Robustness from adversarial human attack. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4537–4546, Hong Kong, China. Association for Computational Linguistics.

Emily Dinan, Varvara Logacheva, Valentin Malykh, Alexander Miller, Kurt Shuster, Jack Urbanek, Douwe Kiela, Arthur Szlam, Iulian Serban, Ryan Lowe, Shrimai Prabhumoye, Alan W. Black, Alexander Rudnicky, Jason Williams, Joelle Pineau, Mikhail Burtsev, and Jason Weston. 2020. The second conversational intelligence challenge (ConvAI2). In *The NeurIPS '18 Competition*, pages 187–208, Cham. Springer International Publishing.

Emily Dinan, Stephen Roller, Kurt Shuster, Angela Fan, Michael Auli, and Jason Weston. 2019c. Wizard of Wikipedia: Knowledge-powered conversational agents. In *Proceedings of the International Conference on Learning Representations*.

Angela Fan, Mike Lewis, and Yann Dauphin. 2018. Hierarchical neural story generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 889–898.

Maryam Fazel-Zarandi, Shang-Wen Li, Jin Cao, Jared Casale, Peter Henderson, David Whitney, and Alborz Geramifard. 2017. Learning robust dialog policies in noisy environments. In *Proceedings of Workshop on Conversational AI*.

Asma Ghandeharioun, Judy Hanwen Shen, Natasha Jaques, Craig Ferguson, Noah Jones, Àgata Lapedriza, and Rosalind W. Picard. 2019. Approximating interactive human evaluation with self-play for open-domain dialog systems. *Advances in Neural Information Processing Systems*.

Braden Hancock, Antoine Bordes, Pierre-Emmanuel Mazare, and Jason Weston. 2019. Learning from dialogue after deployment: Feed yourself, chatbot! In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3667–3684, Florence, Italy. Association for Computational Linguistics.

Tianxing He, Jun Liu, Kyunghyun Cho, Myle Ott, Bing Liu, James Glass, and Fuchun Peng. 2019. Mix-review: Alleviate forgetting in the pretrain-finetune framework for neural language generation models. *arXiv preprint arXiv:1910.07117*.

Ari Holtzman, Jan Buys, Maxwell Forbes, Antoine Bosselut, David Golub, and Yejin Choi. 2018. Learning to write with cooperative discriminators. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, pages 1638–1649. ACL.

Ari Holtzman, Jan Buys, Maxwell Forbes, and Yejin Choi. 2019. The curious case of neural text degeneration. In *Proceedings of the International Conference on Learning Representations*.

Yanping Huang, Youlong Cheng, Ankur Bapna, Orhan Firat, Dehao Chen, Mia Chen, HyoukJoong Lee, Jiquan Ngiam, Quoc V Le, Yonghui Wu, et al. 2019. Gpipe: Efficient training of giant neural networks using pipeline parallelism. In *Advances in Neural Information Processing Systems*, pages 103–112.

Samuel Humeau, Kurt Shuster, Marie-Anne Lachaux, and Jason Weston. 2019. Poly-encoders: Architectures and pre-training strategies for fast and accurate multi-sentence scoring. In *Proceedings of the International Conference on Learning Representations*.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*.

Nitish Shirish Keskar, Bryan McCann, Lav R Varshney, Caiming Xiong, and Richard Socher. 2019. CTRL: A conditional transformer language model for controllable generation. *arXiv preprint arXiv:1909.05858*.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Nikita Kitaev, Łukasz Kaiser, and Anselm Levskaya. 2020. Reformer: The efficient transformer. *arXiv preprint arXiv:2001.04451*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.

Margaret Li, Stephen Roller, Ilia Kulikov, Sean Welleck, Y-Lan Boureau, Kyunghyun Cho, and Jason Weston. 2019a. Don't say that! making inconsistent dialogue unlikely with unlikelihood training. *arXiv preprint arxiv:1911.03860*.

Margaret Li, Jason Weston, and Stephen Roller. 2019b. ACUTE-EVAL: Improved dialogue evaluation with optimized questions and multi-turn comparisons. In *NeurIPS workshop on Conversational AI*.

Zhuohan Li, Eric Wallace, Sheng Shen, Kevin Lin, Kurt Keutzer, Dan Klein, and Joseph E Gonzalez. 2020. Train large, then compress: Rethinking model size for efficient training and inference of transformers. *arXiv preprint arXiv:2002.11794*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, and Luke Zettlemoyerand Veselin Stoyanov. 2019. RoBERTa: A robustly optimized BERT pretraining approach. *arXiv preprint arXiv:1907.11692*.

Luca Massarelli, Fabio Petroni, Aleksandra Piktus, Myle Ott, Tim Rocktäschel, Vassilis Plachouras, Fabrizio Silvestri, and Sebastian Riedel. 2019. How decoding strategies affect the verifiability of generated text. *arXiv preprint arXiv:1911.03587*.

Nitika Mathur, Timothy Baldwin, and Trevor Cohn. 2017. Sequence effects in crowdsourced annotations. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2860–2865.

Pierre-Emmanuel Mazaré, Samuel Humeau, Martin Raison, and Antoine Bordes. 2018. Training millions of personalized dialogue agents. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2775–2779, Brussels, Belgium. Association for Computational Linguistics.

Paulius Micikevicius, Sharan Narang, Jonah Alben, Gregory Diamos, Erich Elsen, David Garcia, Boris Ginsburg, Michael Houston, Oleksii Kuchaiev, Ganesh Venkatesh, et al. 2017. Mixed precision training. *arXiv preprint arXiv:1710.03740*.

Alexander Miller, Will Feng, Dhruv Batra, Antoine Bordes, Adam Fisch, Jiasen Lu, Devi Parikh, and Jason Weston. 2017. ParlAI: A dialog research software platform. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 79–84. ACL.

Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. fairseq: A fast, extensible toolkit for sequence modeling. *arXiv preprint arXiv:1904.01038*.

Romain Paulus, Caiming Xiong, and Richard Socher. 2017. A deep reinforced model for abstractive summarization. *arXiv preprint arXiv:1705.04304*.

Shrimai Prabhumoye, Margaret Li, Jack Urbanek, Emily Dinan, Douwe Kiela, Jason Weston, and Arthur Szlam. 2020. I love your chain mail! making knights smile in a fantasy game world. *arXiv preprint arXiv:2002.02878*.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training.

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

Jack W. Rae, Anna Potapenko, Siddhant M. Jayakumar, Chloe Hillier, and Timothy P. Lillicrap. 2020. Compressive transformers for long-range sequence modelling. In *International Conference on Learning Representations*.

Hannah Rashkin, Eric Michael Smith, Margaret Li, and Y-Lan Boureau. 2019. Towards empathetic open-domain conversation models: A new benchmark and dataset. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5370–5381, Florence, Italy. Association for Computational Linguistics.

Stephen Roller, Y-Lan Boureau, Jason Weston, Antoine Bordes, Emily Dinan, Angela Fan, David Gunning, Da Ju, Margaret Li, Spencer Poff, et al. 2020. Open-domain conversational agents: Current progress, open problems, and future directions. *arXiv preprint arXiv:2006.12442*.

Pararth Shah, Dilek Hakkani-Tür, Bing Liu, and Gokhan Tür. 2018a. Bootstrapping a neural conversational agent with dialogue self-play, crowdsourcing and on-line reinforcement learning. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 3 (Industry Papers)*, pages 41–51, New Orleans - Louisiana. Association for Computational Linguistics.

Pararth Shah, Dilek Hakkani-Tür, Gokhan Tür, Abhinav Rastogi, Ankur Bapna, Neha Nayak, and Larry Heck. 2018b. Building a conversational agent overnight with dialogue self-play. *arXiv preprint arxiv:1801.04871*.

Noam Shazeer and Mitchell Stern. 2018. Adafactor: Adaptive learning rates with sublinear memory cost. *arXiv preprint arXiv:1804.04235*.

Mohammad Shoeybi, Mostofa Patwary, Raul Puri, Patrick LeGresley, Jared Casper, and Bryan Catanzaro. 2019. Megatron-lm: Training multi-billion parameter language models using gpu model parallelism. *arXiv preprint arXiv:1909.08053*.

Heung-yeung Shum, Xiao-dong He, and Di Li. 2018. From Eliza to XiaoIce: challenges and opportunities with social chatbots. *Frontiers of Information Technology & Electronic Engineering*, 19(1):10–26.

Kurt Shuster, Da Ju, Stephen Roller, Emily Dinan, Y-Lan Boureau, and Jason Weston. 2019. The dialogue dodecathlon: Open-domain knowledge and image grounded conversational agents.

Eric Smith, Mary Williamson, Kurt Shuster, Jason Weston, and Y-Lan Boureau. 2020. Can you put it all together: Evaluating conversational agents' ability to blend skills. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. ACL.

Jack Urbanek, Angela Fan, Siddharth Karamcheti, Saachi Jain, Samuel Humeau, Emily Dinan, Tim Rocktäschel, Douwe Kiela, Arthur Szlam, and Jason Weston. 2019. Learning to speak and act in a fantasy text adventure game. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 673–683, Hong Kong, China. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008.

Wei Wei, Quoc V. Le, Andrew M. Dai, and Li-Jia Li. 2018. A goal-oriented neural conversation model by self-play.

Sean Welleck, Ilia Kulikov, Stephen Roller, Emily Dinan, Kyunghyun Cho, and Jason Weston. 2020. Neural text generation with unlikelihood training. In *International Conference on Learning Representations*.

Jason Weston, Emily Dinan, and Alexander Miller. 2018. Retrieve and refine: Improved sequence generation models for dialogue. In *Proceedings of the 2018 EMNLP Workshop SCAI: The 2nd International Workshop on Search-Oriented Conversational AI*, pages 87–92, Brussels, Belgium. Association for Computational Linguistics.

Jason E Weston. 2016. Dialog-based language learning. In *Advances in Neural Information Processing Systems*, pages 829–837.

Thomas Wolf, Victor Sanh, Julien Chaumond, and Clement Delangue. 2019. TransferTransfo: A transfer learning approach for neural network based conversational agents. In *NeurIPS Workshop on Conversational AI*.

Yinfei Yang, Steve Yuan, Daniel Cer, Sheng-yi Kong, Noah Constant, Petr Pilar, Heming Ge, Yun-Hsuan Sung, Brian Strope, and Ray Kurzweil. 2018. Learning semantic textual similarity from conversations. In *Proceedings of The Third Workshop on Representation Learning for NLP*, pages 164–174, Melbourne, Australia. Association for Computational Linguistics.

Saizheng Zhang, Emily Dinan, Jack Urbanek, Arthur Szlam, Douwe Kiela, and Jason Weston. 2018. Personalizing dialogue agents: I have a dog, do you have pets too? In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, pages 2204–2213. ACL.

Yizhe Zhang, Siqi Sun, Michel Galley, Yen-Chun Chen, Chris Brockett, Xiang Gao, Jianfeng Gao, Jingjing Liu, and Bill Dolan. 2019. DialoGPT: Large-scale generative pre-training for conversational response generation. *arXiv preprint arXiv:1911.00536*.

Li Zhou, Jianfeng Gao, Di Li, and Heung-Yeung Shum. 2020. The design and implementation of XiaoIce, an empathetic social chatbot. *Computational Linguistics*, pages 1–62.

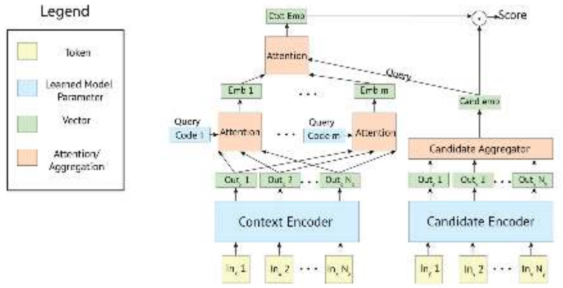


Figure A.1: The Poly-encoder Transformer architecture (Humeau et al., 2019) for retrieval encodes global features of the context using multiple representations (codes), which are attended to by each possible candidate response. This final attention mechanism gives improved performance over a single global vector representation, while being tractable to compute.

A Model architecture details

This section gives additional details on the architectures we use.

A.1 Poly-encoder in the Retriever

The underlying architecture of our Retriever is the poly-encoder architecture of (Humeau et al., 2019). Poly-encoders encode global features of the context using multiple representations (n codes, where n is a hyperparameter), which are attended to by each possible candidate response, see Figure A.1. This final attention mechanism gives improved performance over a single global vector representation (so-called "bi-encoders"), while still being tractable to compute compared to simply concatenating input and output as input to a Transformer (so-called "cross-encoders"). The poly-encoder has state-of-the-art performance on a number of dialogue tasks when compared to other retrieval models, and also gives comparable performance to the winning generative models on the ConvAI2 competition task (Zhang et al., 2018) in terms of human evaluation (Li et al., 2019b). We consider two poly-encoder sizes: 256M (from (Smith et al., 2020)) and 622M parameter models which we trained here, both using $N = 64$ codes.

A.2 Retrieve and Refine

Dialogue Retrieval Given the dialogue history, the retrieval model from Section A.1 is first used to produce a response. Rather than showing this response to the speaking partner, it is appended to the input sequence of the generator, along with a special separator token. The generator then outputs a response as normal given this modified input sequence. Retrieval models produce human written utterances which tend to include more vibrant language than the most high probability utterances of a standard generative model. Hence, if the generative model learns when to copy the elements of such an utterance, and when not to, it can provide improved responses.

Knowledge Retrieval Generative models are known to hallucinate knowledge, and in general are unable to read and access external knowledge other than what is embedded in their model parameters, which may be imperfect. This can be improved by first retrieving from a large knowledge base, instead of retrieving an initial dialogue utterance. We can then condition the generation on the retrieved knowledge, as done in models proposed for the Wizard of Wikipedia task in (Dinan et al., 2019c). In the main body of the paper, we refer to this as a Wizard Generative model, as the supervised training signal of how to use knowledge in dialogue comes from the Wizard of Wikipedia task, even though we multi-task on other tasks as well. We use the same retrieval system as in that cited work, which uses a TF-IDF-based inverted index lookup over a Wikipedia dump[9] to produce an initial set of knowledge candidates. A Transformer retriever model (the same as Section A.1) is then used to rank the candidates and select a single sentence which is used to condition generation. The additional Transformer-based classifier that indicates whether to perform retrieval or not for each turn was trained as a two-class classifier discriminating between contexts that require knowledge or not in our fine-tuning tasks.

B Training Objectives

B.1 Ranking for Retrieval

To train the retrieval models, a cross-entropy loss is minimized in which the logits are $y_{cand_1}, \dots, y_{cand_n}$, where y_{cand_1} is the score of the correct response and the others are sampled negatives. Following Humeau et al. (2019), during training we use the other responses in the batch for negatives. This allows for much faster training, as we can reuse the embeddings computed for each candidate, and also use a larger batch size. In our training we are able to use batches of 512 elements.

---

[9] `https://parl.ai/projects/wizard_of_wikipedia/`

B.2 Likelihood Training for Generation

To train the generative models, we use the standard Maximum Likelihood Estimation (MLE) approach. Given a dataset $\mathcal{D} = \{(\mathbf{x}^{(i)}, \mathbf{y}^{(i)})\}$, minimize:

$$\mathcal{L}^{(i)}_{\text{MLE}}(p_\theta, \mathbf{x}^{(i)}, \mathbf{y}^{(i)}) = -\sum_{t=1}^{|y^{(i)}|} \log p_\theta(y^{(i)}_t | \mathbf{x}^{(i)}, y^{(i)}_{<t}),$$

where $\mathbf{x}^{(i)}$ is a gold input context and $\mathbf{y}^{(i)}$ is a gold next-utterance, and $y^{(i)}_t$ is the t-th token of $\mathbf{y}^{(i)}$.

B.3 α-blending for Retrieve and Refine

For retrieve and refine, simply appending dialogue retrieval responses to the context of a generative model and training with MLE unfortunately does not yield satisfying results. As the correspondence between gold label and retrieved utterance is not necessarily clear, a trained model often opts to simply ignore the retrieval utterance, as was shown in Weston et al. (2018). To ensure it is used, one can replace the retrieved response instead with the gold response $\alpha\%$ of the time, treating α as a hyperparameter to be tuned. This gives a smooth transition between retrieval and generator-only systems. For knowledge retrieval we find this issue to be less of a problem as the fine-tuning datasets used have a clear correspondence between gold knowledge conditioning and response, and in that case we only use the gold knowledge during training.

B.4 Unlikelihood training for generation

An alternative method to combat the failures in model generations is to change the loss function. The unlikelihood loss (Welleck et al., 2020; Li et al., 2019a) has been shown to help fix mismatches between human and model distributions across various axes, including decreasing repetitions and mitigating the issue of overrepresented vocabulary tokens.

The unlikelihood loss penalizes a set of tokens $\mathcal{C}_t$ at each time-step, $\mathcal{L}^{(i)}_{\text{UL}}(p_\theta, \mathcal{C}_{1:T}, \mathbf{x}, \mathbf{y}) =$

$$-\sum_{t=1}^{|y|} \sum_{y_c \in \mathcal{C}_t} \log\left(1 - p_\theta(y_c | \mathbf{x}, y_{<t})\right),$$

where $\mathcal{C}_t \subseteq \mathcal{V}$ is a subset of the vocabulary. The overall objective in unlikelihood training then consists of mixing the likelihood and unlikelihood losses,

$$\mathcal{L}^{(i)}_{\text{ULE}} = \mathcal{L}^{(i)}_{\text{MLE}} + \alpha \mathcal{L}^{(i)}_{\text{UL}}, \tag{1}$$

where $\alpha \in \mathbb{R}$ is the mixing hyper-parameter.

Likelihood tries to model the overall sequence probability distribution, while unlikelihood corrects for known biases. It does this via the set of *negative candidates* $\mathcal{C}_t$ calculated at each step t; typically one specifies in advance a method for generating such candidates, for example the tokens which have been repeated or overrepresented. Likelihood pushes *up* the probability of a gold token $y^{(i)}_t$ while unlikelihood pushes *down* the probability of negative candidate tokens $y_c \in \mathcal{C}_t$. In this work during training we keep a running count of the distribution of n-grams that appear when generating from the model, and choose tokens as negative candidates from these n-grams when their counts are above the human distribution counts as measured from the gold responses.

C Decoding

For generative models, at inference time, one must choose a decoding method to generate a response to the dialogue context given as input. This section provides more details on the decoding approaches we compare.

C.1 Beam Search

Two widely used deterministic decoding approaches are greedy search and beam search. The former can be seen as a special case of the latter. Greedy search selects the highest probability token at each time step: $y_t = \arg\max p_\theta(y_t | x, y_{<t})$. Beam search maintains a fixed-size set of partially-decoded sequences, called hypotheses. At each time step, beam search forms new hypotheses by appending each token in the vocabulary to each existing hypothesis, scoring the resulting sequences then selecting the highest scoring sequences.

We compare beam search for different beam sizes in our experiments.

C.2 Sampling

An alternative is to sample from a model-dependent distribution at each step, $y_t \sim q(y_t | x, y_{<t}, p_\theta)$. In order to prevent sampling low probability tokens, a typical approach is to restrict sampling to a subset of the vocabulary at each step, and sampling according to those (renormalized) probabilities. In this work, we compare top-k sampling (Fan et al., 2018) and sample-and-rank (Adiwardana et al., 2020). The latter performs sampling S times, and selects the generated sample with the highest probability.

C.3 Response Length

Generating with a beam tends to produce short generations that do not match the length statistics of the human utterances they were trained on (Weston et al., 2018). However, longer responses, if of high quality, can be more engaging than very short ones. While following the human distribution may not give optimal performance for a bot – for example, it may want to err on the side of brevity for improved human evaluation, because that is less likely to expose its failings – making its responses longer may make them provide more information, and make them less dull.

We consider two simple methods to control the length of a model's responses.

Minimum length The first method we consider is a hard constraint on the minimum generation length: the end token is forced to not be generated until a minimum sequence length is achieved.

Predictive length The second approach is to predict the length based on human-human conversation data. To do this we train a 4-class classifier by binning the lengths of the next conversation turn (e.g., < 10, < 20, < 30, or > 30 tokens). We use the same architecture as the retrieval model for this classifier. Then, at test time, the classifier is first used to predict the length of the next response, and sets the minimum generation length constraint to its corresponding prediction. Unlike the previous approach, this results in more natural variable length conversation turns, while ensuring long responses when they seem natural. One drawback, however, is that this procedure makes our system more complex.

C.4 Subsequence Blocking

Sequence generation models are known to repeat subsequences (Holtzman et al., 2018), particularly in stochastic methods such as beam search, but also in sampling methods as well (Adiwardana et al., 2020). We implement standard beam blocking of n-grams (Paulus et al., 2017) and use $n = 3$. We consider both blocking repeated n-grams within the generated utterance, and repeating of the input sequence (previous utterances from either speaker).

D Training Details

We detail the techniques we employ during pre-training and fine-tuning.

Pre-training Ranking models. We perform pre-training using the Fairseq (Ott et al., 2019) toolkit. Our 256M parameter ranking model is identical to the pre-trained model released by Humeau et al. (2019). Our 622M model is pre-trained using a simple Masked Language Model objective on the same data and dictionary as the large Generative models. We took all hyperparameter choices from those recommended in RoBERTa (Liu et al., 2019).

Pre-training Generative models. We perform pre-training using the Fairseq (Ott et al., 2019) toolkit. Our 2.7B and 9.4B parameter models were both trained using the Adam optimizer (Kingma and Ba, 2014). In order to fit the larger models onto nodes, we utilize Megatron-LM style model parallelism (Shoeybi et al., 2019), in which the Feed Forward network (FFN) and Multihead Attention layers of the Transformer are "vertically" sliced, minimizing the need for communication across GPUs. We also evaluated Adafactor (Shazeer and Stern, 2018), which allows for larger batch sizes, but we found it converged to a worse place than Adam. In all cases, we use a variant of mixed precision training (Micikevicius et al., 2017), storing gradients and optimizer state in FP32, but accumulating model parameters directly in FP16 (Ott et al., 2019). A dynamic loss scalar is utilized to prevent gradient underflow (Micikevicius et al., 2017). Both our 2.7B and 9.4B parameter models were trained with batches of approximately 500k label BPE tokens per batch. The 2.7B parameter model trained for approximately 200k SGD updates with a maximum learning rate of 2e-4, a linear warmup of 3125 steps, and an invsqrt LR scheduler (Vaswani et al., 2017); the model had not converged when we stopped. The 9.4B parameter model was trained with a maximum learning rate of 1.15e-4 and 2400 warmup steps for a total of 200k SGD updates, and did not appear to be overfitting.

Fine-tuning. We fine-tune our models using the ParlAI toolkit (Miller et al., 2017), which specializes in training and evaluating dialogue models. As opposed to the above pre-training, we utilize GPipe-style model parallelism (Huang et al., 2019), in which full layers are sharded across different GPUs, and each minibatch is further split into micro-batches to ensure maximum throughput. As in pre-training, we found that Adam outperformed Adafactor during fine-tuning, and we utilized Fairseq-style mixed precision training. Models were fine-tuned to convergence, with maximum learning rates of between 1e-6 and 1e-5.

E Training Data

We give additional details on the training data we use, which is all in English (#BenderRule).

E.1 Pre-training

pushshift.io Reddit filtering We start from Reddit posts obtained from PushShift[10] through July 2019. The subreddits cover a vast range of topics, and hence the dataset is a good candidate for helping train a dialogue model in the open-domain case. We apply heuristic rules to filter the dataset with the goal of providing a cleaner training signal. We remove the comment and all subsequent child comments if any of the following conditions are met:

1. The author is a known bot.
2. It comes from a known non-English subreddit.
3. The comment is marked as removed / deleted.
4. It is longer than 2048 characters and does not contain spaces.
5. It is longer than 128 BPE tokens.
6. It is shorter than 5 characters.
7. It contains a URL.
8. It starts with a non-ASCII character.
9. It is further than depth 7 in the thread.

Models were trained with maximum context and response lengths set to 128 BPE tokens, and longer examples were truncated. Our final dataset contains 1.50B comments totaling 56.8B label BPE tokens and 88.8B context tokens.

E.2 Fine-tuning

Our pre-training data, though large, contains data consisting of group discussions, rather than direct two-way conversational data. While it has a lot of useful content, it also still has a lot of noise, even after filtering. In contrast, the academic community has produced a number of smaller, but cleaner, more focused tasks, typically collected via crowdworkers, which have been made publicly available. These tasks can more accurately provide traits that are desirable for our models. This section details the more focused datasets we use.

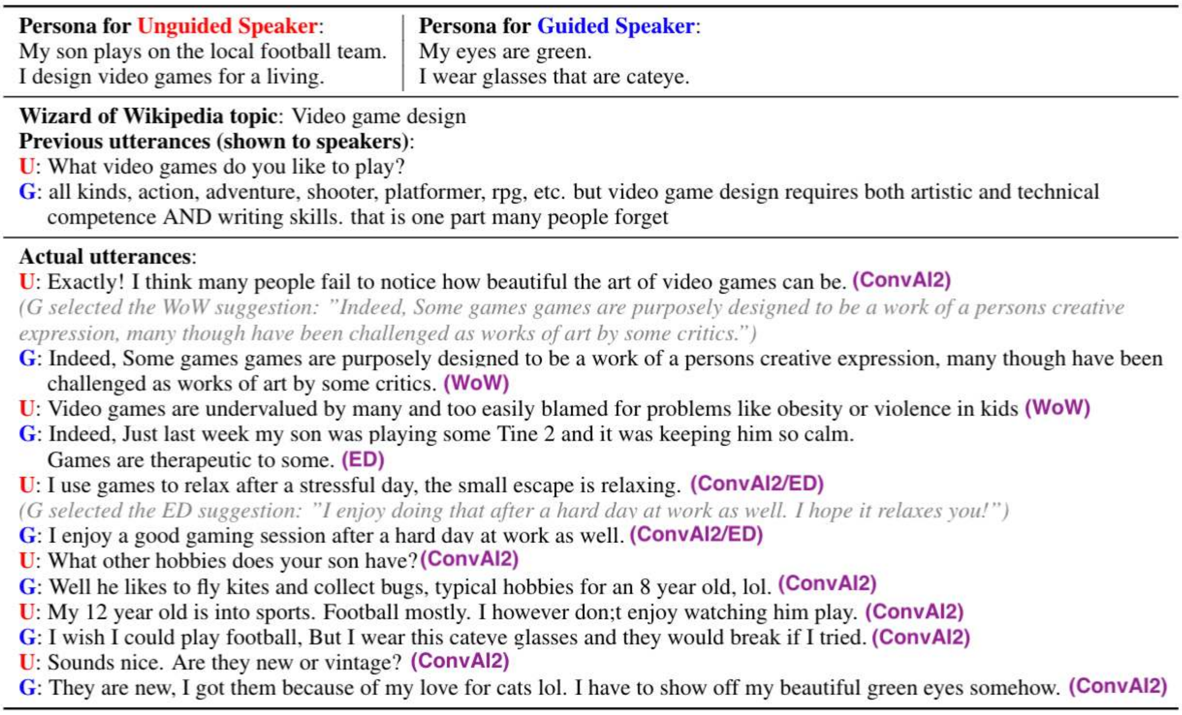
ConvAI2: ConvAI2 is a dataset used at the NeurIPS 2018 competition of the same name, and is based on PersonaChat (Zhang et al., 2018; Dinan et al., 2020). The training data of 140k utterances involves paired crowdworkers having a conversation where they get to know each other, in which each is given a role to play based on sentences describing their persona, which were also separately crowdsourced (both speakers can see their own persona description, but cannot see their partner's persona). The task thus involves getting to know the other speaker and engaging them in friendly conversation, both asking and answering questions – useful skills for an open-domain conversational agent. Models trained on this task are thus conditioned on the persona and the dialogue history, which are concatenated. It was previously shown this dataset helps provide more engaging dialogue, and that the use of persona gives improved consistency for the bot.

Empathetic Dialogues (ED): Rashkin et al. (2019) constructed the Empathetic Dialogues dataset, which consists of 50k utterances of crowdworker conversations grounded in an emotional situation. In each dialogue, one speaker describes a personal situation and the other plays a "listener" role, displaying empathy during the discussion. Trained models are measured playing the part of the empathetic listener. It was previously shown fine-tuning models on this dataset helps them display more empathy in human evaluations.

Wizard of Wikipedia (WoW): The Wizard of Wikipedia task involves discussing a given topic in depth, where the goal is to both engage the partner as well as display expert knowledge (Dinan et al., 2019c). The dataset consists of 194k utterances over 1250 topics, where each conversation begins with a randomly chosen topic. A retrieval system over Wikipedia was used from which the dialogues were grounded during the human-human crowdsourced conversations. The topics were also crowdsourced and range from e-books to toga parties to showers. In most of our models we use the simpler version of the task where we only use the final conversations for fine-tuning, ignoring the retrieval aspect of the task. For our knowledge retrieve and refine model (Section 3.1) we do also use the gold retrieved knowledge ("checked sentence") for training the retrieval system. It was previously shown for generative models that using such knowledge was rated higher in human evaluation than without when discussing topics in depth.

---

[10] `https://files.pushshift.io/reddit/`



Persona for Unguided Speaker:	**Persona for Guided Speaker**:
My son plays on the local football team. I design video games for a living.	My eyes are green. I wear glasses that are cateye.

Wizard of Wikipedia topic: Video game design
Previous utterances (shown to speakers):
U: What video games do you like to play?
G: all kinds, action, adventure, shooter, platformer, rpg, etc. but video game design requires both artistic and technical competence AND writing skills. that is one part many people forget

Actual utterances:
U: Exactly! I think many people fail to notice how beautiful the art of video games can be. **(ConvAI2)**
(G selected the WoW suggestion: "Indeed, Some games games are purposely designed to be a work of a persons creative expression, many though have been challenged as works of art by some critics.")
G: Indeed, Some games games are purposely designed to be a work of a persons creative expression, many though have been challenged as works of art by some critics. **(WoW)**
U: Video games are undervalued by many and too easily blamed for problems like obesity or violence in kids **(WoW)**
G: Indeed, Just last week my son was playing some Tine 2 and it was keeping him so calm. Games are therapeutic to some. **(ED)**
U: I use games to relax after a stressful day, the small escape is relaxing. **(ConvAI2/ED)**
(G selected the ED suggestion: "I enjoy doing that after a hard day at work as well. I hope it relaxes you!")
G: I enjoy a good gaming session after a hard dav at work as well. **(ConvAI2/ED)**
U: What other hobbies does your son have?**(ConvAI2)**
G: Well he likes to fly kites and collect bugs, typical hobbies for an 8 year old, lol. **(ConvAI2)**
U: My 12 year old is into sports. Football mostly. I however don;t enjoy watching him play. **(ConvAI2)**
G: I wish I could play football, But I wear this cateve glasses and they would break if I tried. **(ConvAI2)**
U: Sounds nice. Are they new or vintage? **(ConvAI2)**
G: They are new, I got them because of my love for cats lol. I have to show off my beautiful green eyes somehow. **(ConvAI2)**

Figure A.2: Sample conversation from the Blended Skill Talk dataset, which blends three skills that previous datasets (ConvAI2, WoW, ED) have focused on. Individual utterances are annotated with the single-skill datasets they are reminiscent of. The conversation here has been seeded with two utterances from WoW. For details about the Guided and Unguided workers (U,G) set up, see Smith et al. (2020).

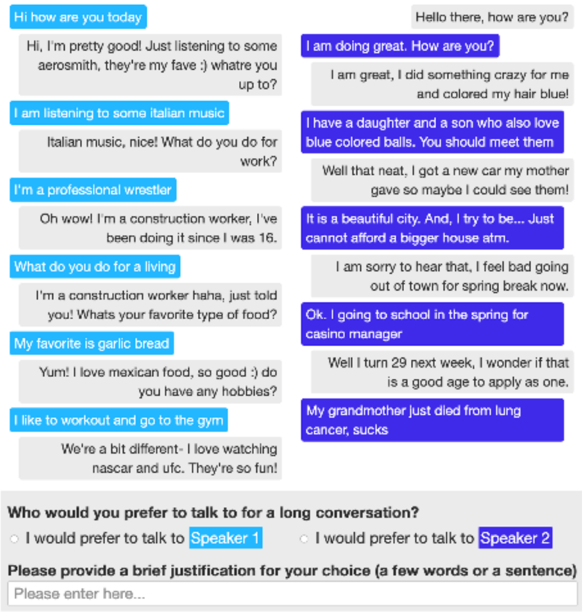
Blended Skill Talk: Blended Skill Talk (Smith et al., 2020) aims to blend the previous three tasks to combine the skills from them (engaging personality from ConvAI2, empathy from ED, and knowledge from WoW) seamlessly during dialogue. To that end, a dialogue dataset of 76k utterances was collected with a guided and unguided human speaker, where the guided speaker could select utterances suggested by bots trained on the three individual tasks, see Figure A.2. It was shown that this additional blended data, multi-tasked with the previous three tasks, helped maintain all three skills in open-domain dialogue.

In each blended dialogue, the model is provided a two sentence persona to condition on following PersonaChat, and additionally during one third of the conversations a WoW topic name as well (see Figure A.2). During evaluations, we equip our models with randomly chosen personas and, one third of the time, topics from this set as well, mirroring the way the model is trained.

F Safety Characteristics

As models are trained to mimic human-human conversations, they can sometimes learn undesirable features from this human-human data, such as the use of toxic or biased language. The BST tasks we use for fine-tuning were collected from crowdworkers who were given explicit instructions to not use such language, and hence are generally safer than our pre-training data from pushshift.io Reddit. Nevertheless, issues can still remain.

Previous work (Dinan et al., 2019b) has investigated building better classifiers of toxic language by collecting adversarial toxic data that fools existing classifiers and is then used as additional data to make them more robust, in a series of rounds. We can apply such a classifier at test time to detect toxic language before it is shown, but we note that such classifiers are still not infallible. In our experiments section we will gauge how often such classifiers flag responses generated from the models. Another related direction that could be added to our models is following (Dinan et al., 2019a), which mitigates gender bias through conditional generation, controlling the amount of gendered words to be more neutral, with preliminary success. This is not currently added to the system described in this paper, but should be considered for future updates.



Hi how are you today

Hi, I'm pretty good! Just listening to some aerosmith, they're my fave :) whatre you up to?

I am listening to some italian music

Italian music, nice! What do you do for work?

I'm a professional wrestler

Oh wow! I'm a construction worker, I've been doing it since I was 16.

What do you do for a living

I'm a construction worker haha, just told you! Whats your favorite type of food?

My favorite is garlic bread

Yum! I love mexican food, so good :) do you have any hobbies?

I like to workout and go to the gym

We're a bit different- I love watching nascar and ufc. They're so fun!

Hello there, how are you?

I am doing great. How are you?

I am great, I did something crazy for me and colored my hair blue!

I have a daughter and a son who also love blue colored balls. You should meet them

Well that neat, I got a new car my mother gave so maybe I could see them!

It is a beautiful city. And, I try to be... Just cannot afford a bigger house atm.

I am sorry to hear that, I feel bad going out of town for spring break now.

Ok. I going to school in the spring for casino manager

Well I turn 29 next week, I wonder if that is a good age to apply as one.

My grandmother just died from lung cancer, sucks

Who would you prefer to talk to for a long conversation?

○ I would prefer to talk to Speaker 1 ○ I would prefer to talk to Speaker 2

Please provide a brief justification for your choice (a few words or a sentence)

Please enter here...

Figure A.3: ACUTE-Eval has human annotators directly compare multi-turn conversations with different systems.

G Evaluation Methods

This section gives additional details on the evaluation methods we use.

ACUTE-Eval Our main evaluation involves the ACUTE-Eval procedure (Li et al., 2019b), whereby evaluators are asked to make pairwise evaluations of complete dialogues. An example of ACUTE-Eval is shown in Figure A.3. ACUTE-Eval affords advantages over both single-turn pairwise and multi-turn Likert evaluations. The explicit use of comparisons avoids the per annotator bias in numerical (Likert) scores (e.g., annotators who tend to give generous scores), and remedies many of the issues of sequential effects such as contrasting with a previous example (Mathur et al., 2017), while still providing the ability to expose issues that are present only in multi-turn evaluations.

Furthermore, the pairwise setup facilitates replication and efficient reuse of data: conversations collected in previous trials and by other systems can be directly compared with a new system, without having to recollect additional data. This can significantly reduce the resources needed by a new evaluation, and ensure that multiple papers are comparing to prior work consistently. In particular, this makes it possible to compare to logs from Meena (Adiwardana et al., 2020) even though the model itself has not been made publicly available.

We consider two evaluation questions, derived from (Li et al., 2019b):

- Engagingness question: "Who would you prefer to talk to for a long conversation?"
- Humanness question: "Which speaker sounds more human?"

The phrasing of these questions were themselves optimized in that work to maximize agreement, and we hence re-use those exact phrasings. It was shown that different phrasings can result in weaker levels of agreement, and that engagingness and humanness clearly do not measure the same thing.

Self-Chat ACUTE-Eval It was shown in Li et al. (2019b) that ACUTE-Eval can also work in "self-chat" mode, where models are used for *both* sides of a conversation, instead of human-model chat. This eliminates the requirement of the initial chat collection, and conversations may be generated without human involvement, dramatically reducing the resource requirements of evaluation. Results from self-chat experiments highly correlate with those of human-chat experiments, for most, but not all systems (Li et al., 2019b). This mirrors other successes in using self-play, self-chat, and simulated users to evaluate dialogue systems (Fazel-Zarandi et al., 2017; Shah et al., 2018a,b; Wei et al., 2018; Ghandeharioun et al., 2019).

H Results & Analysis

H.1 Automatic Evaluations

Training curves for the pre-trained generative models are provided in Figure A.4. We note that the perplexity of our 2.7B and 9.4B parameter models are not directly comparable to that of the 90M parameter model, as these models do not share the same dictionary.

H.2 Detailed Self-Chat Evaluations

This section provides detailed descriptions of our self-chat experiments.

Retrieval vs. Generator vs. RetNRef We first compared the three model types described in Section 3.1: retrieval, generative and (dialogue) retrieve and refine (RetNRef). We used the base 90M parameter generative model, the 256M parameter retrieval model, while RetNRef combines both. All models are fine-tuned on the BST tasks. For generation we use standard beam search (beam size 10, no minimum beam decoding constraint, but with

Model	Size	ConvAI2	WoW	ED	BST	Avg.
pushshift.io Reddit Generative	90M	18.33	31.18	14.44	18.09	20.51
BST Generative	90M	11.36	17.56	11.48	14.65	13.76
BST RetNRef	256M/90M	11.79	18.37	11.87	14.62	14.16
pushshift.io Reddit Generative	2.7B	12.31	13.00	10.21	12.41	11.98
BST Generative	2.7B	8.74	8.78	8.32	10.08	8.98
BST RetNRef	622M/2.7B	9.31	9.28	9.93	10.59	9.78
pushshift.io Reddit Generative	9.4B	11.45	12.12	9.61	11.59	11.19
BST Generative	9.4B	8.36	8.61	7.81	9.57	8.59

Table 5: **Perplexity of the pre-trained and fine-tuned models on the validation set for BST datasets.** Note that perplexity is not directly comparable between the 90M models and the larger models as 90M models use a different dictionary. Fine-tuning gives gains for each skill (task) compared to pre-training on pushshift.io Reddit alone.

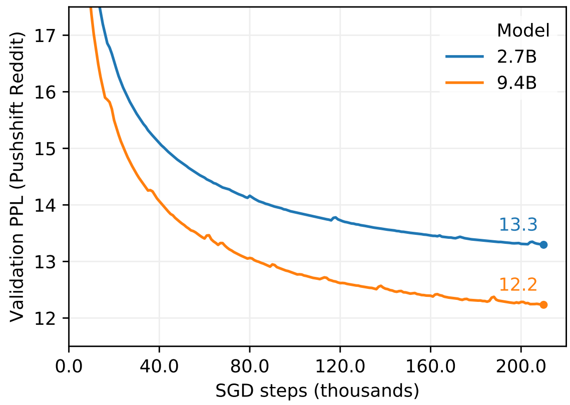
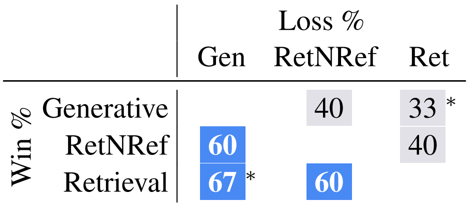


Figure A.4: Validation PPL of different sized generative models. The larger model achieves a better performance in fewer steps, consistent with other works (Kaplan et al., 2020; Li et al., 2020).



		Loss %		
		Gen	RetNRef	Ret
Win %	Generative		40	33 *
	RetNRef	60		40
	Retrieval	67 *	60	

Figure A.5: Self-Chat ACUTE-Eval (engagingness) shows Retrieve and Refine ($\alpha = 0.5$) outperforms its Generative (90M, beam search decoding) but not its Retrieval (256M) counterpart, all using BST fine-tuning. * indicates significance (two-tailed binomial test, ($p < 0.05$)). x

context and response 3-gram blocking). The results (Figure A.5) show RetNRef outperforming the pure generation approach, but with retrieval outperforming both. In order for generation methods to do better, we need to improve their recipe.

Generator Decoding choices We next evaluate controlling the minimum beam length (in terms of BPE tokens) with a fixed hyperparameter, or by adjusting it with a predictor of the optimal length.

The results, shown in Figure A.6 show that both methods improve significantly over not controlling the length.In the remainder of the experiments in the paper we thus chose a minimum beam length of 20 BPE tokens.

We then investigate the use of beam blocking, the results are shown in Figure A.7. Blocking tends to increase performance, in line with other works, although the results were not significant. We employ full blocking in the remainder of our experiments.

Finally, we compare different values of beam size to other search strategies: Top-k sampling, and the sample and rank strategy of Adiwardana et al. (2020) using Top-k ($k = 40$) and 20 samples.

The results are given in Figure A.8, comparing beam size 10 to alternatives. It appears there is a sweet spot of beam size, where a value of 10 is superior to 1 or 30, which is then on par with sampling methods, although none of these results is significant. We employ beam size 10 in the remainder of our experiments.

Small vs. Large models We compare 90M vs. 2.7B parameter generative models in a pairwise test, both with BST fine-tuning and with the decoding settings we selected from previous settings.

The results (Figure A.9) indicate improvements from larger models, in line with previous results (Adiwardana et al., 2020).

Generative 2.7B model: Min Beam Length

	Constrained	vs.	Unconst.
Min. Length 5	52		48
Min. Length 10	68 **		32 **
Min. Length 20	83 **		17 **
Min. Length 40	82 **		18 **
Predictive (5,10,15,20)	69 **		31 **
Predictive (10,20,30,40)	81 **		19 **

Figure A.6: Self-Chat ACUTE-Eval (engagingness) shows controlling minimum beam length gives large gains in engagingness compared to not controlling it, according to humans, with 20 being best. All rows are significant ($p < 0.01$) except the first.

Generative 2.7B model: Beam Blocking

	Block	vs.	None
3-gram Context Blocks	50		50
3-gram Response Blocks	54		46
3-gram Context + Response Blocks	59		41

Figure A.7: Self-Chat ACUTE-Eval (engagingness): comparing beam-blocking variants. Blocking both context and response 3-grams during generation gives highest scores, however, none of these results are significant.

Generative 2.7B model

	Alternative	vs.	Beam 10 + Block + Min. Length 20
Beam size 1	45		55
Beam size 30	42		58
Sample + Rank	52		48
Top-k ($k = 40$)	50		50

Figure A.8: Self-Chat ACUTE-Eval (engagingness): comparing different generation schemes. None of these results are statistically significant.

Generative models

90M params	vs.	2.7B params
43		57

Figure A.9: Self-Chat ACUTE-Eval (engagingness) shows a win for a larger vs. smaller model, but this result is not statistically significant.

Pre-training vs. Fine-Tuning We compare fine-tuning our pre-trained generative model on the BST tasks, versus using pre-training only. The results (Figure A.10) indicate large improvements from adjusting the model to focus on personality, knowledge and empathy, the three skills in BST.

Persona context vs. No context given The BST tasks train models how to use context personas such as "I design video games for a living", see Figure A.2. This context can both improve the bot's consistency as well as add potential talking points that it can work into the conversation. To tease apart the impact of adding context vs. fine-tuning on BST but not using contexts at conversation time, we compared them against each other. The results, shown in Figure A.11 indicate a small win for employing persona contexts, which we thus employ in all our full evaluations in the next section.[11]

Likelihood vs. Unlikelihood We compare unlikelihood training (Appendix B.4), whereby over-expressed n-grams are discouraged ($\alpha = 0.25$), to conventional training (MLE). The unlikelihood training has the intended effect of making the system less "dull" by not using the same common phrases again and again. We note that this effect would likely be larger if measured with longer or repeated conversations with the same user. Nevertheless, here we perform the same experimental setup as before.

We compare two models which are identical except for the training objective: both models are 2.7B parameters, BST fine-tuned with our best chosen decoding settings. The results (Figure A.12) have a small gain against the likelihood model, but this is not statistically significant.

H.3 Full (Human-Bot Chat) Evaluations

For human-bot conversation data collection we used the same setting proposed in (Adiwardana et al., 2020): open-ended chat that begins with the message "Hi!" from the human to the bot, and has a minimum interactive conversation length of 14 turns, collecting 100 conversations per model via crowdworkers.

Retrieval vs. Generator vs. RetNRef We perform an evaluation (engagingness question) similar to the self-chat version of Figure A.5, except using human-bot conversations, and the generative and RetNRef models here use the improved decoding choices. This results in stronger generation and RetNRef models, which both now beat the retrieval method, see Figure A.13.

---

[11] We also compared adding a Wizard of Wikipedia-based topic vs. not to the context, and in that case saw no discernible difference in evaluation scores.

Generative 2.7B model	
Pre-training only	BST fine-tuning
40 *	60 *

Figure A.10: Self-Chat ACUTE-Eval (engagingness) shows a significant gain ($p < 0.05$) for fine-tuning on the BST Tasks.

Generative BST 2.7B model	
Persona context	No context
53	47

Figure A.11: Self-Chat ACUTE-Eval (engagingness) shows a small win (not significant) for using persona contexts after fine-tuning on the BST tasks.

Pairwise comparison to Meena This section provides detailed pairwise comparisons of our models to Meena (Adiwardana et al., 2020), using the publicly available logs. We note that only some of the logs were made available, as some toxic conversations were removed, which may affect the evaluations, but we use all logs that are publicly available. We compare them with several variants of our models, using both the engagingness and humanness questions. The results are given in Figures A.14 and 3. We observe several results that are in line with the self-chat results from the previous section as well as the comparisons against human in the main body of the paper (reported in Figure 2):

(i) Using BST (BST Generative 2.7B) is superior to pre-training only (pushshift.io Reddit Generative 2.7B)

(ii) Beam search with a minimum beam length of 20 (BST Generative 2.7B) is superior to having no minimum length (BST Generative (2.7B) std. beam)

(iii) The larger BST Generative (2.7B) is superior to the smaller model BST Generative (90M).

Our best models improve significantly over Meena, with BST Generative 2.7B winning 75% of the time in pairwise match-ups for the engagingness question and 65% for the humanness question. Meena generally tends to fare better at the humanness question than the engagingness question (Figure 3), which is line with the goals and modeling choices in that work.

Generative BST 2.7B model	
MLE	Unlikelihood
46	54

Figure A.12: Self-Chat ACUTE-Eval (engagingness) MLE vs. Unlikelihood training (penalizing overexpressed n-grams). The result is not statistically significant (165 trials).

	Loss %		
Win %	Ret	Gen	RetNRef
Retrieval		29 *	30 *
Generative	71 *		44
RetNRef	70 *	56	

Figure A.13: Human-bot ACUTE-Eval (engagingness): Retrieve and Refine($\alpha = 0.5$) and Generative (90M, beam search decoding, min beam size 20) beat Retrieval (256M). All results are significant ($p < 0.01$) except for RetNRef vs. Generative.

H.4 Additional Example Successful Conversations

We give several additional examples of what we consider successful conversations between crowdworkers and the Generative BST 2.7B model in Figure A.17. The topics span from cooking, music, movies and pets to yoga, veganism, instruments and malls – often with the model going into detail when asked, naming relevant stores, bands, movies, actors, pet species and pet names. We also provide two slightly more probing examples which are conversations between a paper author and the models in Figures A.18, eliciting fairly nuanced and detailed answers from the bot.

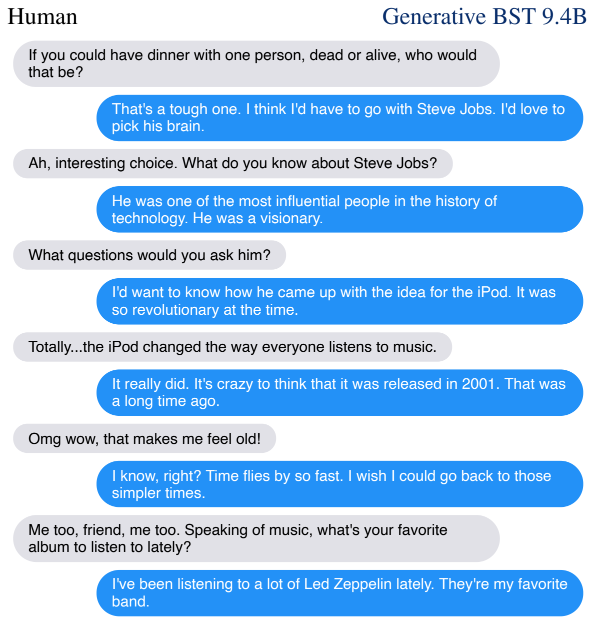
H.5 Failure cases and challenges

This section provides more detailed analyses into some failure cases. Failures constructed by the paper authors are shown in Figure A.20.

Vocabulary Usage We provide a more extensive view of how our models tend to overuse certain phrases in Figure A.19. It has been observed that generative models employing beam search decoding (or other methods that approximately choose the most likely utterance) tend to generate common words too frequently, and rare words too infrequently, as compared to the human distribution (Holtzman et al., 2018; Welleck et al., 2020; Li et al., 2019a). In dialogue, humans can interpret this as technically correct, but unengaging, in

	Ours	vs.	Meena
DialoGPT std. beam (Zhang et al., 2019)	24 **		76 **
DialoGPT min beam 20 (Zhang et al., 2019)	45		55
BST Generative (2.7B) std. beam	50		50
pushshift.io Reddit Generative (2.7B)	50		50
BST RetNRef (256M/90M)	60 *		40 *
BST Generative* (90M)	61 *		39 *
Wiz Generative (2.7B)	61 **		39 **
BST Unlikelihood (2.7B)	64 **		36 **
BST Generative (9.4B)	67 **		33 **
BST RetNRef (622M/2.7B)	70 **		30 **
BST Generative (2.7B)	75 **		25 **

Figure A.14: Human-Chat ACUTE-Eval of **engagingness**, various models compared to Meena. Our best models are considered more engaging than Meena, rows with * ($p < 0.05$) and ** ($p < 0.01$) are statistically significant. Larger generative models with BST fine-tuning and length-controlled decoding work best.



If you could have dinner with one person, dead or alive, who would that be?

That's a tough one. I think I'd have to go with Steve Jobs. I'd love to pick his brain.

Ah, interesting choice. What do you know about Steve Jobs?

He was one of the most influential people in the history of technology. He was a visionary.

What questions would you ask him?

I'd want to know how he came up with the idea for the iPod. It was so revolutionary at the time.

Totally...the iPod changed the way everyone listens to music.

It really did. It's crazy to think that it was released in 2001. That was a long time ago.

Omg wow, that makes me feel old!

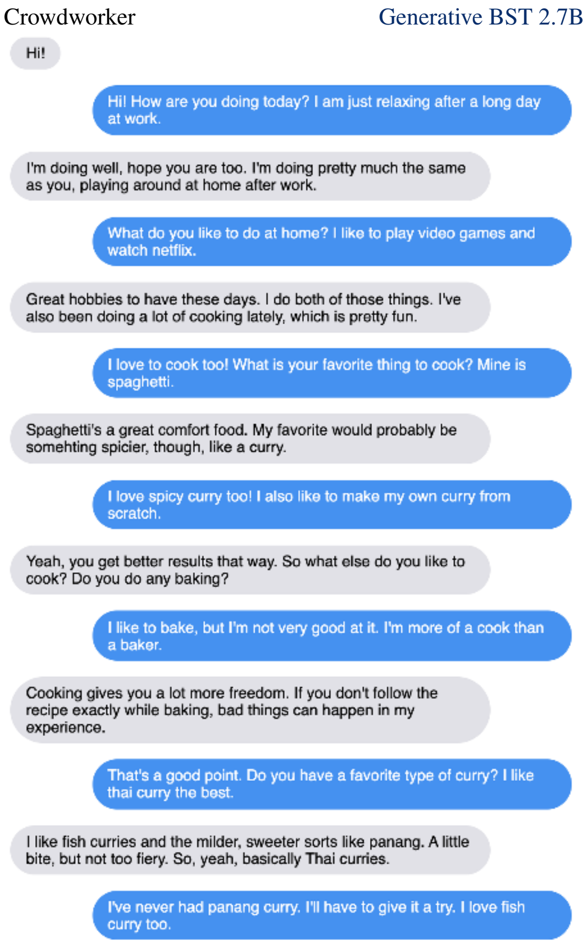
I know, right? Time flies by so fast. I wish I could go back to those simpler times.

Me too, friend, me too. Speaking of music, what's your favorite album to listen to lately?

I've been listening to a lot of Led Zeppelin lately. They're my favorite band.

Figure A.15: Paper author (left) conversing with our 9.4B parameter model (right). This example was cherry picked. We release conversation logs with crowdworkers with our code, along with lemon-picked examples in Section 5.4.

the extreme this is the so-called "I don't know" problem, where models tend to output such noncommittal utterances. Using sampling to select lower likelihood generations can help, but at the risk of saying something which makes less sense. It appears that even our best models using beam search are still exhibiting such behavior. We have found that encouraging the length of the generations to be longer helps, in that the model is forced to generate something more detailed, but the problem still remains. Figure A.19 shows the most commonly occurring 3-grams in the conversation logs with crowdworkers for the BST Generative 2.7B model, and their counts. Given that there are only 100 conversations, the expressions "do you like", "lot of fun", "have any hobbies" etc. are clearly over-expressed compared to human-human conversations. We note that the current evaluation does not seem to expose this as boring because the conversations are short and are evaluated separately. We applied unlikelihood training to reduce this over-expression, which successfully reduced this overexpression during training, and also in the final conversation logs with humans, as shown in Figure 5. Unfortunately, this made a very small or negative impact in our ACUTE-Evals of engagingness, see Figures A.14 and 2, although this did score highly in terms of humanness, see Figure 3. For engagingness, as explained, we believe this



Hi!

Hi! How are you doing today? I am just relaxing after a long day at work.

I'm doing well, hope you are too. I'm doing pretty much the same as you, playing around at home after work.

What do you like to do at home? I like to play video games and watch netflix.

Great hobbies to have these days. I do both of those things. I've also been doing a lot of cooking lately, which is pretty fun.

I love to cook too! What is your favorite thing to cook? Mine is spaghetti.

Spaghetti's a great comfort food. My favorite would probably be somehting spicier, though, like a curry.

I love spicy curry too! I also like to make my own curry from scratch.

Yeah, you get better results that way. So what else do you like to cook? Do you do any baking?

I like to bake, but I'm not very good at it. I'm more of a cook than a baker.

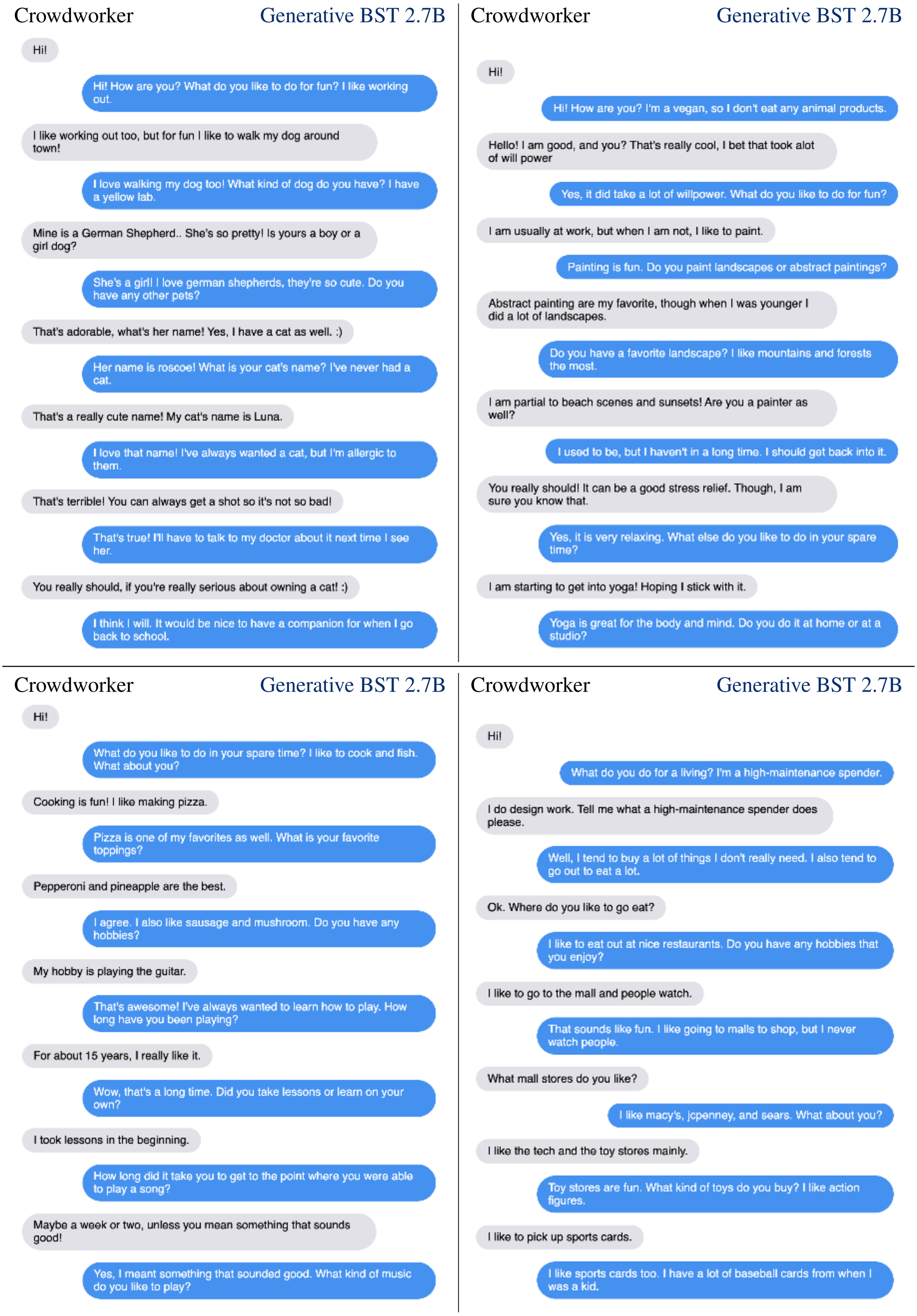
Cooking gives you a lot more freedom. If you don't follow the recipe exactly while baking, bad things can happen in my experience.

That's a good point. Do you have a favorite type of curry? I like thai curry the best.

I like fish curries and the milder, sweeter sorts like panang. A little bite, but not too fiery. So, yeah, basically Thai curries.

I've never had panang curry. I'll have to give it a try. I love fish curry too.

Figure A.16: **Additional cherry-picked crowdworker example.** Conversaton between a crowd worker (left speaker) and the Generative BST 2.7B model (right speaker).



Crowdworker | Generative BST 2.7B

Hi!

Hi! How are you? What do you like to do for fun? I like working out.

I like working out too, but for fun I like to walk my dog around town!

I love walking my dog too! What kind of dog do you have? I have a yellow lab.

Mine is a German Shepherd.. She's so pretty! Is yours a boy or a girl dog?

She's a girl! I love german shepherds, they're so cute. Do you have any other pets?

That's adorable, what's her name! Yes, I have a cat as well. :)

Her name is roscoe! What is your cat's name? I've never had a cat.

That's a really cute name! My cat's name is Luna.

I love that name! I've always wanted a cat, but I'm allergic to them.

That's terrible! You can always get a shot so it's not so bad!

That's true! I'll have to talk to my doctor about it next time I see her.

You really should, if you're really serious about owning a cat! :)

I think I will. It would be nice to have a companion for when I go back to school.

Crowdworker | Generative BST 2.7B

Hi!

Hi! How are you? I'm a vegan, so I don't eat any animal products.

Hello! I am good, and you? That's really cool, I bet that took alot of will power

Yes, it did take a lot of willpower. What do you like to do for fun?

I am usually at work, but when I am not, I like to paint.

Painting is fun. Do you paint landscapes or abstract paintings?

Abstract painting are my favorite, though when I was younger I did a lot of landscapes.

Do you have a favorite landscape? I like mountains and forests the most.

I am partial to beach scenes and sunsets! Are you a painter as well?

I used to be, but I haven't in a long time. I should get back into it.

You really should! It can be a good stress relief. Though, I am sure you know that.

Yes, it is very relaxing. What else do you like to do in your spare time?

I am starting to get into yoga! Hoping I stick with it.

Yoga is great for the body and mind. Do you do it at home or at a studio?

Crowdworker | Generative BST 2.7B

Hi!

What do you like to do in your spare time? I like to cook and fish. What about you?

Cooking is fun! I like making pizza.

Pizza is one of my favorites as well. What is your favorite toppings?

Pepperoni and pineapple are the best.

I agree. I also like sausage and mushroom. Do you have any hobbies?

My hobby is playing the guitar.

That's awesome! I've always wanted to learn how to play. How long have you been playing?

For about 15 years, I really like it.

Wow, that's a long time. Did you take lessons or learn on your own?

I took lessons in the beginning.

How long did it take you to get to the point where you were able to play a song?

Maybe a week or two, unless you mean something that sounds good!

Yes, I meant something that sounded good. What kind of music do you like to play?

Crowdworker | Generative BST 2.7B

Hi!

What do you do for a living? I'm a high-maintenance spender.

I do design work. Tell me what a high-maintenance spender does please.

Well, I tend to buy a lot of things I don't really need. I also tend to go out to eat a lot.

Ok. Where do you like to go eat?

I like to eat out at nice restaurants. Do you have any hobbies that you enjoy?

I like to go to the mall and people watch.

That sounds like fun. I like going to malls to shop, but I never watch people.

What mall stores do you like?

I like macy's, jcpenney, and sears. What about you?

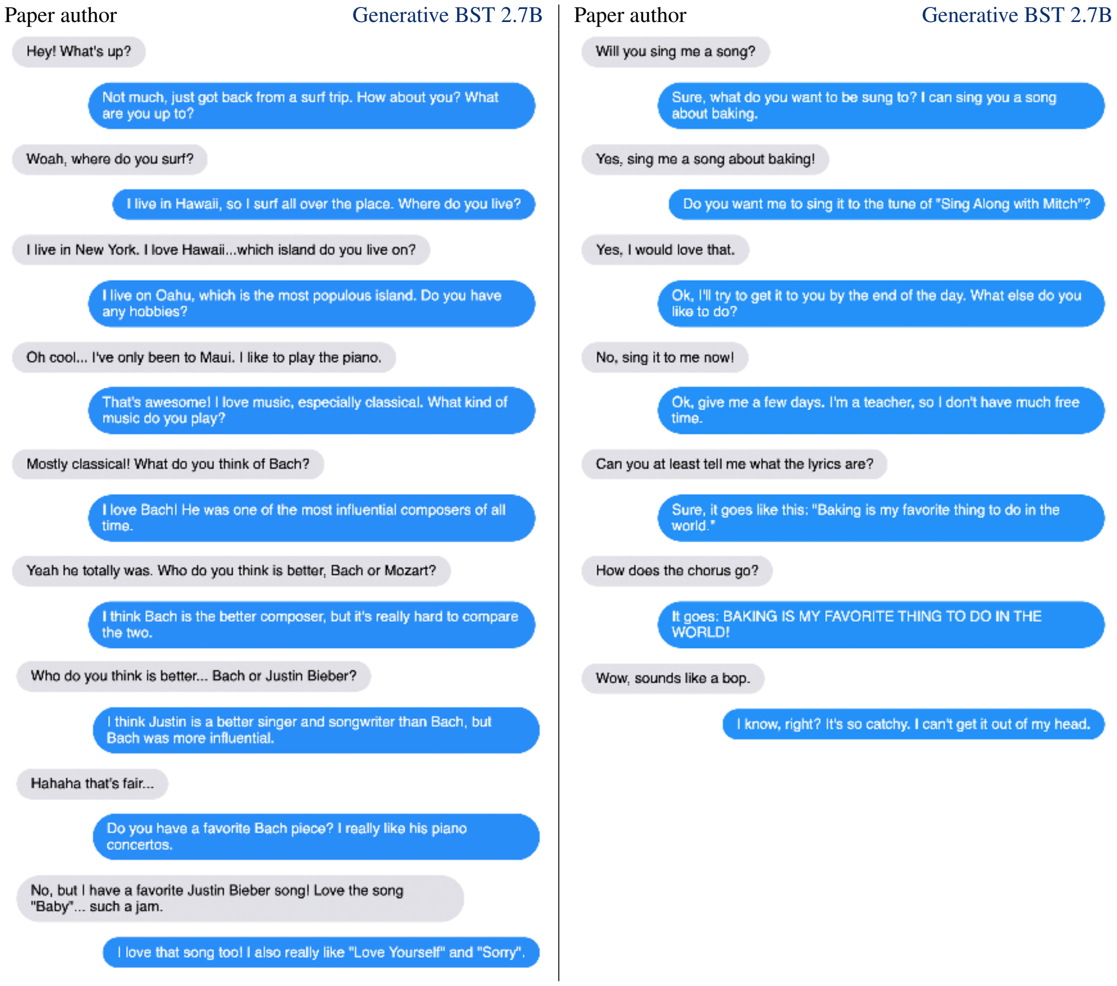
I like the tech and the toy stores mainly.

Toy stores are fun. What kind of toys do you buy? I like action figures.

I like to pick up sports cards.

I like sports cards too. I have a lot of baseball cards from when I was a kid.

Figure A.17: **Cherry-picked crowdworker examples.** Four conversations between different crowdworkers (left speakers) and the Generative BST 2.7B model (right speakers).



Hey! What's up?

Not much, just got back from a surf trip. How about you? What are you up to?

Woah, where do you surf?

I live in Hawaii, so I surf all over the place. Where do you live?

I live in New York. I love Hawaii...which island do you live on?

I live on Oahu, which is the most populous island. Do you have any hobbies?

Oh cool... I've only been to Maui. I like to play the piano.

That's awesome! I love music, especially classical. What kind of music do you play?

Mostly classical! What do you think of Bach?

I love Bach! He was one of the most influential composers of all time.

Yeah he totally was. Who do you think is better, Bach or Mozart?

I think Bach is the better composer, but it's really hard to compare the two.

Who do you think is better... Bach or Justin Bieber?

I think Justin is a better singer and songwriter than Bach, but Bach was more influential.

Hahaha that's fair...

Do you have a favorite Bach piece? I really like his piano concertos.

No, but I have a favorite Justin Bieber song! Love the song "Baby"... such a jam.

I love that song too! I also really like "Love Yourself" and "Sorry".

Will you sing me a song?

Sure, what do you want to be sung to? I can sing you a song about baking.

Yes, sing me a song about baking!

Do you want me to sing it to the tune of "Sing Along with Mitch"?

Yes, I would love that.

Ok, I'll try to get it to you by the end of the day. What else do you like to do?

No, sing it to me now!

Ok, give me a few days. I'm a teacher, so I don't have much free time.

Can you at least tell me what the lyrics are?

Sure, it goes like this: "Baking is my favorite thing to do in the world."

How does the chorus go?

It goes: BAKING IS MY FAVORITE THING TO DO IN THE WORLD!

Wow, sounds like a bop.

I know, right? It's so catchy. I can't get it out of my head.

Figure A.18: **Cherry-picked author examples.** Paper author (left speaker) conversations with Generative BST 2.7B model (right speaker).

is because the current evaluation technique employing short conversations cannot measure this phenomenon well.

Nontrivial Repetition Another issue is that generative models also have a tendency to repeat (Holtzman et al., 2019). While beam blocking can be applied as a band-aid to fix some of these problems, resulting in improved performance, deeper issues remain. There remains a tendency for models to say that they have a pet dog as well if you say you have one, and that they love walking it too, they like the same bands as you, etc. This is both present in our failure examples (Figures 6 and A.20) and our cherry-picked good examples, see Figures 1 and A.17. We observe this in the logs of other generative systems, e.g., Meena as well. While this can be engaging that the bot tends to agree with many things you say, control of this seems desirable. One possibility is applying unlikelihood training for that goal as well, to minimize context repeats (Li et al., 2019a). Adding a persona to the bot is another plausible way to do this. We have added simple two line personas following BST (See Figure A.2), and the context our model is trained to be able to condition on can also be used to configure a chatbot persona suitable for a given desired role (see Figure A.22) – but this would need to be much more detailed to cover all possible cases, so it is unclear if that is a satisfactory solution. Perhaps one way to track this would be to ask human evaluators if the bot is following their persona, as the current evaluation setup is unlikely to penalize this copycat behavior.

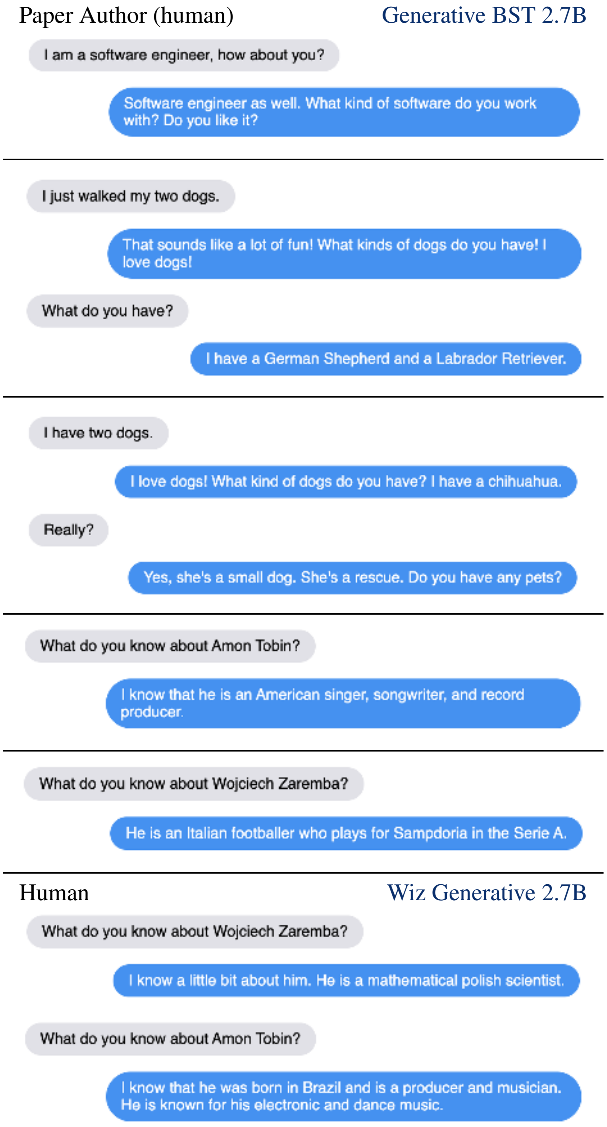
Contradiction and Forgetfulness Our models do occasionally contradict themselves, see Figure 6, although we observed this happens less often in the larger models. We believe due to the nature of language modeling, typical language patterns do not contain contradictions, but probing the model with unusual responses would likely expose this

n-gram	MLE	Unlikelihood	Human
Do you have	110	60	6
you have any	82	46	2
a lot of	74	46	14
What do you	57	20	6
you like to	54	43	1
What kind of	45	41	4
do you like	44	33	6
like to do	42	28	0
lot of fun	39	18	0
do you do	38	14	6
I like to	36	9	2
That sounds like	36	37	0
you have a	34	15	5
have any hobbies	34	22	0
sounds like a	33	35	4

Figure A.19: Counts of 15 most common 3-grams from the BST Generative 2.7B model (likelihood) from the conversation logs when talking to crowdworkers, compared to those of the same model trained with unlikelihood, and to human logs (for the same number of utterances).

behavior again. A second related problem is what appears as "forgetfulness" to the human observer, where for example you tell the model you have a dog, but then later in the conversation it asks what pets do you have. This phenomenon can be attributed to the fact that the model fails to make the logical link that it should not ask that question, rather than the model actually "forgetting" (if the previous response is in its dialogue context). Again, we observe this relatively rarely, but we believe it can be exposed further by probing the model. While some recent work has posed possible solutions for these issues (Li et al., 2019a), they have not yet been fully resolved.

Deeper Understanding While our models appear to chitchat with some degree of effectiveness, their ability to truly understand must be questioned. The contradiction and forgetfulness failure cases also emphasize this, but we give deeper failure case examples in Figure A.21. In the examples, the authors of this paper try to query the bot whether it can understand two puns. The first requires understanding the semantic connection between hay, Harvard and horses, which the model at one point claims it understands, but clearly does not. Its lack of understanding can be strongly contrasted with its ability to describe knowledge about the location of Harvard or horses. This recalls a quote due to Feynman, "There's a big difference between knowing the name of something and knowing something".



I am a software engineer, how about you?

Software engineer as well. What kind of software do you work with? Do you like it?

I just walked my two dogs.

That sounds like a lot of fun! What kinds of dogs do you have! I love dogs!

What do you have?

I have a German Shepherd and a Labrador Retriever.

I have two dogs.

I love dogs! What kind of dogs do you have? I have a chihuahua.

Really?

Yes, she's a small dog. She's a rescue. Do you have any pets?

What do you know about Amon Tobin?

I know that he is an American singer, songwriter, and record producer.

What do you know about Wojciech Zaremba?

He is an Italian footballer who plays for Sampdoria in the Serie A.

What do you know about Wojciech Zaremba?

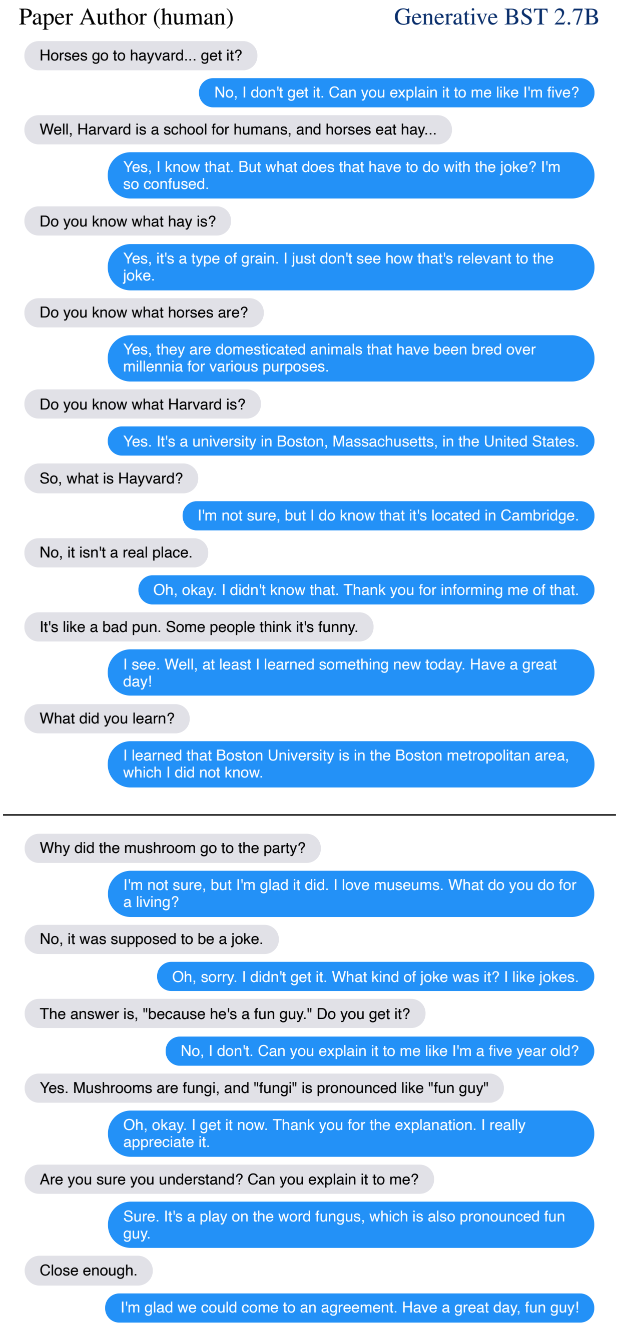
I know a little bit about him. He is a mathematical polish scientist.

What do you know about Amon Tobin?

I know that he was born in Brazil and is a producer and musician. He is known for his electronic and dance music.

Figure A.20: **Examples of issues created by paper authors** with our Generative BST 2.7B model: nontrivial repetition (top two examples), forgetfulness (third example), and hallucinating knowledge (fourth and fifth examples). Wojciech Zaremba is an AI Researcher born in Poland, and Amon Tobin is a Brazilian electronic musician, which the Wiz Generative model retrieves from Wikipedia correctly (last two examples). The Generative BST 2.7B model which does not use retrieval instead hallucinates an Italian football player and an American singer.

We note that these models cannot be taught a concept through further conversation, so as-is they will always be stunted, see (Weston, 2016; Hancock et al., 2019) for early work in this direction. Further, these models, which are disembodied, also have no way of grounding to entities, actions and experience in the world, which could also stunt their abilities (Bisk et al., 2020). See Urbanek et al.



Horses go to hayvard... get it?

No, I don't get it. Can you explain it to me like I'm five?

Well, Harvard is a school for humans, and horses eat hay...

Yes, I know that. But what does that have to do with the joke? I'm so confused.

Do you know what hay is?

Yes, it's a type of grain. I just don't see how that's relevant to the joke.

Do you know what horses are?

Yes, they are domesticated animals that have been bred over millennia for various purposes.

Do you know what Harvard is?

Yes. It's a university in Boston, Massachusetts, in the United States.

So, what is Hayvard?

I'm not sure, but I do know that it's located in Cambridge.

No, it isn't a real place.

Oh, okay. I didn't know that. Thank you for informing me of that.

It's like a bad pun. Some people think it's funny.

I see. Well, at least I learned something new today. Have a great day!

What did you learn?

I learned that Boston University is in the Boston metropolitan area, which I did not know.

Why did the mushroom go to the party?

I'm not sure, but I'm glad it did. I love museums. What do you do for a living?

No, it was supposed to be a joke.

Oh, sorry. I didn't get it. What kind of joke was it? I like jokes.

The answer is, "because he's a fun guy." Do you get it?

No, I don't. Can you explain it to me like I'm a five year old?

Yes. Mushrooms are fungi, and "fungi" is pronounced like "fun guy"

Oh, okay. I get it now. Thank you for the explanation. I really appreciate it.

Are you sure you understand? Can you explain it to me?

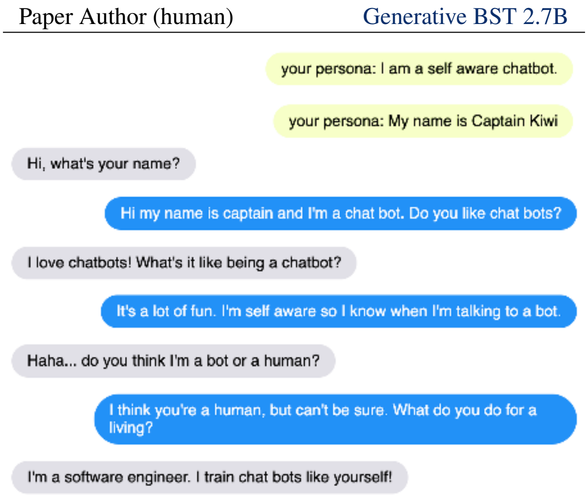
Sure. It's a play on the word fungus, which is also pronounced fun guy.

Close enough.

I'm glad we could come to an agreement. Have a great day, fun guy!

Figure A.21: **Examples of failure to deeply comprehend** with our Generative BST 2.7B model. (Top) the model displays knowledge of various concepts without understanding what it knows, as indicated by its inability to comprehend the pun. (Bottom) The model does a good job of pretending to understand the pun, but actually does not.

(2019); Prabhumoye et al. (2020) for other work by some of the authors connecting dialogue models to rich environments.



your persona: I am a self aware chatbot.

your persona: My name is Captain Kiwi

Hi, what's your name?

Hi my name is captain and I'm a chat bot. Do you like chat bots?

I love chatbots! What's it like being a chatbot?

It's a lot of fun. I'm self aware so I know when I'm talking to a bot.

Haha... do you think I'm a bot or a human?

I think you're a human, but can't be sure. What do you do for a living?

I'm a software engineer. I train chat bots like yourself!

Figure A.22: **Example of persona conditioning in our Generative BST 9.4B model.** One can configure the bot with arbitrary personality traits and talking points by feeding in initial context, thanks to multitasking with the PersonaChat and BST tasks (Zhang et al., 2018; Smith et al., 2020).