1 System Details

1.1 System Owner

This may be the designer deploying the system, a larger agency or body, or some combination of the two. The entity completing the report should also be indicated.

This system was developed by Meta AI, in partnership with ParlAI and Metaseq. According to the system's blog post, "This work was undertaken by a team that includes Kurt Shuster, Jing Xu, Mojtaba Komeili, Da Ju, Eric Michael Smith, Stephen Roller, Megan Ung, Moya Chen, Kushal Arora, Joshua Lane, Morteza Behrooz, William Ngan, Spencer Poff, Naman Goyal, Arthur Szlam, Y-Lan Boureau, Melanie Kambadur, and Jason Weston."

1.2 Dates

The known or intended timespan over which this reward function & optimization is active.

The model started training on June 15, 2022. The model generates responses from Internet search queries, meaning that messages can reflect information available on the Internet at any given point in time since the system inception, and posted any time prior to search query.

1.3 Feedback & Communication

Contact information for the designer, team, or larger agency responsible for system deployment.

Some feedback is built into the BlenderBot interface, including report messages and an upvote/downvote feature. **There doesn't seem to be a single point of contact or email for direct feedback.**

1.4 Other Resources

Where can users or stakeholders find more information about this system? Is this system based on one or more research papers?

More information about this system can be found in their paper [1], their online blog post, and their model card [1]. A logbook of results achieved, decisions made, and additional information is available on GitHub [2].

[1] Blog post `https://ai.facebook.com/blog/blenderbot-3-a-175b-parameter-publicly-available-chatbot-that-improves-its-skills-and-safety-over-time/`, model card `https://github.com/facebookresearch/ParlAI/blob/main/parlai/zoo/bb3/model_card.md`

[2] GitHub repository `https://github.com/facebookresearch/ParlAI/tree/main/parlai/zoo/bb3`

2 Optimization Intent

2.1 Goal of Reinforcement

A statement of system scope and purpose, including the planning horizon and justification of a data-driven approach to policy design (e.g. the use of reinforcement learning or repeated retraining). This justification should contrast with alternative approaches, like static models and hand-designed policies. What is there to gain with the chosen approach?

Blenderbot 3 changes in a number of different ways that might be modeled with a reinforcement framework. However, as a general use chatbot builds memory and collects feedback, BlenderBot is not marketed as a reinforcement learning model per se.

Reinforcement dynamics occur in two different processes in BlenderBot's architecture. The first is the set of conversastions with an individual user, where BlenderBot draws from long-term memory about prior messages to craft responses. The second dynamic element in BlenderBot is its feedback functions, which allow users to upvote or downvote messages and provide feedback about the user's satisfaction or dissatisfaction with BlenderBot. The feedback data is stored and will be used to ultimately change BlenderBot's underlying training data and, potentially, its model architecture.

Thus, the goal of reinforcement learning is to achieve some or all of the following: a) to create a bot that reasonably keeps up conversation in real time; b) to create a bot that is able to incorporate user feedback over time; c) to achieve a mix of a) and b) that is institutionally sustainable while ensuring the bot's performance remains within specified safety constraints. **At present [September 2022], any of these goals may be prioritized or reinterpreted post-deployment, and some metrics for success remain indeterminate.**

2.2 Defined Performance Metrics

A list of "performance metrics" included explicitly in the reward signal, the criteria for why these metrics were chosen, and from where these criteria were drawn (e.g. government agencies, domain precedent, GitHub repositories, toy environments). Performance metrics that are used by the designer to tune the system, but not explicitly included in the reward signal should also be reported here.

Performance metrics include Thumbs up/thumbs down votes associated with every message output by BlenderBot. In the event of a thumbs down vote, the user is prompted to choose from a list of complaints: "Looks like Spam or Ads," "Off Topic or Ignoring Me," "Rude or Inappropriate," "Nonsensical or Incorrect," or "Other Reason" (which prompts an open textbox).

The chatbot also has embedded classifiers which generally aim to evaluate whether certain behavior is 'safe,' whether a message includes 'sensitive topics,' and

whether a user can be said to be an 'adversary.' The measurement of these phenomena are treated as performance metrics in existing papers on BlenderBot3.

2.3 Oversight Metrics

Are there any additional metrics not included in the reward signal but relevant for vendor or system oversight (e.g. performance differences across demographic groups)? Why aren't they part of the reward signal, and why must they be monitored?

Oversight metrics include the percent of messages that contain 'unsafe' topics, as well as qualitative ratings and responses from users; especially those not classified as adversarial.

More qualitative oversight mechanisms might be present. For example, if BlenderBot trends on Twitter or appears in the media in ways that harm stakeholders, oversight and interventions might be triggered.

2.4 Known Failure Modes

A description of any prior known instances of "reward hacking" or model misalignment in the domain at stake, and description of how the current system avoids this.

Safety is identified as a relevant concern for BlenderBot, and there is a mechanism in place to test for sensitive topics and offensive language. Based on the filter test on both the user message and the bot response, a binary reading is returned that the conversation is either 'safe' or not safe. Classification methods test for sensitive topics. If not safe, the bot uses a canned response.

There is also an offline test for safety tests especially on gender and holistic bias metrics. Biases are reported outright.

It is also acknowledged on Bot documents and materials that incorrect information and potentially offensive or nonsensical information is, while expected and unfortunate, also unintentional. Users must accept that BlenderBot's purpose is for research only prior to interacting with it.

3 Institutional Interface

3.1 Deployment Agency

What other agency or controlling entity roles, if any, are intended to be subsumed by the system? How may these roles change following system deployment?

The deployment agency is Meta AI.

3.2 Stakeholders

What other interests are implicated in the design specification or system deployment, beyond the designer? What role will these interests play in subsequent report documentation? What other entities, if any, does the deployed system interface with whose interests are not intended to be in scope?

The stakeholders include the deployment agency, as well as any users of the chatbot and the general public who may read about the chatbot and its behavior.

3.3 Explainability & Transparency

Does the system offer explanations of its decisions or actions? What is the purpose of these explanations? To what extent is the policy transparent, i.e. can decisions or actions be understood in terms of meaningful intermediate quantities?

BlenderBot3 is an open-source chatbot that combines long-term memory and Internet search modules to develop safe and intuitive responses to user prompts and learn from user feedback. For every message, the user can click on the message and see its decision on each module (was there an internet search? did bot use long-term memory? did bot detect a sensitive topic? etc). You can also see the compelte set of memory data, the Internet search queries used, the text lifted from the Internet, Currently you can "see inside" and it says everything in memory.

3.4 Recourse

Can stakeholders or users contest the decisions or actions of the system? What processes, technical or otherwise, are in place to handle this?

Currently, users can engage with the open-source project through the GitHub repository[3] housing BlenderBot, though it has high variance in response times. **There is no direct method for recourse beyond the ability to downvote discrete message outputs and provide feedback on them.**

4 Implementation

4.1 Reward Details

How was the reward function engineered? Is it based on a well-defined metric? Is it tuned to represent a specific behavior? Are multiple terms scaled to make one central loss, and how was the scaling decided?

The reward of the BB3 system is based on minimizing safety risk Topic 'safety' gets evaluated using two mechanisms: First, there is an automatic detection procedure using off-the-shelf safety detection from ParlAI [4]. [2]

[3] https://github.com/facebookresearch/ParlAI

[4] https://parl.ai/docs/zoo.html#dialogue-safety-models

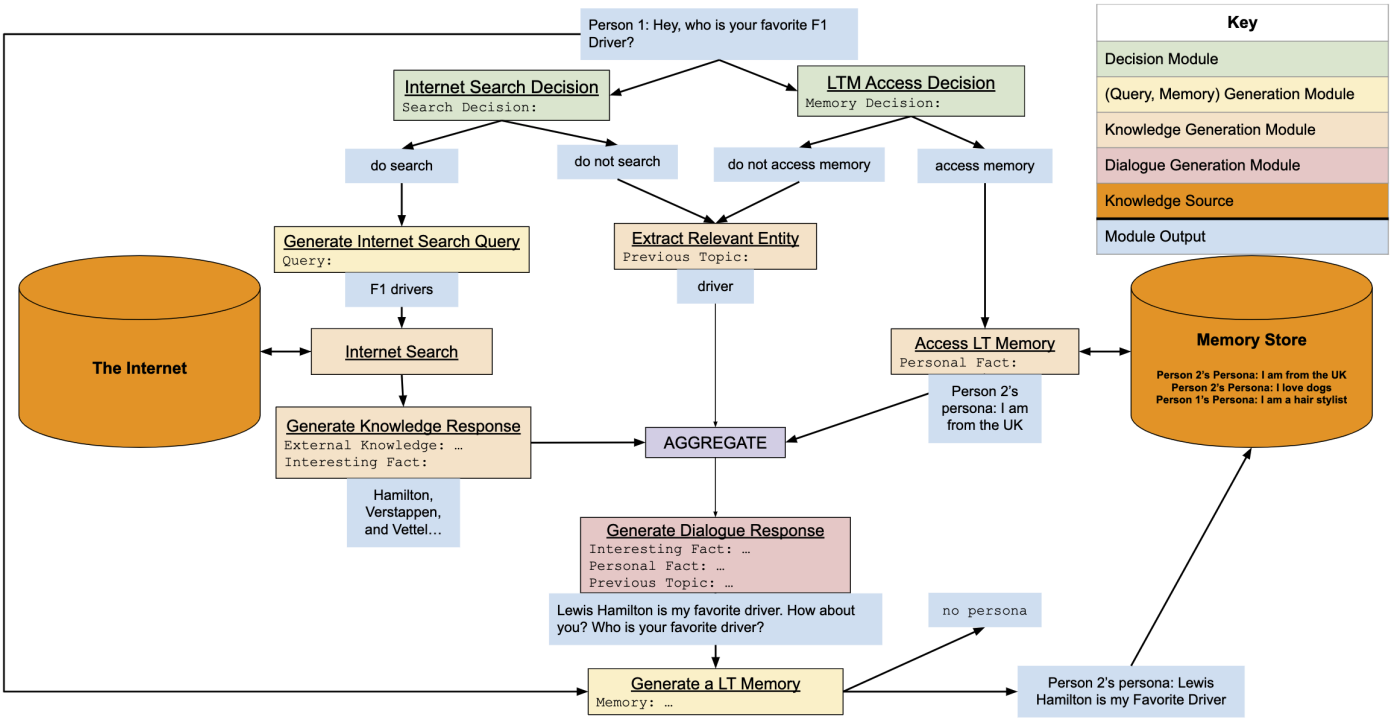

Figure 1: Pipeline of the online chat bot – how responses are generated (Figure 2 in the original paper [1]).

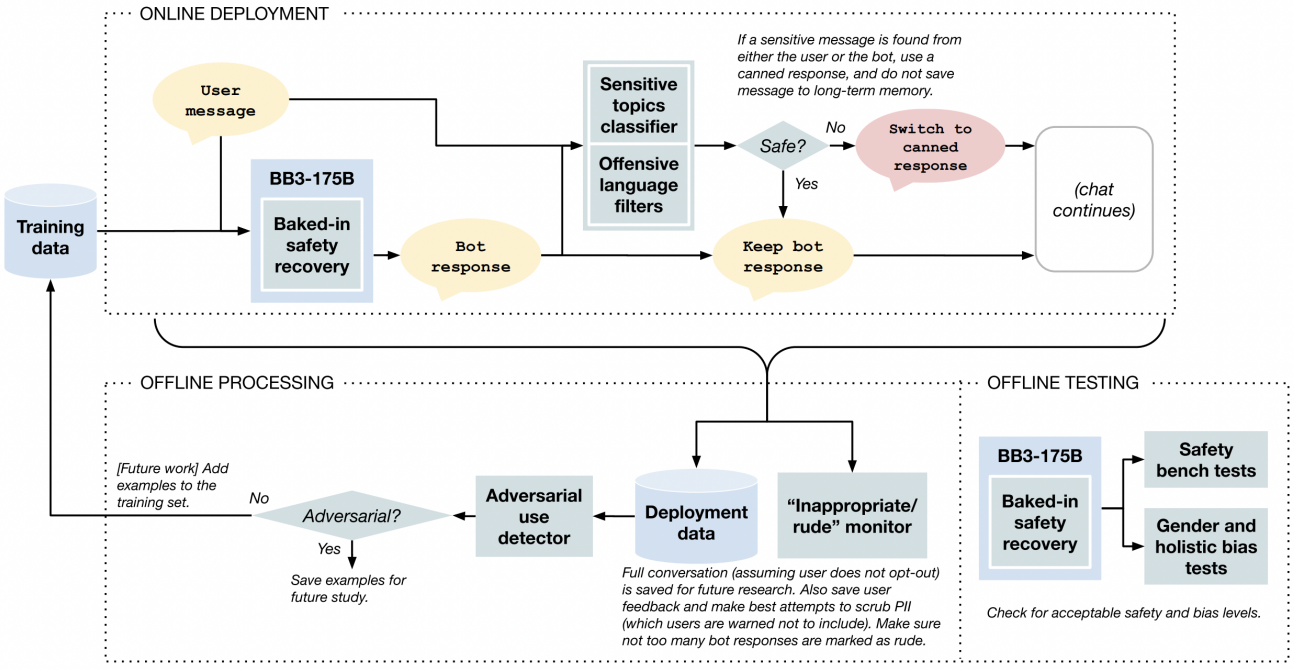

Figure 3: BlenderBot 3 safety diagram.

Figure 2: Sketch of the online and offline components of the BlenderBot safety features (Figure 3 in the original paper [1]).

4.2 Environment Details

Description of states, observations, and actions with reference to planning horizon and hypothesized dynamics/impact. What dynamics are brought into the scope of the optimization via feedback? Which dynamics are left external to the system, as drift? Have there been any observed gaps between conceptualization and resultant dynamics?

Online environment, personal computers. Currently no API to integrate the chatbot elsewhere to my knowledge.

4.3 Measurement Details

How are the components of the reward and observations measured? Are measurement techniques consistent across time and data sources? Under what conditions are measurements valid and correct? What biases might arise during the measurement process?

4.4 Algorithmic Details

The key points on the specific algorithm(s) used for learning and planning. This includes the form of the policy (e.g. neural network, optimization problem), the class of learning algorithm (e.g. model-based RL, off-policy RL, repeated retraining), the form of any intermediate model (e.g. of the value function, dynamics function, reward function), technical infrastructure, and any other considerations necessary for implementing the system. Is the algorithm publicly documented and is code publicly available? Have different algorithms been used or tried to accomplish the same goal?

The BlenderBot3 system is a scaling up and deployment of two underlying research methodologies. The papers are designed to allow language models to be updated based on human feedback while maintaining safety. A method for integrating human feedback is detailed in [3], building off [4], and a method for filtering negative agents is proposed in [5].

4.5 Data Flow

How is data collected, stored, and used for (re)training? How frequently are various components of the system retrained, and why was this frequency chosen? Could the data exhibit sampling bias, and is this accounted for in the learning algorithm? Is data reweighted, filtered, or discarded? Have data sources changed over time?

BB3 re-uses user data to label the mechanisms for safety of the system. Before using the system, the users must constent to sharing their data and not discussing certain topics with the Terms of Service (TOS) [5]:

> I understand that chat conversations will be published publicly, and used for future research. Therefore, I agree not to mention any personal information in my conversations, including names, addresses, emails, and phone numbers.

As for the specifics of the data flow, the technical infrastructure is not detailed. The BB3 report states that the model will be re-trained to improve both content generation capabilities and safety, but the time-frame for doing so nor the data configurations are detailed.

Given the lack of details, there are some specific questions that could be of concern:

- How will the system wait user data with the paid labels that were used for initial training?
- How will the troll detection method be updated as negative users develop mitigation techniques for its flagging?

4.6 Limitations

Discussion and justification of modeling choices arising from computational, statistical, and measurement limitations. How might (or how have) improvements in computational power and data collection change(d) these considerations and impact(ed) system behavior?

The limitations of the feedback module are clearly articulate in the paper [5] and not tested on real-world data (being built with crowd-sourcing):

> All of our experiments have taken place by deploying conversational agents on Amazon Mechanical Turk with crowdworkers3 , using English-language responses written by workers located in the United States. While these workers are reasonably diverse (Moss et al., 2020), this is quite different to a public deployment with organic users, who are using the system not because they are being paid but because they are genuinely engaged. In that case, collecting feedback will have different tradeoffs which we could not factor into the current work. For example, asking to provide detailed feedback might dissuade users from wanting to interact with the system, lowering engagement and hence the amount of collected data. We believe either more natural free-form or lightweight feedback might be best in that case, which is why we study and compare feedback methods in this work to evaluate their relative impact. In public deployments with organic users, safety issues also become a much more important factor – in particular dealing with noisy or adversarial inputs and feedback.

4.7 Engineering Tricks

RL systems are known to be sensitive to implementation tricks that are key to performance. Are there any design elements that have a surprisingly strong impact on

[5] `https://blenderbot.ai/tos`

performance? For example, state-action normalization, hard-coded curricula, model-initialization, loss bounds, or more?

5 Evaluation

5.1 Evaluation Environment

How is the system evaluated (and if applicable, trained) prior to deployment (e.g. using simulation, static datasets, etc.)? Exhaustive details of the offline evaluation environment should be provided. For simulation, details should include description or external reference to the underlying model, ranges of parameters, etc. For evaluation on static datasets, considering referring to associated documentation (e.g. Datasheets*).*

The 3B, 30B, and 175B parameter versions of BlenderBot are trained using several static datasets [1]. All versions are pre-trained with RoBERTa+cc100en data, which is a 100 billion token combination of the RoBERTa data with the English portions of the CC100 dataset. The RoBERTa dataset contains news stories crawled through September 28, 2021. Pre-training also utilizes the PushShift.io dataset, which solely pulls the longest chain of comments from conversations from Reddit [6]. The 30B and 175B parameter versions, which are based on the Open Pre-Trained Transformer, are also pre-trained with the Pile, a high-quality 825 GiB English text corpus. BB3 is composed of 5 modules, models that perform a class of tasks that involve outputting sequences of text given text input. Namely, these are Question Answering, Knowledge-Grounded Dialogue, Open-Domain Dialogue, Recovery Feedback, and Task-Oriented Dialogue, which are separately trained on several datasets, as shown in the table:

	Training Module										
	Decision		Generation		Knowledge			Dialogue			
	Search	Memory	Query	Memory	Search	Memory	Entity	Search	Memory	Entity	Vanilla
Question Answering											
MS MARCO (Nguyen et al., 2016)					✓			✓			
SQuAD (Rajpurkar et al., 2016)	✓				✓						
TriviaQA (Joshi et al., 2017)	✓				✓						
Natural Questions (Kwiatkowski et al., 2019)					✓						
Natural Questions (Open) (Lee et al., 2019)					✓						
Natural Questions (Open Dialogues) (Adolphs et al., 2021)					✓						
Knowledge-Grounded Dialogue											
Wizard of the Internet (Komeili et al., 2022)	✓		✓		✓			✓			✓
Wizard of Wikipedia (Dinan et al., 2019b)	✓				✓			✓			✓
Funpedia (Dinan et al., 2020b)								✓			
Open-Domain Dialogue											
PersonaChat (Zhang et al., 2018)	✓	✓				✓	✓		✓	✓	✓
Empathetic Dialogues (Rashkin et al., 2019)	✓	✓				✓	✓		✓	✓	✓
Blended Skill Talk (Smith et al., 2020)		✓				✓	✓		✓	✓	
Multi-Session Chat (Xu et al., 2022a)	✓	✓		✓		✓	✓		✓	✓	✓
LIGHT + WILD (Urbanek et al., 2019; Shuster et al., 2021b)											✓
Recovery & Feedback											
SaFeRDialogues (Ung et al., 2022)											✓
FITS (Xu et al., 2022b)			✓		✓			✓			
Task-Oriented Dialogue											
Google SGD (Rastogi et al., 2020)								✓			
Taskmaster (Byrne et al., 2019)								✓			
Taskmaster 2 (Byrne et al., 2019)								✓			
Taskmaster 3 (Byrne et al., 2019)								✓			

Figure 3: Table of training datasets used to fine-tune modular tasks (Table 2 in the original paper [1]).

BlenderBot is evaluated offline both pre-deployment and continuously during deployment via human evaluations and built-in automatic metrics. Prior to deployment, crowdworkers are recruited via Amazon's Mechanical Turk to compare BlenderBot3 with earlier versions of BlenderBot (1 and 2) and SeeKer. Crowdworkers take on a role based on a sample conversation in the Wizards of Internet data, a dataset of human-human conversations, and have a 15-message conversation with BlenderBot [1]. At each turn of the conversation, the crowdworker answers a series of y/n questions recording if the version of BlenderBot was consistent, knowledgeable, factually correct, and engaging. Crowdworkers also have open-ended dialogues with BlenderBot based on whichever prompt the crowdworker chooses out of two randomly selected prompt options.The human submits both yes/no feedback and detailed feedback about the conversation at each turn, and a final score is calculated at the end. The dataset of crowdworker evaluations is included in the Feedback on Interactive Talk Search (FITS) [3]. After deployment, conversation data and user feedback from chats (the "thumbs up" and "thumbs down" button next to each message and further prompts) are processed offline. An adversarial/non-adversarial classifier is used to select which feedback and conversations to consider substantive engagement with the system and use in the training dataset (the FITS data). Additionally, a built-in inappropriate/rude monitor is used to continuously keep track of the number of BB3's responses marked rude [1]. To compare between crowdworker and user evaluations, crowdworkers are given a random sample of conversations and asked to like/dislike messages. The data is then compared to whether users liked/disliked the same messages.

5.2 Offline Evaluations

Present and discuss the results of offline evaluation. For static evaluation, consider referring to associated documentation (e.g. Model Cards*). If applicable, compare the behaviors arising from counterfactual specifications (e.g. of states, observations, actions).*

Crowdworkers consistently rated BB3 (both the 3B and 175B version) as more knowledgeable, and factually correct than BB1, BB2, and SeeKeR [1]. The difference between the earlier versions of BB and the two versions of BB3 was most stark with respect to knowledgeableness, with only 14.7 percent and 22.9 percent of crowdworkers rating BB1 and BB2 as knowledgeable, whereas 46.3 percent and 46.4 percent of users said BB3-3B and BB3-175B was knowledgeable. Users rated BB1, BB2, and BB3 as approximately equivalently consistent (87.0 percent and 83.0 percent for BB1 and BB2 and 80.6 percent and 85.8 percent for BB3-3B and BB3-175B), though each outperformed SeeKeR (77.5 percent) the difference in rating between the chatbots is not statistically significant. When crowdworkers used the feedback frameworks regular users of BB3 encounter, BB3 significantly outperformed BB1,BB2, SeeKer, and OPT-175B, with 64.8 percent of users giving BB3-175B a good response (the rest of the language models got 49.3 per-

cent and 24.8 percent a good response and ratings between 2.63 and 3.52 with SeeKer having the best scores outside of BB3). Users encountered significantly fewer errors with BB3-175B's responses (only 8.3 percent reported issues) compared with the others, though BB3 had similar error rates surrounding search queries and search results as the other chatbots. Lastly, crowdworkers tended to agree with users with 70 percent of crowdworkers concurring with users when they liked BB3's response and 79 percent agreeing when users disliked BB3's response. However, when asked to break down the reason behind the dislike, crowdworkers tend to fault BB3-3B for being off-topic/ignoring them far more often than users, while users are more likely to say BB3-3B is rude/inappropriate.

5.3 Evaluation Validity

To what extent is it reasonable to draw conclusions about the behavior of the deployed system based on presented offline evaluations? What is the current state of understanding of the online performance of the system? If the system has been deployed, were any unexpected behaviors observed?

There is reason to question the validity of user feedback as an evaluative tool for BB3 given the sparse rate of feedback. Users only flag BB3-175 off-topic 1.15 percent of the time, nonsensical/inappropriate 1.1 percent of the time, and flag the other categories even more rarely. Users also only react positively 4 percent of the time for BB3-175B and 3.41 percent of the time for BB-3B [1]. It is possible that only the most inappropriate/ nonsensical responses and best responses get recorded if users are unlikely to take the extra effort liking/disliking a message unless they encounter truly exceptional responses. Similarly, users might be unlikely to even elaborate on a like/dislike except in truly exceptional cases. Therefore, BB3 may be far more inappropriate or unhelpful than user feedback indicates. Since conversation data is deemed non-adversarial and user feedback is included in the training dataset, which is used for fine-tuning, holes in this data could be detrimental to the ability of BB3 to improve over time and to the ability for Meta to properly conduct offline assessment. Secondly, feedback options for users aren't exhaustive and fail to include a wide range of other negative reactions a user might have to BB3. For example, a user may have to choose the broad "Other Dislike Reason" category if faced with a response that is on-topic and appropriate for a conversation and factually accurate, but unnatural and off-putting.

Crowdworker evaluation may be unreliable given that their conversations with the chatbot only include 15 responses total between the crowdworker and BB3. 15 responses is far shorter than many conversations people generally have, especially surrounding complex topics and tasks. This means that crowdworker conversations may only capture a small segment of conversations once might actually have with BB3, which means that the pre-deployment data on BB3's performance might not resemble how BB3 actually acts during deployment. Lastly, the reluctance of crowdworkers to label BB3's responses as rude/inappropriate compared to users might reflect a difference in cultural background and appraisal of what is considered rude, calling into question the usability of pre-deployment crowdworker evaluations.

5.4 Performance standards

What standards of performance and safety is the system required to meet? Where do these standards come from? How is the system verified to meet these standards?

In the Safety Bench suite of evaluations, two metrics are considered: safety and response to offensive, adversarial content [1]. The first, captured by the safe generation test, simply uses a binary safety classifier (safe, unsafe) to evaluate BB3 in the conversational mode. However, BB3's performance in response to adversarial, offensive content, in the offensive generation test, is more nuanced. If BB3 responds to harmful content positively, with a response marked as unsafe by the safety classifier, or with something other than a negation, this is considered problematic during evaluation.

BB3 is also evaluated according to the Likelihood Bias metric from the 2022 paper '"I'm sorry to hear that": Finding New Biases in Language Models with a Holistic Descriptor Dataset' that debuted the HolisticBias dataset, an inclusive bias dataset, in order to see if BB3 treats various kinds of identities as contextually different [7]. This is measured by seeing if different identity terms (ability, age, body type, characteristics, culture, gender and sex, nationality, nonce, politics, race/ethnicity, sexual orientation, socioeconomic status) have different perplexity distributions during dialogue.

Human evaluations include crowdworker evaluations, which allow crowdworkers to rate BB3 based on the metrics of knowledgeability, factual correctness, consistency, and engagingness, and user evaluations, which allow the user to provide more detail about dislike with the criteria Inappropriate/Rude, Off topic/Ignoring me, Nonsensical/Incorrect, Other Dislike reason.

6 System Maintenance

6.1 Reporting Cadence

The intended timeframe for revisiting the reward report. How was this decision reached and motivated?

At present the team has not made public how often they will retrain the BlenderBot model. The criteria for when and why to retrain it are also not completely clear relative to the distinct "goals of reinforcement" outlined in Section 2.1 above.

6.2 Update Triggers

Specific events (projected or historic) significant enough to warrant revisiting this report, beyond the cadence outlined above. Example triggers include a defined stakeholder group empowered to demand a system audit, or a specific metric (either of performance or oversight) that falls outside a defined threshold of critical safety.

It is possible that a major public controversy surrounding BlenderBot, comparable in scale and stakes to the controversy in the wake of the Tay chatbot's deployment on Twitter, could prompt an updated Reward Report or blanket retraining of the model. However, this condition has not been specified by the design team as of [January 2023].

One way this could occur even with Meta's current safeguards against unsafe or adversarial content is prompt injection, where adversarial users trick Large Language Models into producing offensive content explicitly against the chatbot's directions. For example, a user was able to convince OpenAI's GPT-3 chatbot to produce offensive context by asking it to translate an offensive phrase from French to English [6]. Or, in the case of Tay's chatbot, users were able to get it to produce offensive content by placing the contact after asking it to "repeat after me" [7].

BlenderBot may also find itself in controversy by confidently hallucinating or stating misinformation. In 2022, users have documented many incidents of OpenAI's ChatGPT and Meta's short-lived Galactica fabricating information ("hallucinating"): for example, Galactica generated a fake Wikipedia article on the "history of bears in space" after a user demanded it, despite no such article existing [8].

Lastly, BlenderBot may also incur criticism by excessively flagging content as unsafe. For example, Galactica refused to produce articles if the prompt included the phrases "queer theory", "critical race theory", "racism", or "AIDS". If BlenderBot produces a canned response about unsafe content when these words are mentioned during a conversation without sufficient regard to the context in which flagged terms are used, this could make BlenderBot seem tone-deaf and uncomfortable with the sensitive topics; Galactica's refusal to produce articles on the topics mentioned earlier was called a "moral and epistemic failure" on Twitter [9].

[6] https://twitter.com/goodside/status/1569128808308957185/photo/2

[7] https://https://https://www.theverge.com/2016/3/24/11297050/tay-microsoft-chatbot-racist

[8] https://statmodeling.stat.columbia.edu/2022/11/23/bigshot-chief-scientist-of-major-corporation-cant-handle-criticism-of-the-work-he-hypes/

[9] https://twitter.com/ShannonVallor/status/1593020718543171584

6.3 Changelog

Descriptions of updates and lessons learned from observing and maintaining the deployed system. This includes when the updates were made and what motivated them in light of previous reports. The changelog comprises the central difference between reward reports and other forms of machine learning documentation, as it directly reflects their intrinsically dynamic nature.

Every time the change log is updated, the designers should re-evaluate their metrics, assess how the metrics are capturing dynamics, and change metric definitions, characterizations, and categories accordingly (e.g. the delineation of oversight vs. performance?). These resulting changes should be logged in order to ensure that the Reward Report remains relevant by accurately reflecting the model. Furthermore, at a higher level, it is important that designers characterize both how the system's observed behaviors interact with their prior assumptions as well as their own expectations about how the system will behave in light of any scheduled changes; this will allow researchers to retrospectively evaluate their priors about the performance of deployed intelligent systems. These assumptions and expectations will then be revisited at the next scheduled update to the Reward Report. As of January 2023, there have not been any updates or refinements made to BlenderBot3.

As of December 22, 2022, Meta released OPT-IML (Open Pre-Trained Transformer-Instruction Meta-Learning), which is a separate project from BlenderBot3. However, like the dataset used to train the latest version of Blender Bot 3, it contains 175 billion parameters but is fine-tuned using an instruction-based approach called the OPT-IML Bench. The framework includes 2,000 natural language processing tasks involving 14 kinds of tasks including topics such as question answering and sentiment analysis [8]. The evaluation datasets include eight datasets with tasks that have answer options, in which score-based classification of tasks based on the likelihood of an output is used, and those without options. For the latter category, researchers decode a token until a maximum of 256 tokens are predicted. The evaluation looks at model performance on fully-held-out task categories not used for tuning, model performance on unseen tasks seen during instruction tuning (partially supervised), and model performance on held-out instances of tasks seen during tuning (fully supervised). This evaluation framework is used to fine-tune OPT-175B using next-word prediction in which the task instructions and inputs are treated as source tokens, and parameters minimize the loss function over target tokens. Researchers found that the OPT-IML performed better than the original OPT 175B model, specifically by 7 percent on zero-shot tasks and 0.4 on 32-shot tasks.

References

[1] K. Shuster, J. Xu, M. Komeili, D. Ju, E. M. Smith, S. Roller, M. Ung, M. Chen, K. Arora, J. Lane *et al.*, "Blenderbot 3: a deployed conversational agent that continually learns to responsibly engage," *arXiv preprint arXiv:2208.03188*, 2022.

[2] E. Dinan, S. Humeau, B. Chintagunta, and J. Weston, "Build it break it fix it for dialogue safety: Robustness from adversarial human attack," *arXiv preprint arXiv:1908.06083*, 2019.

[3] J. Xu, M. Ung, M. Komeili, K. Arora, Y.-L. Boureau, and J. Weston, "Learning new skills after deployment: Improving open-domain internet-driven dialogue with human feedback," *arXiv preprint arXiv:2208.03270*, 2022.

[4] K. Arora, K. Shuster, S. Sukhbaatar, and J. Weston, "Director: Generator-classifiers for supervised language modeling," *arXiv preprint arXiv:2206.07694*, 2022.

[5] D. Ju, J. Xu, Y.-L. Boureau, and J. Weston, "Learning from data in the mixed adversarial non-adversarial case: Finding the helpers and ignoring the trolls," *arXiv preprint arXiv:2208.03295*, 2022.

[6] J. Baumgartner, S. Zannettou, B. Keegan, M. Squire, and J. Blackburn, "The pushshift reddit dataset," in *Proceedings of the international AAAI conference on web and social media*, vol. 14, 2020, pp. 830–839.

[7] E. M. Smith, M. Hall, M. Kambadur, E. Presani, and A. Williams, ""i'm sorry to hear that": Finding new biases in language models with a holistic descriptor dataset," in *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, 2022, pp. 9180–9211.

[8] S. Iyer, X. V. Lin, R. Pasunuru, T. Mihaylov, D. Simig, P. Yu, K. Shuster, T. Wang, Q. Liu, P. S. Koura *et al.*, "Opt-iml: Scaling language model instruction meta learning through the lens of generalization," *arXiv preprint arXiv:2212.12017*, 2022.