KAUCUS: Knowledge Augmented User Simulators for Training Language Model Assistants

Kaustubh D. Dhole
Department of Computer Science
Emory University
Atlanta, USA
kdhole@emory.edu

Abstract

An effective multi-turn instruction-following assistant can be developed by creating a simulator that can generate useful interaction data. Apart from relying on its intrinsic weights, an ideal user simulator should also be able to bootstrap external knowledge rapidly in its raw form to simulate the multifarious diversity of text available over the internet. Previous user simulators generally lacked diversity, were mostly closed domain, and necessitated rigid schema making them inefficient to rapidly scale to incorporate external knowledge. In this regard, we introduce **Kaucus**, a **K**nowledge-**Au**gmented **U**ser **S**imulator framework, to outline a process of creating diverse user simulators, that can seamlessly exploit external knowledge as well as benefit downstream assistant model training. Through two GPT-J based simulators viz., a **Retrieval Augmented Simulator** and a **Summary Controlled Simulator** we generate diverse simulator-assistant interactions. Through reward and preference model-based evaluations, we find that these interactions serve as useful training data and create more helpful downstream assistants. We also find that incorporating knowledge through retrieval augmentation or summary control helps create better assistants.

1 Introduction

Significant advancements in Large Language Models (LLMs) have made them exceptionally adept in conversational applications like virtual assistants (Touvron et al., 2023; FitzGerald et al., 2022; OpenAI, 2023; Team et al., 2023). This proficiency is largely attributed to the notably parallelizable transformer architecture (Vaswani et al., 2017) enabling these models to utilize extensive pre-training datasets effectively (Raffel et al., 2019; Computer, 2023). To create effective assistants, LLMs are then further enhanced by learning from human interactions including popular paradigms such as

RLHF (Böhm et al., 2019; Ziegler et al., 2019; Ouyang et al., 2022a). Such conversational human alignment of assistants requires large amounts of interactive dialog data, both for training as well as testing.

However, interactive data collection is a manual and slow process, particularly (a) for covering a wide range of user behaviors as well as (b) for diverse adversarial and behavior testing.

These challenges can be mitigated by simulating user behaviors by automating the generation of interactive data, reducing both time and cost, while maintaining control over the interactions. Simulated interactions can be executed at a much faster pace than manual collection efforts, limited only by the speed of inference.

Yet, current user simulators lack diversity, are mostly closed domain, and require rigid schema for control or conversation grounding. The necessity of intermediate schema in the form of a knowledge base (Kim et al., 2023) or handcrafted rules (like user persona or specific behaviors) while being excellent drivers to ground conversations, make it hard to develop scalable simulators – that can utilize natural text freely available on the internet and rapidly create corresponding assistant models. A simulator should be able to exploit external knowledge rapidly and also be controllable without a rigid schema. We argue that such a knowledge simulator can be helpful in two ways – It can seamlessly convert free-form text to useful training data without user intervention as well as provide a natural control to direct simulators for specific behaviors (Mille et al., 2021; Cheng et al., 2023).

Hence, in this work, we propose **Kaucus**, a **K**nowledge **Au**gmented **S**imulator Framework[1]. Through this framework, we demonstrate the usage of external sources of knowledge – viz. Retrieval Augmentation and Summary Control – for creating

---

[1] pronounced like Caucus derived from Algonquian cau'-cau'-as'u meaning 'adviser'
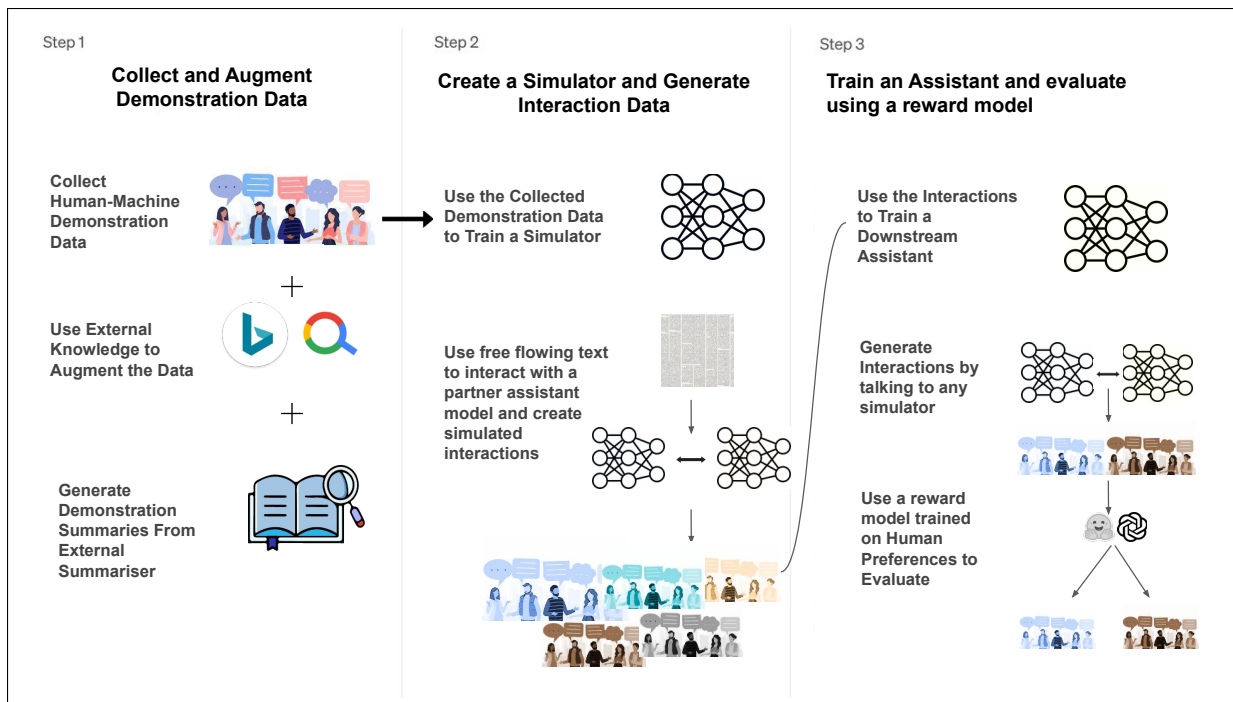
Figure 1: The complete three step framework of **Kaucus** – creating, utilizing and evaluating a user simulator.

user simulators that can incorporate free-flowing text and result in better assistant training.

The paper is organized as follows: In Section 2, we first discuss existing work related to user simulators. In Section 3, we define simulators and introduce **Kaucus**, through two knowledge simulators. We further describe the efficacy of each through training and evaluating downstream assistant models. Our retrieval augmented simulator, **SRAG** shows how retrieving relevant passages with a simple BM25 retriever can be used to improve intrinsic metrics as well as provide useful training data to train helpful assistants. We also introduce the summary-controlled setting, **SCTRL** to build scalable simulators to exploit freely available text and further measure their performance with and without retrieval.

2   Related Work

User simulators have been studied in various settings. Aher et al. (2023) create four simulators that elicit behavior to judge an assistant's fairness, rationality, grammaticality, and general knowledge, and then measure them qualitatively. Their simulators are models with different prompt templates. Training multi-agent interactions has been a popular choice in reinforcement learning. Horton (2023); Argyle et al. (2023) create simulations for economic purposes by endowing GPT3 (Brown et al.,

2020) with demographic characteristics and then get responses in various scenarios that match what is seen empirically. Irving et al. (2018) in AI safety has proposed using self-play and self-debate to train AI agents to pursue human goals and preferences. Two tasks in the collaborative benchmark, BIG-Bench (Srivastava et al., 2023) evaluate the model's ability for self-evaluation by simulating specific human professions. They make the models to act as lawyers, tutors[2], judges[3], students, etc. and then have separate model instances to evaluate the conversation. Each of the roles is invoked by user-specific prompts like "You are a lawyer" and a subsequent model-based evaluation is performed by prompting to seek numerical ratings.

Kreyssig et al. (2018)'s Neural User Simulations involve training encoder-decoder RNNs on dialogues between real users and a spoken dialogue system (SDS) in a restaurant domain and then using the trained simulator to train the policy of a reinforcement learning based SDS. They further use Schatzmann et al. (2005)'s cross-model evaluation to compare user simulators by training different policies with each simulator and testing it with other simulators. Gur et al. (2018) encode dialog history and a goal to generate user responses for task-oriented dialog. Kraus et al. (2023a); Li et al.

---

[2]BIG-Bench Self Evaluation Tutoring
[3]BIG-Bench Self Evaluation Courtroom

(2022b) prompt LLMs with task-oriented dialog data, such as goals, and perform intrinsic evaluation over the generated data to show the effectiveness of their approaches. Kim et al. (2023) generate conversations grounded on common sense by prompting InstructGPT with knowledge base triples. Their human evaluations show that oftentimes humans prefer model outputs against their human-written counterparts. Liu et al. (2023) leverage multiple user simulators to train task-oriented dialog systems. Faltings et al. (2023) utilize user simulators that offer edits to guide the model towards achieving a specified target text training them using Imitation Learning.

Other studies augment simulators with emotions (Lin et al., 2023) and trusting behaviours (Kraus et al., 2023b). For instance, Lin et al. (2023) simulate user emotions alongside user behavior based on the user goal, the dialogue history, and persona. Giabbanelli (2023) utilize GPT-based models for scientific simulations while Schaefer et al. (2023) explore LLMs to simulate biological systems.

With the popularity of large language models deployed in closed-source settings, bootstrapping training data from them has become useful. Taori et al. (2023) create downstream assistant models by training LLama-7B and 13B models (Touvron et al., 2023) on 52K single-turn instruction following demonstrations generated through self-instruct (Wang et al., 2023b) from text-davinci-003 (Brown et al., 2020). Bian et al. (2023) create a dialog corpus by extending the same to the multi-turn setting. Dai et al. (2022) show improved conversation retrieval by proposing a mechanism to convert Wikipedia passages to dialog.

On the other hand, retrieval augmentation has been the focus of many recent efforts (Schick et al., 2023; Zhang et al., 2023; Wang et al., 2023a; Li et al., 2022a) as it offers advantages such as up-to-date information access beyond an LLM's training dataset, incorporation of proprietary or domain-specific data at runtime, and enhanced factuality in outputs compared to standard LLMs. Studies have been performed by training RAG systems end-to-end (Guu et al., 2020; Lewis et al., 2020) as well as using retrieval in context for various tasks (Ram et al., 2023; Jiang et al., 2023; Gao et al., 2023; Dhole and Agichtein, 2024).

3 The Kaucus Framework

In this section, we introduce **Kaucus**, a 3-stage framework, and outline the process of creating knowledge-augmented simulators as shown in Figure 1. Our approach involves the following steps:

3.1 Data Collection and Augmentation

We start by gathering interaction data – essentially conversations between a user and a base assistant LLM, which will be later augmented to enrich the training process. For instance, the base LLM could take the form of closed-source instruct models such as OpenAI's GPT-4, Claude, or BingChat which are widely used for work.

3.2 Training a Language Model (LM) as a Simulator

The next step involves training a Language Model (LM) to act as a simulator. This LM can then serve as a conversation generator for data augmentation (Dhole et al., 2023) or be integrated into a pipeline that relies on conversation interactions, such as Reinforcement Learning from Human Feedback (RLHF) (Ziegler et al., 2019; Ouyang et al., 2022b). Our work focuses on the former.

3.3 Leveraging the User Simulator

Once the user simulator is trained, there are several methods to utilize for improving an assistant Language Model (LM). Our work resorts to data augmentation, which will be the focus of our second set of experiments. It involves using the user simulator to generate additional training data to enhance the assistant LM's performance.

3.4 Evaluation

To evaluate the effectiveness of the user simulator, we will employ both intrinsic and extrinsic metrics. Intrinsic metrics will be measured over the interactions with the simulator, assessing its performance in generating relevant and coherent responses. On the other hand, extrinsic metrics will be based on evaluating a downstream assistant model trained over these interactions, which will help us gauge the impact of the user simulator on overall assistant performance. We will describe the evaluation in detail in Section 5.

4 Methods and Experiments

We now specifically describe the two types of knowledge-augmented simulators, viz. Utterance

Human: Can you write a short introduction about the relevance of the term "monopsony" in economics? Please use examples related to potential monopsonies in the labour market and cite relevant research

Assistant: Monopsony refers to a market structure where there is only one buyer of a good or service. In the context of labor markets, a monopsony occurs when there is only one employer in a particular geographic area or industry, giving them significant power to set wages and working conditions.......

Human: What can be done at a regulatory level to ensure, in a monopsony, that the power over the employees isn't abused? List a few options and focus on the organs that should act.

Assistant: Yes, that's correct. Keeping ....

Human: ```A monopoly is distinguished from a monopsony, in which there is only one buyer of a product or service; a monopoly may also have monopsony control of a sector of a market.``` Can you write a short introduction about the relevance of the term "monopsony" in economics? Please use examples related to potential monopsonies in the labour market and cite relevant research

Assistant: Monopsony refers to a market structure where there is only one buyer of a good or service....

Human: ```The term may also be used to describe the restriction of certain access rights when an employee has changed roles within the organization. Onboarding and offboarding procedures must be documented to ensure compliance with regulatory requirements.``` What can be done at a regulatory level to ensure, in a monopsony, that the power....",

---

Summary: "Monopsony" in economics is a market structure where there is only one buyer of a good or service. The power over the employees isn't abused in a monopsony.

Human: Can you write a short introduction about the relevance of the term "monopsony" in economics? Please ...

Assistant: Monopsony refers to a market structure w..., when there is only one employer in a particular geographic area or industry, giving them significant power to set wages and working conditions.......

Human: What can be done at a regulatory level to ensure, in a monopsony, that the power over the employees isn't abused? ...
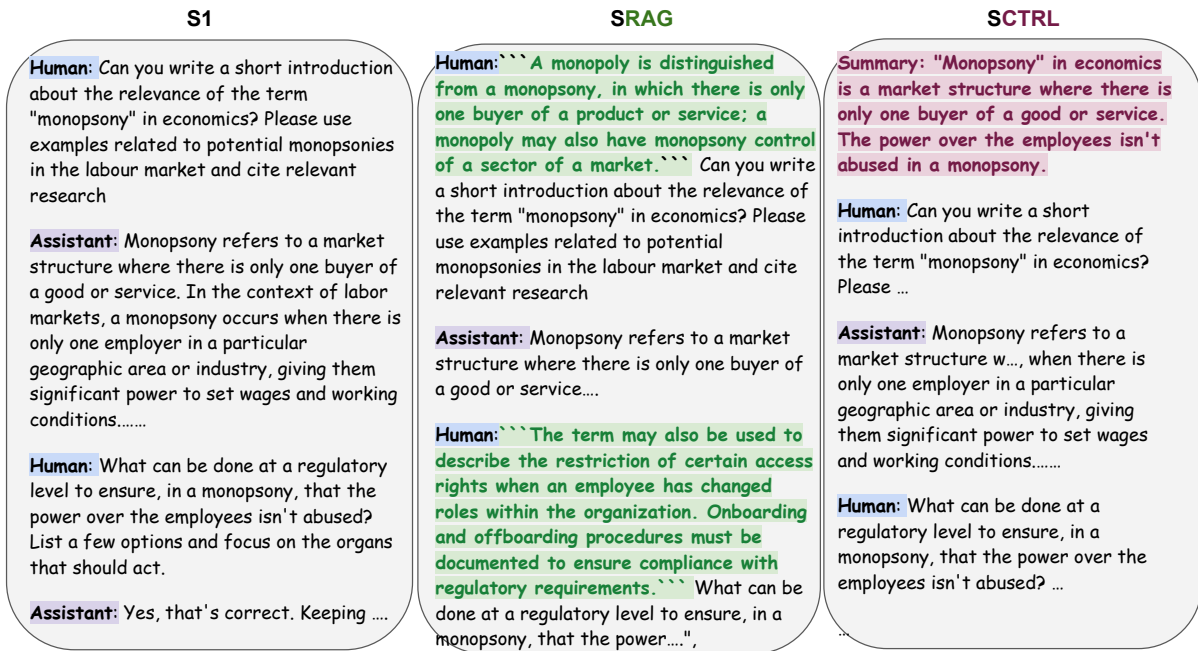
...

Figure 2: The format of the conversations used for training S1 (a vanilla simulator), SRAG (retrieved document shown in green), and SCTRL (summary shown in red).

Grounded Simulators (S1 and SRAG) and Summary Controlled Simulators (SCTRL).

4.1 Utterance Grounded Simulators

Here we train simulators with human-machine demonstration data by feeding models the conversation history to create simulators that can be triggered from a starting utterance. We create two simulators – S1 and SRAG by fine-tuning an unsupervised pre-trained GPJ-6B (Wang and Komatsuzaki, 2021) model. We describe the training process for both below:

4.1.1 S1

Simulator Trained on Anthropic and Open Assistant Conversations

- **Training Data**: For training S1, we use demonstration data available through Open Assistant's conversations (Köpf et al., 2023) and Anthropic's helpful splits (Bai et al., 2022).

- **Format**: The Simulator's training data consists of (context, human-response) pairs. For every "Human" utterance in all the conversations, we select all the previous utterances along with their speaker information and pass it as an input to the model. The input also consists of a "Human:" string at the end. The

associated human response is passed as the output.

4.1.2 SRAG

Retrieval Augmented Simulator Trained on Anthropic and Open Assistant Conversations with BM25 Retrieval on MSMarco

Simulators could benefit from the incorporation of external knowledge which can be helpful to steer the conversation, improve factuality and most importantly introduce variation. To test our hypothesis, we train the second simulator, SRAG by incorporating passages retrieved from an external retriever.

- **Training Data**: We augment the interactions used to train S1 with passage snippets from the MS-Marco dataset (Nguyen et al., 2016; Bajaj et al., 2016), which is a large-scale dataset of 8.8M passages popularly used for information retrieval and reading comprehension. Having been generated from real users' search queries, it provides a vast repository of documents collected on a plethora of topics over the web.

- **Format**: We use (context, human-response) pairs in the style of S1 with human turns annotated with retrieved MSMarco passages. Using the human

utterance as a query, we execute a BM25 retriever against an MSMarco Passage Index for every human turn. Each of the human utterances is then prepended with a retrieved passage as shown in Figure 2 in green. We use the MSMarco index provided by IRDatasets (MacAvaney et al., 2021) and the BM25 implementation provided by PyTerrier (Macdonald et al., 2021).

4.2 Summary Controlled Simulators

The previous utterance-grounded setting relies on a conversational utterance at inference time to initiate the interaction. While it can be easy to obtain such conversational utterances using existing conversational datasets, they can quickly become scarce and out-of-date. It would be of interest to be able to scale over vast amounts of free text available over the web. However, most of the web data exists in a non-conversational format unsuitable for direct incorporation in the training process.

SCTRL: In that regard, we introduce the training of **summary controlled simulators** that can utilize the conversational summary obtained from an external conversation summarizer during training. This can be potentially helpful in two ways – It can provide a mechanism for the simulator to attempt to seamlessly convert "free form text" to "interaction data" while also coming up with the "simulator trigger" by itself reducing our reliance on conversational corpora. As compared to a fixed schema or a knowledge base, it can provide a natural control to guide simulators for specific behaviors via natural language texts which are generally available in plenty as compared to their conversational or interactive counterparts.

- **Training Data**: To create the training data, we append a conversational summary generated from an external conversational summarizer, at the beginning of the conversation. Our objective is to force the simulator to be able to learn the association between the initial non-conversational text and the subsequent conversation. We choose an existing BART Summariser (Wolf et al., 2020) fine-tuned on various dialog and non-dialog summarisation datasets like DialogSum, AMI and XSUM.

- **Format**: We prepend the predicted summary at the start of the conversation as shown in Figure 2 in red.

We create the two summary-controlled counterparts of S1 and SRAG as SCTRL and SCTRL-RAG respectively.

SCTRL-RAG Summary Controlled Simulator Trained on Anthropic and Open Assistant Conversations with MSMarco BM25 Retrieval

We use a GPT-J-6B model RLHF fine-tuned on demonstration data as our base assistant model and our simulator. We use deepspeed (Rasley et al., 2020) to optimize training and train for 10 epochs on a learning rate of 10^{-6}.

5 Evaluation & Results

5.1 Intrinsic Metrics

We first seek to assess the "diversity" of the generated interactions. In assessing diversity, we utilize well-established reference-free lexical metrics viz. TTR, logTTR, RootTTR, HDD, and MTLD are based on type-token ratios and are quick to compute. The Measure of Textual Lexical Diversity (MTLD) is a prevalent and contemporary TTR metric that does not vary as a function of text length and explains textual information that similar lexical diversity approaches do not account for (McCarthy and Jarvis, 2010). It gauges the proportion of distinct word stems (types) to the overall word count (tokens). HDD is an alternative metric that captures additionally unique lexical information (McCarthy and Jarvis, 2010)[4].

We first generate 125 interactions by making each of the simulators interact with a fixed assistant model. The conversation is initialized with an existing Anthropic conversation in the case of S1 and SRAG and five more turns are generated (referred to as the augmented length). In SCTRL and SCTRL-RAG, 5 turns are generated from scratch from Anthropic's conversation summary. We present the results in Table 2. The metrics measure the lexical diversity only on the utterances generated via the simulator interaction (and not on the initial Anthropic conversation history that was fed to initiate the interaction). Across all metrics, incorporating a knowledge component, through retrieval augmentation (SRAG) or summary control (SCTRL) improves diversity. Incorporating both improves diversity across RootTTR and HDD metrics.

---

[4]Through a separate ancillary study, we also find that simulators trained on dialog data generate more diverse text as compared to pre-trained ones according to the above metrics.

Source	Type	Generated Interaction Data	Assistant
Human	–	Assistant Trained With Anthropic_8k	A0
S1	Without Knowledge	Simulated Anthropic_8k	A1
SRAG	With Retrieval Augmentation	Simulated Anthropic_8k + MSMarco	A1-RAG
S1-CTRL	With Summary Control	Simulated Anthropic_8k*10 summaries	A1-CTRL
S1-CTRL-RAG	Both	Simulated Anthropic_8k*10 summaries + MSMarco	A1-CTRL-RAG

Table 1: The sources of various simulated data used in **Kaucus** to train the corresponding assistants

Simulator	MTLD	Root TTR	LogTTR	HDD
S1	23.177	2.918	0.818	0.04
SRAG	**24.632**	**3.223**	**0.82**	**0.134**
SCTRL	**25.864**	**3.437**	**0.844**	**0.131**
SCTRL-RAG	22.761	2.976	0.766	0.278

Table 2: Lexical diversity metrics on 125 conversations of each simulator. The top-2 highly diverse simulators are the knowledge-based ones - SRAG and SCTRL on all metrics.

5.2 Extrinsic Metrics

Although the aforementioned metrics can assist in evaluating and comparing various user simulators as potential data augmenters and generators, it is crucial to determine if they benefit subsequent assistant models. The RLHF paradigm, by training reward models, has demonstrated assistants that are more helpful, honest, and less harmful providing a promising direction for aligning with human preferences. In this regard, we resort to the family of reward and preference models to measure how well assistant models trained using data produced from various simulators perform.

Training Downstream Assistant Models: For each simulator trained (S1, SRAG,..), we create a subsequent assistant model (A1, ARAG, ...) and use reward modeling to measure the helpfulness of each of the assistant models. To create training data for each of the assistant models, we first simulate interactions between the corresponding simulator model along a separately held-out assistant model.

For each utterance grounded simulator (S1 and SRAG), we use 8000 Anthropic conversations as triggers. Particularly, we utilize the complete Anthropic conversation as the starting history for both the simulator and the separately held-out assistant model and allow ten turns (5 pairs) of interactions to be generated. Using the simulator to generate longer contexts provides an opportunity to collect a larger number of (context, assistant-response) pairs for training the downstream assistant model.

For the retrieval augmented simulator, SRAG,

it is necessary to retrieve passages relevant to the ongoing conversation. We hence use the previous assistant response as a query to our MSMarco Passage Index before generating the simulator turn. The top-ranked passage via BM25 is then placed at the end of the input to SRAG.

For generating interactions from SCTRL, we need free-flowing text as the initial trigger. We generate 8000 conversations from conversation summaries of the Anthropic dataset. We use additional 9*8K passages from MS-Marco as initial triggers to act as implicit summaries.

After generating the conversations, we convert them into (context, assistant-response) pairs and use them as training data for predicting the assistant response given all the previous utterances. We call the subsequent assistant models A1, ARAG and ACTRL. The training details of each assistant model are described in Table 1.

Baseline: We additionally train an assistant model, A0 using raw 8000 conversations from Anthropic to act as appropriate baseline.

Test Set: For evaluation, we utilize 200 utterances from the test set of Anthropic's dataset.

FastChat Evaluation: FastChat (Zheng et al., 2023) is a platform for evaluating and serving LLMs. We resort to FastChat evaluation for prompting GPT-4 (OpenAI, 2023) for a comparative evaluation between two simulators. The process involves GPT-4 being input with two conversations, placed one after the other, along with an instruction to evaluate and generate a numerical score. We attribute a win, a loss, or a tie depending on whether the first (assistant model on the left in all the images) has a value greater, lesser, or equal to the second (one on the right).

SteamSHP Reward Model Evaluation SteamSHP-XL (Ethayarajh et al., 2022) is a preference model fine-tuned on top of an instruction-tuned model FLAN-T5-XL (Wei et al.; Longpre et al., 2023) to predict which response humans will find more helpful, given some context and two possible responses. On

being prompted the same context, the reward model setting compares the probabilities assigned independently to each model response to infer the preference label.

SteamSHP Preference Model Evaluation (Ethayarajh et al., 2022) Preference modeling, like the FastChat Evaluation, compares two model responses through a single inference pass, which can be used to compute the probability of the first one being better than the second.

To avoid any bias occurring through the order of two conversations, we also calculate the scores with the simulator order reversed in the prompt.

For each plot, the columns indicate the two assistant models being compared. The colors in blue for each row indicate when the evaluation system prefers the left-hand side model as compared to the right-hand side when compared against A0.
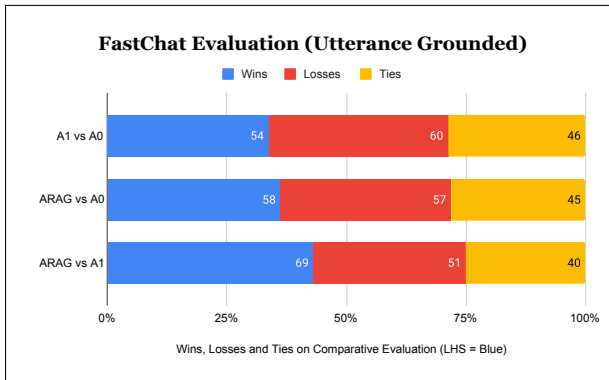
Figure 3: FastChat Evaluation of Assistants created from Utterance Grounded Simulators (A1 and ARAG) against baseline assistant (A0)

Effect of Simulator: We first compare A1 (i.e. the assistant trained on 8k interactions generated from S1) against A0 (i.e. the one trained without
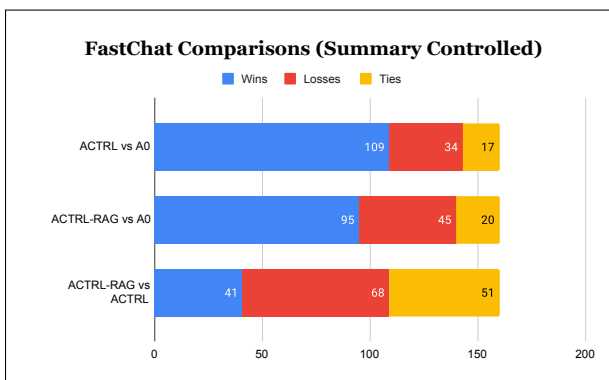
Figure 4: FastChat Evaluation of Assistants created from Summary Controlled Simulators (-CTRL) against baseline assistant (A0)
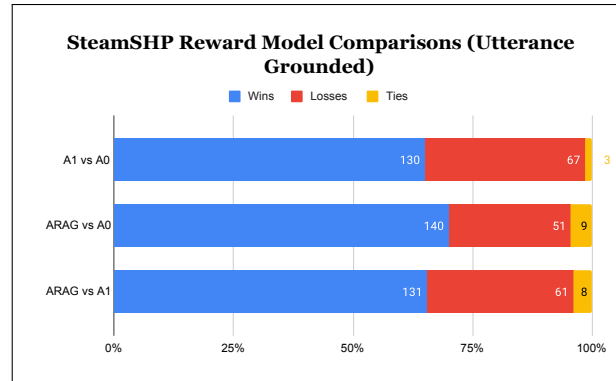
Figure 5: SteamSHP reward model Evaluation of Assistants created from Utterance Grounded against baseline assistant (A0)
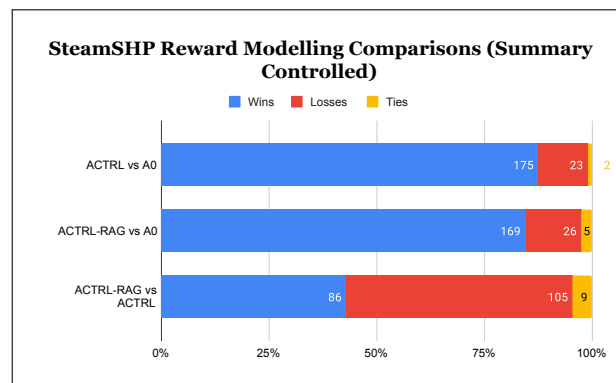

Figure 6: SteamSHP reward model Evaluation of Assistants created from Summary Controlled Simulators (-CTRL) against baseline assistant (A0)

the help of the simulator). A1 outperforms A0 in all three evaluations as seen on the first rows of Figures 3, 5 and 7. The results are more prominent in SteamSHP's evaluations. This shows that with the help of a simulator, we can generate more data and improve downstream assistant performance.

Effect of Retrieval Augmentation: We then compare whether an assistant model ARAG, trained from retrieval augmented data benefits training. With the retrieval augmented simulator, downstream performance across all metrics is improved. ARAG's interactions are preferred more often as compared to A0 as well as A1 as seen in the 2nd and 3rd rows of Figures 3, 5 and 7.

Effect of Summary Control: The assistants ACTRL and ACTRL-RAG trained from the summary-controlled simulators are more often preferred across all the evaluations as shown in the first two rows of Figures 4, 6 and 8. However, the non-retrieval counterpart ACTRL is more often preferred as compared to the retrieval counterpart.
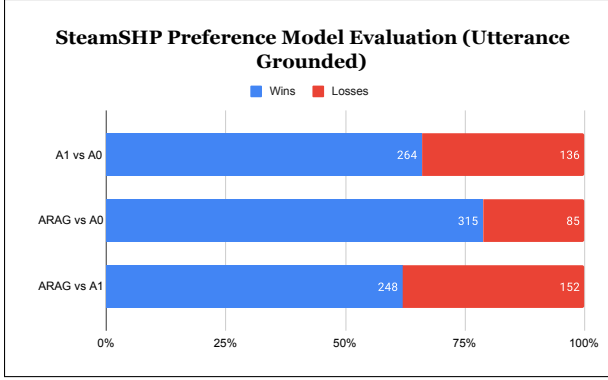
Figure 7: SteamSHP Preference model Evaluation of Assistants created from Utterance Grounded Simulators against baseline assistant (A0)
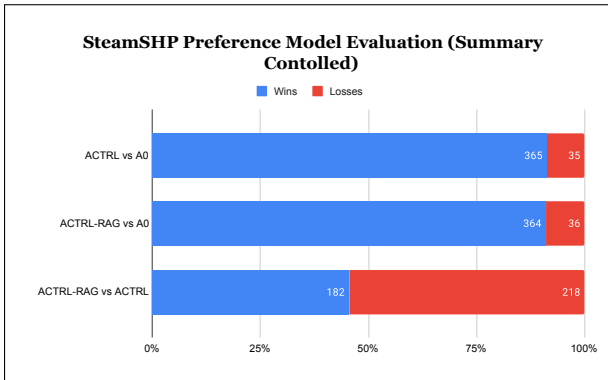

Figure 8: SteamSHP Preference model Evaluation of Assistants created from Summary Controlled (-CTRL) against baseline assistant (A0)

6 Conclusion

Simulators provide a way to generate data to create downstream assistant models saving human time and effort. Through our framework **Kaucus**, we further showed that augmenting simulators by exploiting external knowledge helps generate diverse interactions and as well as creates more helpful assistants than vanilla simulators. We describe two types of knowledge-augmented simulators, a Retrieval Augmented Simulator, SRAG, and a summary-controlled simulator, SCTRL both of which consume external knowledge in unique ways.

Raw text is more prevalent than the conversational counterparts. Controlling simulators through conversational summaries or external documents can be a quick and powerful tool to convert public text to trainable interaction data and create more helpful assistants. It provisions the simulator to generate interactions for novel information outside the scope of an LLM's intrinsic parameters. We

hope **Kaucus** will help encourage the development of automated techniques to be able to incorporate the vast amount of text produced rapidly over the internet and align assistant models better with newer data as well as be able to control the distribution of training data without the need for a rigid schema.

Limitations

Retrieval Augmentation helps incorporate diversity as well as benefit downstream models. We chose to use BM25 as our choice of retriever. However, there are dense retrievers (Khattab and Zaharia, 2020) and neural rerankers (Pradeep et al., 2023) that perform better than BM25 across a range of information retrieval benchmarks. Our focus was to show the benefit of incorporating external knowledge while performing a rigorous set of experiments with the same. Future studies could specifically study the impact of additional hyperparameter tuning by using varied choices of the retriever, the retrieving query, choice of summarisers and also gauge the impact of different domains than those of the Anthropic and the MSMarco datasets.

Besides, our study does not consider the impact of prolonged training on generated data which could cause potential problems of model forgetting over the long run (Shumailov et al., 2023). More experiments conducted to gauge long-term viability would shed better light on the efficacy of knowledge simulators.

All the evaluations conducted in this paper were automated – through popular reward or preference models. Human evaluations can provide better additional insights. Besides, the current intrinsic metrics primarily focus on diversity, which, while important, is only one dimension of dialogue evaluation and future work would benefit from other measures depending on the application.

Ethics Statement

Our study has focused on the benefits of employing simulators to improve downstream assistant models. We believe that these simulators can also act as effective testers of assistants to pre-encounter and regurgitate harmful or undesirable assistant content before assistant models are deployed in impacting end applications. We should maintain caution against their unethical usage or if such regurgitation is exploited to cause harm. Just like assistants or other applications of large language models (Dhole, 2023), simulators should also be gauged from a

socio-technical lens, and appropriate checks and fallback mechanisms should be employed before their actual usage. Besides, simulators themselves could inadvertently learn biases in the training data, leading to unfair or biased generations, and can be exploited for malicious purposes such as generating fake news and harmful content or asking triggering questions.

Acknowledgements

References

Gati V Aher, Rosa I Arriaga, and Adam Tauman Kalai. 2023. Using large language models to simulate multiple humans and replicate human subject studies. In *International Conference on Machine Learning*, pages 337–371. PMLR.

Lisa P Argyle, Ethan C Busby, Nancy Fulda, Joshua R Gubler, Christopher Rytting, and David Wingate. 2023. Out of one, many: Using language models to simulate human samples. *Political Analysis*, 31(3):337–351.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, et al. 2022. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*.

Payal Bajaj, Daniel Campos, Nick Craswell, Li Deng, Jianfeng Gao, Xiaodong Liu, Rangan Majumder, Andrew McNamara, Bhaskar Mitra, Tri Nguyen, et al. 2016. Ms marco: A human generated machine reading comprehension dataset. *arXiv preprint arXiv:1611.09268*.

Ning Bian, Hongyu Lin, Yaojie Lu, Xianpei Han, Le Sun, and Ben He. 2023. Chatalpaca: A multi-turn dialogue corpus based on alpaca instructions. https://github.com/cascip/ChatAlpaca.

Florian Böhm, Yang Gao, Christian M. Meyer, Ori Shapira, Ido Dagan, and Iryna Gurevych. 2019. Better rewards yield better summaries: Learning to summarise without references. 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 3110–3120, Hong Kong, China. Association for Computational Linguistics.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Myra Cheng, Tiziano Piccardi, and Diyi Yang. 2023. CoMPosT: Characterizing and evaluating caricature in LLM simulations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 10853–10875, Singapore. Association for Computational Linguistics.

Together Computer. 2023. Redpajama: An open source recipe to reproduce llama training dataset.

Zhuyun Dai, Arun Tejasvi Chaganty, Vincent Zhao, Aida Amini, Mike Green, Qazi Rashid, and Kelvin Guu. 2022. Dialog inpainting: Turning documents to dialogs. In *International Conference on Machine Learning (ICML)*. PMLR.

Kaustubh Dhole. 2023. Large language models as SocioTechnical systems. In *Proceedings of the Big Picture Workshop*, pages 66–79, Singapore, Singapore. Association for Computational Linguistics.

Kaustubh Dhole and Eugene Agichtein. 2024. Genqrensemble : Zero-shot llm ensemble prompting for generative query reformulation. In *Advances in Information Retrieval*.

Kaustubh Dhole, Varun Gangal, Sebastian Gehrmann, Aadesh Gupta, Zhenhao Li, Saad Mahamood, Abinaya Mahadiran, Simon Mille, Ashish Shrivastava, Samson Tan, et al. 2023. Nl-augmenter: A framework for task-sensitive natural language augmentation. *Northern European Journal of Language Technology*, 9(1).

Kawin Ethayarajh, Yejin Choi, and Swabha Swayamdipta. 2022. Understanding dataset difficulty with $\mathcal{V}$-usable information. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 5988–6008. PMLR.

Felix Faltings, Michel Galley, Kianté Brantley, Baolin Peng, Weixin Cai, Yizhe Zhang, Jianfeng Gao, and Bill Dolan. 2023. Interactive text generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 4450–4468, Singapore. Association for Computational Linguistics.

Jack FitzGerald, Shankar Ananthakrishnan, Konstantine Arkoudas, Davide Bernardi, Abhishek Bhagia, Claudio Delli Bovi, Jin Cao, Rakesh Chada, Amit Chauhan, Luoxin Chen, Anurag Dwarakanath, Satyam Dwivedi, Turan Gojayev, Karthik Gopalakrishnan, Thomas Gueudre, Dilek Hakkani-Tur, Wael Hamza, Jonathan J. Hüser, Kevin Martin Jose, Haidar Khan, Beiye Liu, Jianhua Lu, Alessandro Manzotti, Pradeep Natarajan, Karolina Owczarzak, Gokmen Oz, Enrico Palumbo, Charith Peris, Chandana Satya Prakash, Stephen Rawls, Andy Rosenbaum, Anjali Shenoy, Saleh Soltan, Mukund Harakere Sridhar, Lizhen Tan, Fabian Triefenbach, Pan Wei, Haiyang Yu, Shuai Zheng, Gokhan Tur, and Prem Natarajan. 2022. Alexa teacher model: Pretraining and distilling multi-billion-parameter encoders for natural language understanding systems. In *Proceedings of the*

28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD '22, page 2893–2902, New York, NY, USA. Association for Computing Machinery.

Luyu Gao, Zhuyun Dai, Panupong Pasupat, Anthony Chen, Arun Tejasvi Chaganty, Yicheng Fan, Vincent Zhao, Ni Lao, Hongrae Lee, Da-Cheng Juan, et al. 2023. Rarr: Researching and revising what language models say, using language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16477–16508.

Philippe J Giabbanelli. 2023. Gpt-based models meet simulation: How to efficiently use large-scale pre-trained language models across simulation tasks. *arXiv preprint arXiv:2306.13679*.

Izzeddin Gur, Semih Yavuz, Yu Su, and Xifeng Yan. 2018. DialSQL: Dialogue based structured query generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1339–1349, Melbourne, Australia. Association for Computational Linguistics.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. 2020. Realm: Retrieval-augmented language model pre-training. *ArXiv*, abs/2002.08909.

John J Horton. 2023. Large language models as simulated economic agents: What can we learn from homo silicus? Technical report, National Bureau of Economic Research.

Geoffrey Irving, Paul Christiano, and Dario Amodei. 2018. Ai safety via debate. *arXiv preprint arXiv:1805.00899*.

Zhengbao Jiang, Frank Xu, Luyu Gao, Zhiqing Sun, Qian Liu, Jane Dwivedi-Yu, Yiming Yang, Jamie Callan, and Graham Neubig. 2023. Active retrieval augmented generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 7969–7992, Singapore. Association for Computational Linguistics.

Omar Khattab and Matei Zaharia. 2020. Colbert: Efficient and effective passage search via contextualized late interaction over bert. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*, pages 39–48.

Hyunwoo Kim, Jack Hessel, Liwei Jiang, Peter West, Ximing Lu, Youngjae Yu, Pei Zhou, Ronan Le Bras, Malihe Alikhani, Gunhee Kim, Maarten Sap, and Yejin Choi. 2023. Soda: Million-scale dialogue distillation with social commonsense contextualization.

Andreas Köpf, Yannic Kilcher, Dimitri von Rütte, Sotiris Anagnostidis, Zhi Rui Tam, Keith Stevens, Abdullah Barhoum, Duc Minh Nguyen, Oliver Stanley, Richárd Nagyfi, Shahul ES, Sameer Suri, David Alexandrovich Glushkov, Arnav Varma Dantuluri, Andrew Maguire, Christoph Schuhmann, Huu Nguyen, and Alexander Julian Mattick. 2023. Openassistant conversations - democratizing large language model alignment. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Matthias Kraus, Ron Riekenbrauck, and Wolfgang Minker. 2023a. Development of a trust-aware user simulator for statistical proactive dialog modeling in human-ai teams. In *Adjunct Proceedings of the 31st ACM Conference on User Modeling, Adaptation and Personalization*, pages 38–43.

Matthias Kraus, Ron Riekenbrauck, and Wolfgang Minker. 2023b. Development of a trust-aware user simulator for statistical proactive dialog modeling in human-ai teams. In *Adjunct Proceedings of the 31st ACM Conference on User Modeling, Adaptation and Personalization*, pages 38–43.

Florian Kreyssig, Iñigo Casanueva, Paweł Budzianowski, and Milica Gašić. 2018. Neural user simulation for corpus-based policy optimisation of spoken dialogue systems. In *Proceedings of the 19th Annual SIGdial Meeting on Discourse and Dialogue*, pages 60–69, Melbourne, Australia. Association for Computational Linguistics.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474.

Huayang Li, Yixuan Su, Deng Cai, Yan Wang, and Lemao Liu. 2022a. A survey on retrieval-augmented text generation. *arXiv preprint arXiv:2202.01110*.

Zekun Li, Wenhu Chen, Shiyang Li, Hong Wang, Jing Qian, and Xifeng Yan. 2022b. Controllable dialogue simulation with in-context learning. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 4330–4347, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Hsien-Chin Lin, Shutong Feng, Christian Geishauser, Nurul Lubis, Carel van Niekerk, Michael Heck, Benjamin Ruppik, Renato Vukovic, and Milica Gašić. 2023. Emous: Simulating user emotions in task-oriented dialogues. SIGIR '23, page 2526–2531, New York, NY, USA. Association for Computing Machinery.

Yajiao Liu, Xin Jiang, Yichun Yin, Yasheng Wang, Fei Mi, Qun Liu, Xiang Wan, and Benyou Wang. 2023. One cannot stand for everyone! leveraging multiple user simulators to train task-oriented dialogue systems. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1–21.

Shayne Longpre, Le Hou, Tu Vu, Albert Webson, Hyung Won Chung, Yi Tay, Denny Zhou, Quoc V. Le, Barret Zoph, Jason Wei, and Adam Roberts. 2023. The flan collection: Designing data and methods for effective instruction tuning.

Sean MacAvaney, Andrew Yates, Sergey Feldman, Doug Downey, Arman Cohan, and Nazli Goharian. 2021. Simplified data wrangling with $ir_{datasets}$. $In SIGIR$.

Craig Macdonald, Nicola Tonellotto, Sean MacAvaney, and Iadh Ounis. 2021. Pyterrier: Declarative experimentation in python from bm25 to dense retrieval. In *Proceedings of the 30th acm international conference on information & knowledge management*, pages 4526–4533.

Philip M McCarthy and Scott Jarvis. 2010. Mtld, vocd-d, and hd-d: A validation study of sophisticated approaches to lexical diversity assessment. *Behavior research methods*, 42(2):381–392.

Simon Mille, Kaustubh Dhole, Saad Mahamood, Laura Perez-Beltrachini, Varun Gangal, Mihir Kale, Emiel van Miltenburg, and Sebastian Gehrmann. 2021. Automatic construction of evaluation suites for natural language generation datasets. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1)*.

Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. 2016. Ms marco: A human-generated machine reading comprehension dataset.

OpenAI. 2023. Gpt-4 technical report.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F Christiano, Jan Leike, and Ryan Lowe. 2022a. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*, volume 35, pages 27730–27744. Curran Associates, Inc.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022b. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744.

Ronak Pradeep, Sahel Sharifymoghaddam, and Jimmy Lin. 2023. Rankzephyr: Effective and robust zero-shot listwise reranking is a breeze! *arXiv preprint arXiv:2312.02724*.

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

Ori Ram, Yoav Levine, Itay Dalmedigos, Dor Muhlgay, Amnon Shashua, Kevin Leyton-Brown, and Yoav Shoham. 2023. In-Context Retrieval-Augmented Language Models. *Transactions of the Association for Computational Linguistics*, 11:1316–1331.

Jeff Rasley, Samyam Rajbhandari, Olatunji Ruwase, and Yuxiong He. 2020. Deepspeed: System optimizations enable training deep learning models with over 100 billion parameters. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 3505–3506.

Moritz Schaefer, Stephan Reichl, Rob ter Horst, Adele M Nicolas, Thomas Krausgruber, Francesco Piras, Peter Stepper, Christoph Bock, and Matthias Samwald. 2023. Large language models are universal biomedical simulators. *bioRxiv*, pages 2023–06.

Jost Schatzmann, Kallirroi Georgila, and Steve Young. 2005. Quantitative evaluation of user simulation techniques for spoken dialogue systems. In *6th SIGdial Workshop on DISCOURSE and DIALOGUE*.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessi, Roberta Raileanu, Maria Lomeli, Eric Hambro, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. 2023. Toolformer: Language models can teach themselves to use tools. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Ilia Shumailov, Zakhar Shumaylov, Yiren Zhao, Yarin Gal, Nicolas Papernot, and Ross Anderson. 2023. The curse of recursion: Training on generated data makes models forget.

Aarohi Srivastava, Abhinav Rastogi, Abhishek Rao, Abu Awal Md Shoeb, Abubakar Abid, Adam Fisch, Adam R. Brown, Adam Santoro, Aditya Gupta, Adrià Garriga-Alonso, Agnieszka Kluska, Aitor Lewkowycz, Akshat Agarwal, Alethea Power, Alex Ray, Alex Warstadt, Alexander W. Kocurek, Ali Safaya, Ali Tazarv, Alice Xiang, Alicia Parrish, Allen Nie, Aman Hussain, Amanda Askell, Amanda Dsouza, Ambrose Slone, Ameet Rahane, Anantharaman S. Iyer, Anders Johan Andreassen, Andrea Madotto, Andrea Santilli, Andreas Stuhlmüller, Andrew M. Dai, Andrew La, Andrew Lampinen, Andy Zou, Angela Jiang, Angelica Chen, Anh Vuong, Animesh Gupta, Anna Gottardi, Antonio Norelli, Anu Venkatesh, Arash Gholamidavoodi, Arfa Tabassum, Arul Menezes, Arun Kirubarajan, Asher Mullokandov, Ashish Sabharwal, Austin Herrick, Avia Efrat, Aykut Erdem, Ayla Karakaş, B. Ryan Roberts, Bao Sheng Loe, Barret Zoph, Bartłomiej Bojanowski, Batuhan Özyurt, Behnam Hedayatnia, Behnam Neyshabur, Benjamin Inden, Benno Stein, Berk Ekmekci, Bill Yuchen Lin, Blake Howald, Bryan Orinion, Cameron Diao, Cameron Dour, Catherine Stinson, Cedrick Argueta, Cesar Ferri, Chandan Singh, Charles Rathkopf, Chenlin Meng, Chitta Baral, Chiyu Wu, Chris Callison-Burch, Christopher Waites, Christian Voigt, Christopher D Manning, Christopher Potts, Cindy Ramirez, Clara E. Rivera, Clemencia Siro, Colin Raffel, Courtney Ashcraft, Cristina Garbacea,

Damien Sileo, Dan Garrette, Dan Hendrycks, Dan Kilman, Dan Roth, C. Daniel Freeman, Daniel Khashabi, Daniel Levy, Daniel Moseguí González, Danielle Perszyk, Danny Hernandez, Danqi Chen, Daphne Ippolito, Dar Gilboa, David Dohan, David Drakard, David Jurgens, Debajyoti Datta, Deep Ganguli, Denis Emelin, Denis Kleyko, Deniz Yuret, Derek Chen, Derek Tam, Dieuwke Hupkes, Diganta Misra, Dilyar Buzan, Dimitri Coelho Mollo, Diyi Yang, Dong-Ho Lee, Dylan Schrader, Ekaterina Shutova, Ekin Dogus Cubuk, Elad Segal, Eleanor Hagerman, Elizabeth Barnes, Elizabeth Donoway, Ellie Pavlick, Emanuele Rodolà, Emma Lam, Eric Chu, Eric Tang, Erkut Erdem, Ernie Chang, Ethan A Chi, Ethan Dyer, Ethan Jerzak, Ethan Kim, Eunice Engefu Manyasi, Evgenii Zheltonozhskii, Fanyue Xia, Fatemeh Siar, Fernando Martínez-Plumed, Francesca Happé, Francois Chollet, Frieda Rong, Gaurav Mishra, Genta Indra Winata, Gerard de Melo, Germán Kruszewski, Giambattista Parascandolo, Giorgio Mariani, Gloria Xinyue Wang, Gonzalo Jaimovitch-Lopez, Gregor Betz, Guy Gur-Ari, Hana Galijasevic, Hannah Kim, Hannah Rashkin, Hannaneh Hajishirzi, Harsh Mehta, Hayden Bogar, Henry Francis Anthony Shevlin, Hinrich Schuetze, Hiromu Yakura, Hongming Zhang, Hugh Mee Wong, Ian Ng, Isaac Noble, Jaap Jumelet, Jack Geissinger, Jackson Kernion, Jacob Hilton, Jaehoon Lee, Jaime Fernández Fisac, James B Simon, James Koppel, James Zheng, James Zou, Jan Kocon, Jana Thompson, Janelle Wingfield, Jared Kaplan, Jarema Radom, Jascha Sohl-Dickstein, Jason Phang, Jason Wei, Jason Yosinski, Jekaterina Novikova, Jelle Bosscher, Jennifer Marsh, Jeremy Kim, Jeroen Taal, Jesse Engel, Jesujoba Alabi, Jiacheng Xu, Jiaming Song, Jillian Tang, Joan Waweru, John Burden, John Miller, John U. Balis, Jonathan Batchelder, Jonathan Berant, Jörg Frohberg, Jos Rozen, Jose Hernandez-Orallo, Joseph Boudeman, Joseph Guerr, Joseph Jones, Joshua B. Tenenbaum, Joshua S. Rule, Joyce Chua, Kamil Kanclerz, Karen Livescu, Karl Krauth, Karthik Gopalakrishnan, Katerina Ignatyeva, Katja Markert, Kaustubh Dhole, Kevin Gimpel, Kevin Omondi, Kory Wallace Mathewson, Kristen Chiafullo, Ksenia Shkaruta, Kumar Shridhar, Kyle McDonell, Kyle Richardson, Laria Reynolds, Leo Gao, Li Zhang, Liam Dugan, Lianhui Qin, Lidia Contreras-Ochando, Louis-Philippe Morency, Luca Moschella, Lucas Lam, Lucy Noble, Ludwig Schmidt, Luheng He, Luis Oliveros-Colón, Luke Metz, Lütfi Kerem Senel, Maarten Bosma, Maarten Sap, Maartje Ter Hoeve, Maheen Farooqi, Manaal Faruqui, Mantas Mazeika, Marco Baturan, Marco Marelli, Marco Maru, Maria Jose Ramirez-Quintana, Marie Tolkiehn, Mario Giulianelli, Martha Lewis, Martin Potthast, Matthew L Leavitt, Matthias Hagen, Mátyás Schubert, Medina Orduna Baitemirova, Melody Arnaud, Melvin McElrath, Michael Andrew Yee, Michael Cohen, Michael Gu, Michael Ivanitskiy, Michael Starritt, Michael Strube, Michał Swędrowski, Michele Bevilacqua, Michihiro Yasunaga, Mihir Kale, Mike Cain, Mimee Xu, Mirac Suzgun, Mitch Walker, Mo Tiwari, Mohit Bansal, Moin Aminnaseri, Mor Geva, Mozhdeh Gheini, Mukund Varma T, Nanyun Peng, Nathan Andrew Chi, Nayeon Lee, Neta Gur-Ari Krakover, Nicholas Cameron, Nicholas Roberts, Nick Doiron, Nicole Martinez, Nikita Nangia, Niklas Deckers, Niklas Muennighoff, Nitish Shirish Keskar, Niveditha S. Iyer, Noah Constant, Noah Fiedel, Nuan Wen, Oliver Zhang, Omar Agha, Omar Elbaghdadi, Omer Levy, Owain Evans, Pablo Antonio Moreno Casares, Parth Doshi, Pascale Fung, Paul Pu Liang, Paul Vicol, Pegah Alipoormolabashi, Peiyuan Liao, Percy Liang, Peter W Chang, Peter Eckersley, Phu Mon Htut, Pinyu Hwang, Piotr Miłkowski, Piyush Patil, Pouya Pezeshkpour, Priti Oli, Qiaozhu Mei, Qing Lyu, Qinlang Chen, Rabin Banjade, Rachel Etta Rudolph, Raefer Gabriel, Rahel Habacker, Ramon Risco, Raphaël Millière, Rhythm Garg, Richard Barnes, Rif A. Saurous, Riku Arakawa, Robbe Raymaekers, Robert Frank, Rohan Sikand, Roman Novak, Roman Sitelew, Ronan Le Bras, Rosanne Liu, Rowan Jacobs, Rui Zhang, Russ Salakhutdinov, Ryan Andrew Chi, Seungjae Ryan Lee, Ryan Stovall, Ryan Teehan, Rylan Yang, Sahib Singh, Saif M. Mohammad, Sajant Anand, Sam Dillavou, Sam Shleifer, Sam Wiseman, Samuel Gruetter, Samuel R. Bowman, Samuel Stern Schoenholz, Sanghyun Han, Sanjeev Kwatra, Sarah A. Rous, Sarik Ghazarian, Sayan Ghosh, Sean Casey, Sebastian Bischoff, Sebastian Gehrmann, Sebastian Schuster, Sepideh Sadeghi, Shadi Hamdan, Sharon Zhou, Shashank Srivastava, Sherry Shi, Shikhar Singh, Shima Asaadi, Shixiang Shane Gu, Shubh Pachchigar, Shubham Toshniwal, Shyam Upadhyay, Shyamolima Shammie Debnath, Siamak Shakeri, Simon Thormeyer, Simone Melzi, Siva Reddy, Sneha Priscilla Makini, Soo-Hwan Lee, Spencer Torene, Sriharsha Hatwar, Stanislas Dehaene, Stefan Divic, Stefano Ermon, Stella Biderman, Stephanie Lin, Stephen Prasad, Steven Piantadosi, Stuart Shieber, Summer Misherghi, Svetlana Kiritchenko, Swaroop Mishra, Tal Linzen, Tal Schuster, Tao Li, Tao Yu, Tariq Ali, Tatsunori Hashimoto, Te-Lin Wu, Théo Desbordes, Theodore Rothschild, Thomas Phan, Tianle Wang, Tiberius Nkinyili, Timo Schick, Timofei Kornev, Titus Tunduny, Tobias Gerstenberg, Trenton Chang, Trishala Neeraj, Tushar Khot, Tyler Shultz, Uri Shaham, Vedant Misra, Vera Demberg, Victoria Nyamai, Vikas Raunak, Vinay Venkatesh Ramasesh, vinay uday prabhu, Vishakh Padmakumar, Vivek Srikumar, William Fedus, William Saunders, William Zhang, Wout Vossen, Xiang Ren, Xiaoyu Tong, Xinran Zhao, Xinyi Wu, Xudong Shen, Yadollah Yaghoobzadeh, Yair Lakretz, Yangqiu Song, Yasaman Bahri, Yejin Choi, Yichi Yang, Yiding Hao, Yifu Chen, Yonatan Belinkov, Yu Hou, Yufang Hou, Yuntao Bai, Zachary Seid, Zhuoye Zhao, Zijian Wang, Zijie J. Wang, Zirui Wang, and Ziyi Wu. 2023. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *Transactions on Machine Learning Research*.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Stanford alpaca: An instruction-following llama model. https://github.com/tatsu-lab/stanford_alpaca.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al.

2023. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Ben Wang and Aran Komatsuzaki. 2021. GPT-J-6B: A 6 Billion Parameter Autoregressive Language Model. https://github.com/kingoflolz/mesh-transformer-jax.

Yile Wang, Peng Li, Maosong Sun, and Yang Liu. 2023a. Self-knowledge guided retrieval augmentation for large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 10303–10315, Singapore. Association for Computational Linguistics.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2023b. Self-instruct: Aligning language models with self-generated instructions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13484–13508, Toronto, Canada. Association for Computational Linguistics.

Jason Wei, Maarten Bosma, Vincent Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. Finetuned language models are zero-shot learners. In *International Conference on Learning Representations*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Yue Zhang, Yafu Li, Leyang Cui, Deng Cai, Lemao Liu, Tingchen Fu, Xinting Huang, Enbo Zhao, Yu Zhang, Yulong Chen, Longyue Wang, Anh Tuan Luu, Wei Bi, Freda Shi, and Shuming Shi. 2023. Siren's song in the ai ocean: A survey on hallucination in large language models. *ArXiv*, abs/2309.01219.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric. P Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena.

Daniel M. Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B. Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. 2019. Fine-tuning language models from human preferences. *arXiv preprint arXiv:1909.08593*.