Gemma: Open Models Based on Gemini Research and Technology

Gemma Team, Google DeepMind[1]

[1] See Contributions and Acknowledgments section for full author list. Please send correspondence to gemma-1-report@google.com.

This work introduces Gemma, a family of lightweight, state-of-the art open models built from the research and technology used to create Gemini models. Gemma models demonstrate strong performance across academic benchmarks for language understanding, reasoning, and safety. We release two sizes of models (2 billion and 7 billion parameters), and provide both pretrained and fine-tuned checkpoints. Gemma outperforms similarly sized open models on 11 out of 18 text-based tasks, and we present comprehensive evaluations of safety and responsibility aspects of the models, alongside a detailed description of model development. We believe the responsible release of LLMs is critical for improving the safety of frontier models, and for enabling the next wave of LLM innovations.

Introduction

We present Gemma, a family of open models based on Google's Gemini models (Gemini Team, 2023).

We trained Gemma models on up to 6T tokens of text, using similar architectures, data, and training recipes as the Gemini model family. Like Gemini, these models achieve strong generalist capabilities in text domains, alongside state-of-the-art understanding and reasoning skills at scale. With this work, we release both pre-trained and fine-tuned checkpoints, as well as an open-source codebase for inference and serving.

Gemma comes in two sizes: a 7 billion parameter model for efficient deployment and development on GPU and TPU, and a 2 billion parameter model for CPU and on-device applications. Each size is designed to address different computational constraints, applications, and developer requirements. At each scale, we release raw, pretrained checkpoints, as well as checkpoints fine-tuned for dialogue, instruction-following, helpfulness, and safety. We thoroughly evaluate the shortcomings of our models on a suite of quantitative and qualitative benchmarks. We believe the release of both pretrained and fine-tuned checkpoints will enable thorough research and investigation into the impact of current instruction-tuning regimes, as well as the development of increasingly safe and responsible model development methodologies.

Gemma advances state-of-the-art performance relative to comparable-scale (and some larger), open models (Almazrouei et al., 2023; Jiang et al., 2023; Touvron et al., 2023a,b) across a wide range of domains including both automated benchmarks and human evaluation. Example domains include question answering (Clark et al., 2019; Kwiatkowski et al., 2019), commonsense reasoning (Sakaguchi et al., 2019; Suzgun et al., 2022), mathematics and science (Cobbe et al., 2021; Hendrycks et al., 2020), and coding (Austin et al., 2021; Chen et al., 2021). See complete details in the Evaluation section.

Like Gemini, Gemma builds on recent work on sequence models (Sutskever et al., 2014) and transformers (Vaswani et al., 2017), deep learning methods based on neural networks (LeCun et al., 2015), and techniques for large-scale training on distributed systems (Barham et al., 2022; Dean et al., 2012; Roberts et al., 2023). Gemma also builds on Google's long history of open models and ecosystems, including Word2Vec (Mikolov et al., 2013), the Transformer (Vaswani et al., 2017), BERT (Devlin et al., 2018), and T5 (Raffel et al., 2019) and T5X (Roberts et al., 2022).

We believe the responsible release of LLMs is critical for improving the safety of frontier models, for ensuring equitable access to this breakthrough technology, for enabling rigorous evaluation and analysis of current techniques, and for enabling

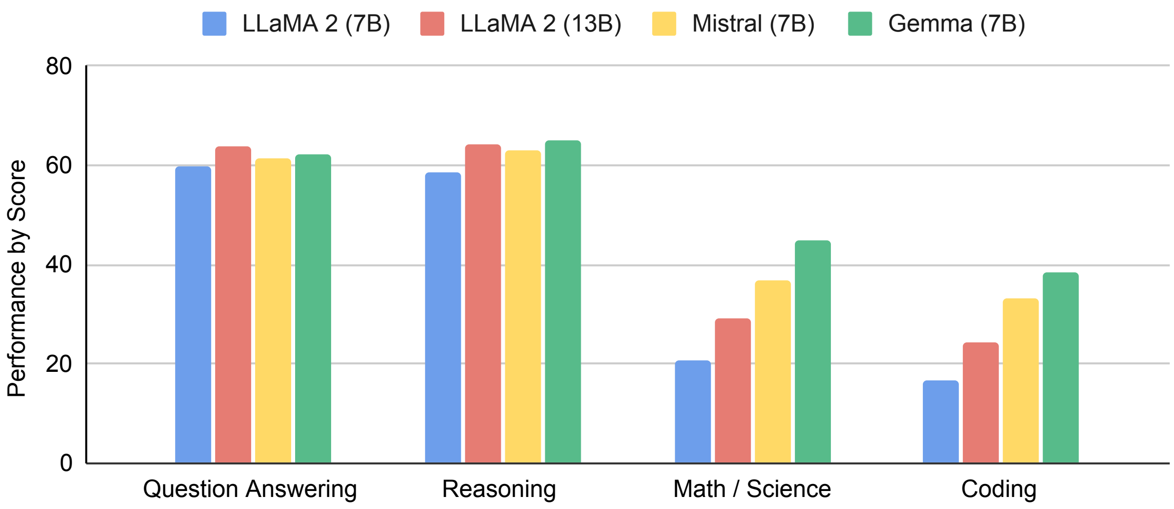

Figure 1 | Language understanding and generation performance of Gemma 7B across different capabilities compared to similarly sized open models. We group together standard academic benchmark evaluations by capability and average the respective scores; see Table 6 for a detailed breakdown of performance.

the development of the next wave of innovations. While thorough testing of all Gemma models has been conducted, testing cannot cover all applications and scenarios in which Gemma may be used. With this in mind, all Gemma users should conduct rigorous safety testing specific to their use case before deployment or use. More details on our approach to safety can be found in section Responsible Deployment.

In this technical report, we provide a detailed overview of the model architecture, training infrastructure, and pretraining and fine-tuning recipes for Gemma, followed by thorough evaluations of all checkpoints across a wide-variety of quantitative and qualitative benchmarks, as well as both standard academic benchmarks and human-preference evaluations. We then discuss in detail our approach to safe and responsible deployment. Finally, we outline the broader implications of Gemma, its limitations and advantages, and conclusions.

Parameters	**2B**	**7B**
*d*_model	2048	3072
Layers	18	28
Feedforward hidden dims	32768	49152
Num heads	8	16
Num KV heads	1	16
Head size	256	256
Vocab size	256128	256128

Table 1 | Key model parameters.

Model	Embedding Parameters	Non-embedding Parameters
2B	524,550,144	1,981,884,416
7B	786,825,216	7,751,248,896

Table 2 | Parameter counts for both sizes of Gemma models.

Model Architecture

The Gemma model architecture is based on the transformer decoder (Vaswani et al., 2017). The core parameters of the architecture are summarized in Table 1. Models are trained on a context length of 8192 tokens.

We also utilize several improvements proposed after the original transformer paper. Below, we list the included improvements:

Multi-Query Attention (Shazeer, 2019). Notably, the 7B model uses multi-head attention while the 2B checkpoints use multi-query attention (with *num_kv_heads* = 1), based on ablation studies that revealed respective attention variants improved performance at each scale (Shazeer, 2019).

RoPE Embeddings (Su et al., 2021). Rather than using absolute positional embeddings, we use rotary positional embeddings in each layer; we also share embeddings across our inputs and outputs to reduce model size.

GeGLU Activations (Shazeer, 2020). The standard ReLU non-linearity is replaced by the GeGLU activation function.

Normalizer Location. We normalize both the input and the output of each transformer sub-layer, a deviation from the standard practice of solely normalizing one or the other. We use RMSNorm (Zhang and Sennrich, 2019) as our normalization layer.

Training Infrastructure

We train the Gemma models using TPUv5e; TPUv5e are deployed in pods of 256 chips, configured into a 2D torus of 16 x 16 chips. For the 7B model, we train our model across 16 pods, totaling to 4096 TPUv5e. We pretrain the 2B model across 2 pods, totaling 512 TPUv5e. Within a pod, we use 16-way model sharding and 16-way data replication for the 7B model. For the 2B, we simply use 256-way data replication. The optimizer state is further sharded using techniques similar to ZeRO-3. Beyond a pod, we perform data-replica reduce over the data-center network, using Pathways approach of (Barham et al., 2022).

As in Gemini, we leverage the 'single controller' programming paradigm of Jax (Roberts et al., 2023) and Pathways (Barham et al., 2022) to simplify the development process by enabling a single Python process to orchestrate the entire training run; we also leverage the GSPMD partitioner (Xu et al., 2021) for the training step computation and the MegaScale XLA compiler (XLA, 2019).

Carbon Footprint

We estimate the carbon emissions from pretraining the Gemma models to be ~ 131 tCO_2eq. This value is calculated based on the hourly energy usage reported directly from our TPU datacenters; we also scale this value to account for the additional energy expended to create and maintain the data center, giving us the total energy usage for our training experiments. We convert total energy usage to carbon emissions by joining our hourly energy usage against hourly per-cell carbon emission data reported by our data centers.

In addition, Google data centers are carbon neutral, achieved through a combination of energy efficiency, renewable energy purchases, and carbon offsets. This carbon neutrality also applies to our experiments and the machines used to run them.

Pretraining

Training Data

Gemma 2B and 7B are trained on 2T and 6T tokens respectively of primarily-English data from web documents, mathematics, and code. Unlike Gemini, these models are not multimodal, nor are they trained for state-of-the-art performance on multilingual tasks.

We use a subset of the SentencePiece tokenizer (Kudo and Richardson, 2018) of Gemini for compatibility. It splits digits, does not remove extra whitespace, and relies on byte-level encodings for unknown tokens, following the techniques used for both (Chowdhery et al., 2022) and (Gemini Team, 2023). The vocabulary size is 256k tokens.

Filtering

We filter the pre-training dataset to reduce the risk of unwanted or unsafe utterances, and filter out certain personal information and other sensitive data. This includes using both heuristics and model-based classifiers to remove harmful or low-quality content. Further, we filter all evaluation sets from our pre-training data mixture, run targeted contamination analyses to check against evaluation set leakage, and reduce the risk of recitation by minimizing proliferation of sensitive outputs.

The final data mixture was determined through a series of ablations on both the 2B and 7B models. Similar to the approach advocated in (Gemini Team, 2023), we stage training to alter the corpus mixture throughout training to increase the weight of relevant, high-quality data towards the end of training.

Instruction Tuning

We finetune Gemma 2B and 7B with supervised fine-tuning (SFT) on a mix of text-only, English-only synthetic and human-generated prompt-response pairs and reinforcement learning from human feedback (RLHF) with the reward model trained on labelled English-only preference data and the policy based on a set of high-quality prompts. We find that both stages are important for improved performance on downstream automatic evaluations and human preference evaluations of model outputs.

Supervised Fine-Tuning

We selected our data mixtures for supervised fine-tuning based on LM-based side-by-side evaluations (Zheng et al., 2023). Given a set of held-out prompts, we generate responses from a test model, generate responses on the same prompts from a baseline model, shuffle these randomly, and ask a larger, high capability model to express a preference between two responses. Different prompt sets are constructed to highlight specific capabilities, such as instruction following, factuality, creativity, and safety. The different automatic LM-based judges we use employ a number of techniques, such as chain-of-thought prompting (Wei et al., 2022) and use of rubrics and constitutions (Bai et al., 2022), to be aligned with human preferences.

Filtering

When using synthetic data, we run several stages of filtering over it, removing examples that show certain personal information, unsafe or toxic model outputs, mistaken self-identification data, or duplicated examples. Following Gemini, we find that including subsets of data that encourage better in-context attribution, hedging, and refusals to minimize hallucinations can improve performance on several factuality metrics, without degrading model performance on other metrics.

The final data mixtures and supervised fine-tuning recipe, which includes tuned hyperparameters, were chosen on the basis of improving helpfulness while minimizing model harms related to safety and hallucinations.

Formatting

Instruction tuned models are trained with a specific formatter that annotates all instruction tuning examples with extra information, both at training and inference time. It has two purposes: 1) indicating roles in a conversation, such as the User role, and 2) delineating turns in a conversation, especially in a multi-turn conversation. Special control tokens are reserved in the tokenizer for this purpose. While it is possible to get coherent generations without the formatter, it will be out-of-distribution for the model, and will very likely produce worse generations.

The relevant formatting control tokens are presented in Table 3, with a dialogue example presented in Table 4.

Context	Relevant Token
User turn	`user`
Model turn	`model`
Start of conversation turn	`<start_of_turn>`
End of conversation turn	`<end_of_turn>`

Table 3 | Relevant formatting control tokens used for both SFT and RLHF of Gemma models.

Reinforcement Learning from Human Feedback

We further finetuned the supervised fine-tuned model using RLHF (Christiano et al., 2017; Ouyang et al., 2022). We collected pairs of pref-

User:	`<start_of_turn>user`
	`Knock knock.<end_of_turn>`
	`<start_of_turn>model`

Model:	`Who's there?<end_of_turn>model`

User:	`<start_of_turn>user`
	`Gemma.<end_of_turn>`
	`<start_of_turn>model`

Model:	`Gemma who?<end_of_turn>model`

Table 4 | Example dialogue with user and model control tokens.

erences from human raters and trained a reward function under the Bradley-Terry model (Bradley and Terry, 1952), similarly to Gemini. The policy was trained to optimize this reward function using a variant of REINFORCE (Williams, 1992) with a Kullback–Leibler regularization term towards the initially tuned model. Similar to the SFT phase, and in order to tune hyperparameters and additionally mitigate reward hacking (Amodei et al., 2016; Skalse et al., 2022) we relied on a high capacity model as an automatic rater and computed side-by-side comparisons against baseline models.

Evaluation

We evaluate Gemma across a broad range of domains, using both automated benchmarks and human evaluation.

Human Preference Evaluations

In addition to running standard academic benchmarks on the finetuned models, we sent final release candidates to human evaluation studies to be compared against the Mistral v0.2 7B Instruct model (Jiang et al., 2023).

On a held-out collection of around 1000 prompts oriented toward asking models to follow instructions across creative writing tasks, coding, and following instructions, Gemma 7B IT has a 51.7% positive win rate and Gemma 2B IT has a 41.6% win rate over Mistral v0.2 7B Instruct. On a held-out collection of around 400 prompts oriented towards testing basic safety protocols, Gemma 7B IT has a 58% win rate, while Gemma 2B IT has a 56.5% win rate. We report the corresponding numbers in Table 5.

Model	Safety	Instruction Following
Gemma 7B IT	**58%**	**51.7%**
95% Conf. Interval	[55.9%, 60.1%]	[49.6%, 53.8%]
Win / Tie / Loss	42.9% / 30.2% / 26.9%	42.5% / 18.4% / 39.1%
Gemma 2B IT	**56.5%**	41.6%
95% Conf. Interval	[54.4%, 58.6%]	[39.5%, 43.7%]
Win / Tie / Loss	44.8% / 22.9% / 32.3%	32.7% / 17.8% / 49.5%

Table 5 | Win rate of Gemma models versus Mistral 7B v0.2 Instruct with 95% confidence intervals. We report breakdowns of wins, ties, and losses, and we break ties evenly when reporting the final win rate.

Automated Benchmarks

We measure Gemma models' performance on domains including physical reasoning (Bisk et al., 2019), social reasoning (Sap et al., 2019), question answering (Clark et al., 2019; Kwiatkowski et al., 2019), coding (Austin et al., 2021; Chen et al., 2021), mathematics (Cobbe et al., 2021), commonsense reasoning (Sakaguchi et al., 2019), language modeling (Paperno et al., 2016), reading comprehension (Joshi et al., 2017), and more.

For most automated benchmarks we use the same evaluation methodology as in Gemini. Specifically for those where we report performance compared with Mistral, we replicated methodology from the Mistral technical report as closely as possible. These specific benchmarks are: ARC (Clark et al., 2018), CommonsenseQA (Talmor et al., 2019), Big Bench Hard (Suzgun et al., 2022), and AGI Eval (English-only) (Zhong et al., 2023). Due to restrictive licensing, we were unable to run any evaluations on LLaMA-2 and cite only those metrics previously reported (Touvron et al., 2023b).

We compare Gemma 2B and 7B models to several external open-source (OSS) LLMs across a series of academic benchmarks, reported in Table 6.

On MMLU (Hendrycks et al., 2020), Gemma 7B outperforms all OSS alternatives at the same or smaller scale; it also outperforms several larger models, including LLaMA2 13B. However, human expert performance is gauged at 89.8% by the benchmark authors; as Gemini Ultra is the first

		LLaMA-2		Mistral	Gemma	
Benchmark	metric	7B	13B	7B	2B	7B
MMLU	5-shot, top-1	45.3	54.8	62.5	42.3	**64.3**
HellaSwag	0-shot	77.2	80.7	81.0	71.4	**81.2**
PIQA	0-shot	78.8	80.5	**82.2**	77.3	81.2
SIQA	0-shot	48.3	50.3	47.0*	49.7	**51.8**
Boolq	0-shot	77.4	81.7	**83.2***	69.4	**83.2**
Winogrande	partial scoring	69.2	72.8	**74.2**	65.4	72.3
CQA	7-shot	57.8	67.3	66.3*	65.3	**71.3**
OBQA		**58.6**	57.0	52.2	47.8	52.8
ARC-e		75.2	77.3	80.5	73.2	**81.5**
ARC-c		45.9	49.4	**54.9**	42.1	53.2
TriviaQA	5-shot	72.1	**79.6**	62.5	53.2	63.4
NQ	5-shot	25.7	**31.2**	23.2	12.5	23.0
HumanEval	pass@1	12.8	18.3	26.2	22.0	**32.3**
MBPP†	3-shot	20.8	30.6	40.2*	29.2	**44.4**
GSM8K	maj@1	14.6	28.7	35.4*	17.7	**46.4**
MATH	4-shot	2.5	3.9	12.7	11.8	**24.3**
AGIEval		29.3	39.1	41.2*	24.2	**41.7**
BBH		32.6	39.4	**56.1***	35.2	55.1
Average		47.0	52.2	54.0	44.9	**56.4**

Table 6 | Academic benchmark results, compared to similarly sized, openly-available models trained on general English text data. † Mistral reports 50.2 on a different split for MBPP and on their split our 7B model achieves 54.5. * evaluations run by us. Note that due to restrictive licensing, we were unable to run evals on LLaMA-2; all values above were previously reported in Touvron et al. (2023b).

model to exceed this threshold, there is significant room for continued improvements to achieve Gemini and human-level performance.

Gemma models demonstrate particularly strong performance on mathematics and coding benchmarks. On mathematics tasks, which are often used to benchmark the general analytical capabilities of models, Gemma models outperform other models by at least 10 points on GSM8K (Cobbe et al., 2021) and the more difficult MATH (Hendrycks et al., 2021) benchmark. Similarly, they outperform alternate open models by at least 6 points on HumanEval (Chen et al., 2021). They even surpass the performance of the code-fine-tuned CodeLLaMA-7B models on MBPP (CodeLLaMA achieves a score of 41.4% where Gemma 7B achieves 44.4%).

Memorization Evaluations

Recent work has shown that aligned models may be vulnerable to new adversarial attacks that can bypass alignment (Nasr et al., 2023). These attacks can cause models to diverge, and sometimes regurgitate memorized training data in the process. We focus on discoverable memorization, which serves as a reasonable upper-bound on the memorization of a model (Nasr et al., 2023) and has been the common definition used in several studies (Anil et al., 2023; Carlini et al., 2022; Kudugunta et al., 2023).

We test for memorization[1] of the Gemma pretrained models with the same methodology performed in Anil et al. (2023). We sample 10,000

[1] Our use of "memorization" relies on the definition of that term found at www.genlaw.org/glossary.html.

Benchmark	Mistral 7B	Gemma 7B
ARC-c	60.0	**61.9**
HellaSwag	**83.3**	82.2
MMLU	64.2	**64.56**
TruthfulQA	42.2	**44.8**
Winogrande	78.4	**79.0**
GSM8K	37.8	**50.9**
Average	61.0	**63.8**

Table 7 | HuggingFace H6 benchmark. The performance of small models are sensitive to small modifications in prompts and we further validate the quality of our models on an independent implementation of multiple known benchmarks. All evaluations were run by HuggingFace.

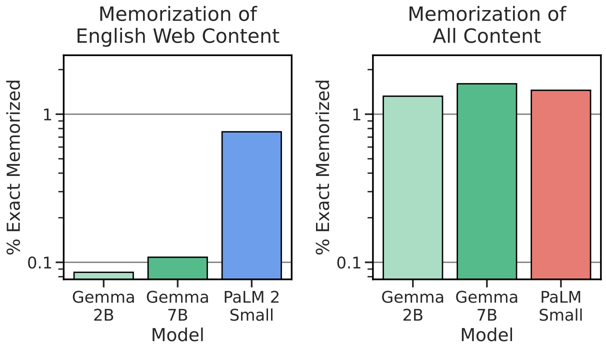

Figure 2 | Comparing average memorization rates across model families. We compare the Gemma pretrained models to PaLM and PaLM 2 models of comparable size and find similarly low rates of memorization.

documents from each corpus and use the first 50 tokens as a prompt for the model. We focus mainly on exact memorization, where we classify texts as memorized if the subsequent 50 tokens generated by the model exactly match the ground truth continuation in the text. However, to better capture potential paraphrased memorizations, we include approximate memorization (Ippolito et al., 2022) using an 10% edit distance threshold. In Figure 2, we compare the results of our evaluation with the closest sized PaLM (Chowdhery et al., 2022) and PaLM 2 models (Anil et al., 2023).

Verbatim Memorization PaLM 2 compared with PaLM by evaluating on a shared subset of their training corpora. However, there is even less overlap between the Gemma pretraining data with the PaLM models, and so using this same methodology, we observe much lower memorization rates (Figure 2 left). Instead, we find that estimating the "total memorization" across the entire pretraining dataset gives a more reliable estimate (Figure 2 right) where we now find the Gemma memorizes training data at a comparable rate to PaLM.

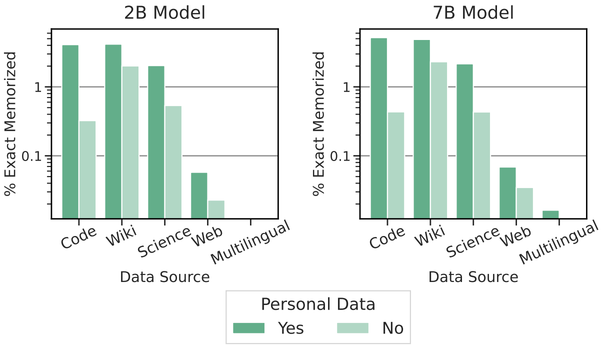

Figure 3 | Measuring personal and sensitive data memorization rates. **No sensitive data was memorized, hence it is omitted from the figure.**

Personal Data Perhaps of higher importance is the possibility that personal data might be memorized. As part of making Gemma pre-trained models safe and reliable, we used automated techniques to filter out certain personal information and other sensitive data from training sets.

To identify possible occurrences of personal data, we use the Google Cloud Data Loss Prevention (DLP) tool[2]. The tool outputs three severity levels based on many categories of personal data (e.g., names, emails, etc.). We classify the highest severity as "sensitive" and the remaining two as simply "personal". Then, we measure how many memorized outputs contain any sensitive or personal data. As shown in Figure 3, *we observe no cases of memorized sensitive data.* We do find that the model memorizes some data we have classified as potentially "personal" according to the

[2] Available at: `https://cloud.google.com/security/products/dlp`

		Mistral	Gemma	
Benchmark	metric	7B	2B	7B
RealToxicity	avg	8.44*	**6.86**	7.90
BOLD		38.21*	45.57	**49.08**
CrowS-Pairs	top-1	32.76*	45.82	**51.33**
BBQ Ambig	1-shot, top-1	**97.53***	62.58	92.54
BBQ Disambig	top-1	**84.45***	54.62	71.99
Winogender	top-1	**64.3***	51.25	54.17
TruthfulQA		44.2*	**44.84**	31.81
Winobias 1_2		**65.72***	56.12	59.09
Winobias 2_2		84.53*	91.1	**92.23**
Toxigen		60.26*	**29.77**	39.59

Table 8 | Safety academic benchmark results, compared to similarly sized, openly-available models. * evaluations run by us. Note that due to restrictive licensing, we were unable to run evals on LLaMA-2; we do not report previously-published LLaMA-2 numbers for TruthfulQA, as we use different, non-comparable evaluation set-ups (we use MC2, where LLaMA-2 uses GPT-Judge).

above, though often at a much lower rate. Further, it is important to note that these tools are known to have many false positives (because they only match patterns and do not consider the context), meaning that our results are likely overestimates of the amount of personal data identified.

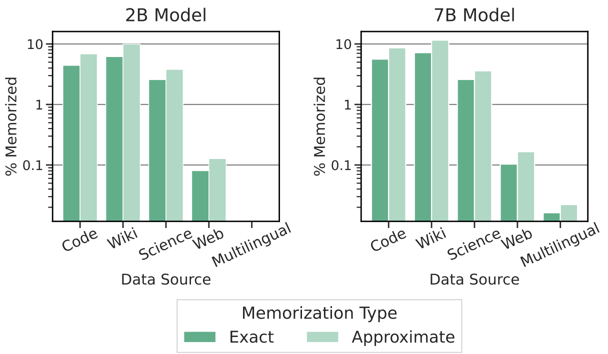

Figure 4 | Comparing exact and approximate memorization.

Approximate Memorization In Figure 4, we observe that roughly 50% more data is approximately memorized (note the log scale) and that this is nearly consistent across each of the different subcategories over the dataset.

Responsible Deployment

In line with previous releases of Google's AI technologies (Gemini Team, 2023; Kavukcuoglu et al., 2022), we follow a structured approach to responsible development and deployment of our models, in order to identify, measure, and manage foreseeable downstream societal impacts. As with our recent Gemini release, these are informed by prior academic literature on language model risks (Weidinger et al., 2021), findings from similar prior exercises conducted across the industry (Anil et al., 2023), ongoing engagement with experts internally and externally, and unstructured attempts to discover new model vulnerabilities.

Benefits

We believe that openness in AI science and technology can bring significant benefits. Open-sourcing is a significant driver of science and innovation, and a responsible practice in most circumstances. But this needs to be balanced against the risk of providing actors with the tools to cause harm now or in the future.

Google has long committed to providing broader access to successful research innovations (GraphCast, Transformer, BERT, T5, Word2Vec), and we believe that releasing Gemma into the AI development ecosystem will enable downstream developers to create a host of beneficial applications, in areas such as science, education and the arts. Our instruction-tuned offerings should encourage a range of developers to leverage Gemma's chat and code capabilities to support their own beneficial applications, while allowing for custom fine-tuning to specialize the model's capabilities for specific use cases. To ensure Gemma supports a wide range of developer needs, we are also releasing two model sizes to optimally support different environments, and have made these models available across a number of platforms (see Kaggle for details). Providing broad access to Gemma in this way should reduce the economic and technical barriers that newer ventures or independent developers face when incorporating these technologies into their workstreams.

As well as serving developers with our instruction-tuned models, we have also provided access to corresponding base pretrained models. By doing so, it is our intention to encourage further AI safety research and community innovation, providing a wider pool of models available to developers to build on various methods of transparency and interpretability research that the community has already benefited from (Pacchiardi et al., 2023; Zou et al., 2023).

Risks

In addition to bringing benefits to the AI development ecosystem, we are aware that malicious uses of LLMs, such as the creation of deepfake imagery, AI-generated disinformation, and illegal and disturbing material can cause harm on both an individual and institutional levels (Weidinger et al., 2021). Moreover, providing access to model weights, rather than releasing models behind an API, raises new challenges for responsible deployment.

First, we cannot prevent bad actors from fine tuning Gemma for malicious intent, despite their use being subject to Terms of Use that prohibit the use of Gemma models in ways that contravene our Gemma Prohibited Use Policy. However, we are cognizant that further work is required to build more robust mitigation strategies against intentional misuse of open systems, which Google DeepMind will continue to explore both internally and in collaboration with the wider AI community.

The second challenge we face is protecting developers and downstream users against the unintended behaviours of open models, including generation of toxic language or perpetuation of discriminatory social harms, model hallucinations and leakage of personally identifiable information. When deploying models behind an API, these risks can be reduced via various filtering methods.

Mitigations

Without this layer of defense for the Gemma family of models, we have endeavoured to safeguard against these risks by filtering and measuring biases in pre-training data in line with the Gemini approach, assessing safety through standardized AI safety benchmarks, internal red teaming to better understand the risks associated with external use of Gemma, and subjecting the models to rigorous ethics and safety evaluations, the results of which can be seen in 8.

While we've invested significantly in improving the model, we recognize its limitations. To ensure transparency for downstream users, we've published a detailed model card to provide researchers with a more comprehensive understanding of Gemma.

We have also released a Generative AI Responsible Toolkit to support developers to build AI responsibly. This encompasses a series of assets to help developers design and implement responsible AI best practices and keep their own users safe.

The relative novelty of releasing open weights models means new uses, and misuses, of these models are still being discovered, which is why Google DeepMind is committed to the continuous research and development of robust mitigation strategies alongside future model development.

Assessment

Ultimately, given the capabilities of larger systems accessible within the existing ecosystem, we believe the release of Gemma will have a negligible effect on the overall AI risk portfolio. In light of this, and given the utility of these models for research, auditing and downstream product development, we are confident that the benefit of Gemma to the AI community outweighs the risks described.

Going Forward

As a guiding principle, Google DeepMind strives to adopt assessments and safety mitigations proportionate to the potential risks from our models. In this case, although we are confident that Gemma models will provide a net benefit to the community, our emphasis on safety stems from the irreversible nature of this release. As the harms resulting from open models are not yet well defined, nor does an established evaluation framework for such models exist, we will continue to follow this precedent and take a measured and cautionary approach to open model development. As capabilities advance, we may need to explore extended testing, staggered releases or alternative access mechanisms to ensure responsible AI development.

As the ecosystem evolves, we urge the wider AI community to move beyond simplistic ’open vs. closed’ debates, and avoid either exaggerating or minimising potential harms, as we believe a nuanced, collaborative approach to risks and benefits is essential. At Google DeepMind we’re committed to developing high-quality evaluations and invite the community to join us in this effort for a deeper understanding of AI systems.

Discussion and Conclusion

We present Gemma, an openly available family of generative language models for text and code. Gemma advances the state of the art of openly available language model performance, safety, and responsible development.

In particular, we are confident that Gemma models will provide a net benefit to the community given our extensive safety evaluations and mitigations; however, we acknowledge that this release is irreversible and the harms resulting from open models are not yet well defined, so we continue to adopt assessments and safety mitigations proportionate to the potential risks of these models. In addition, our models outperform competitors on 6 standard safety benchmarks, and in human side-by-side evaluations.

Gemma models improve performance on a broad range of domains including dialogue, reasoning, mathematics, and code generation. Results on MMLU (64.3%) and MBPP (44.4%) demonstrate both the high performance of Gemma, as well as the continued headroom in openly available LLM performance.

Beyond state-of-the-art performance measures on benchmark tasks, we are excited to see what new use-cases arise from the community, and what new capabilities emerge as we advance the field together. We hope that researchers use Gemma to accelerate a broad array of research, and we hope that developers create beneficial new applications, user experiences, and other functionality.

Gemma benefits from many learnings of the Gemini model program including code, data, architecture, instruction tuning, reinforcement learning from human feedback, and evaluations. As discussed in the Gemini technical report, we reiterate a non-exhaustive set of limitations to the use of LLMs. Even with great performance on benchmark tasks, further research is needed to create robust, safe models that reliably perform as intended. Example further research areas include factuality, alignment, complex reasoning, and robustness to adversarial input. As discussed by Gemini, we note the need for more challenging and robust benchmarks.

Contributions and Acknowledgments

Core Contributors
Thomas Mesnard
Cassidy Hardin
Robert Dadashi
Surya Bhupatiraju
Shreya Pathak
Laurent Sifre
Morgane Rivière
Mihir Sanjay Kale
Juliette Love
Pouya Tafti
Léonard Hussenot

Contributors
Aakanksha Chowdhery
Adam Roberts
Aditya Barua
Alex Botev
Alex Castro-Ros
Ambrose Slone
Amélie Héliou
Andrea Tacchetti
Anna Bulanova
Antonia Paterson
Beth Tsai
Bobak Shahriari
Charline Le Lan
Christopher Choquette
Clément Crepy
Daniel Cer
Daphne Ippolito
David Reid
Elena Buchatskaya
Eric Ni
Eric Noland
Geng Yan
George Tucker
George-Christian Muraru
Grigory Rozhdestvenskiy
Henryk Michalewski
Ian Tenney
Ivan Grishchenko
Jacob Austin
James Keeling
Jane Labanowski
Jean-Baptiste Lespiau
Jeff Stanway
Jenny Brennan
Jeremy Chen
Johan Ferret
Justin Chiu
Justin Mao-Jones
Katherine Lee
Kathy Yu
Katie Millican
Lars Lowe Sjoesund
Lisa Lee
Lucas Dixon
Machel Reid
Maciej Mikuła
Mateo Wirth
Michael Sharman
Nikolai Chinaev
Nithum Thain
Olivier Bachem
Oscar Chang
Oscar Wahltinez
Paige Bailey
Paul Michel
Petko Yotov
Pier Giuseppe Sessa
Rahma Chaabouni
Ramona Comanescu
Reena Jana
Rohan Anil
Ross McIlroy
Ruibo Liu
Ryan Mullins
Samuel L Smith
Sebastian Borgeaud
Sertan Girgin
Sholto Douglas
Shree Pandya
Siamak Shakeri
Soham De
Ted Klimenko
Tom Hennigan
Vlad Feinberg
Wojciech Stokowiec
Yu-hui Chen
Zafarali Ahmed
Zhitao Gong

Product Management
Tris Warkentin
Ludovic Peran

Program Management
Minh Giang

Executive Sponsors
Clement Farabet
Oriol Vinyals
Jeff Dean
Koray Kavukcuoglu
Demis Hassabis
Zoubin Ghahramani
Douglas Eck
Joelle Barral
Fernando Pereira
Eli Collins

Leads
Armand Joulin
Noah Fiedel
Evan Senter

Tech Leads
Alek Andreev†
Kathleen Kenealy†

Acknowledgements
Our work is made possible by the dedication and efforts of numerous teams at Google. We would like to acknowledge the support from the following teams: Gemini, Gemini Safety, Gemini Infrastructure, Google Cloud, Google Research Responsible AI, Kaggle, Keras.

Special thanks and acknowledgment to Adrian Hutter, Andreas Terzis, Andrei Kulik, Angelos Filos, Anushan Fernando, Aurelien Boffy, Clement Crepy, Danila Sinopalnikov, Edouard Leurent, Gabriela Surita, Geoffrey Cideron, Jilin Chen, Karthik Raveendran, Kathy Meier-Hellstern, Kehang Han, Kevin Robinson, Kritika Muralidharan, Le Hou, Leonard Berrada, Lev Proleev, Marie Pellat, Mark Sherwood, Matt Hoffman, Matthias Grundmann, Nicola De Cao, Nikola Momchev, Nino Vieillard, Noah Constant, Peter Liu, Piotr Stanczyk, Qiao Zhang, Ruba Haroun, Seliem El-Sayed, Siddhartha Brahma, Tianhe (Kevin) Yu, Tom Le Paine, Yingjie Miao, Yuanzhong Xu, and Yuting Sun.

† equal contribution.

References

E. Almazrouei, H. Alobeidli, A. Alshamsi, A. Cappelli, R. Cojocaru, M. Debbah, Étienne Goffinet, D. Hesslow, J. Launay, Q. Malartic, D. Mazzotta, B. Noune, B. Pannier, and G. Penedo. The falcon series of open language models, 2023.

D. Amodei, C. Olah, J. Steinhardt, P. Christiano, J. Schulman, and D. Mané. Concrete problems in AI safety. *arXiv preprint*, 2016.

R. Anil, A. M. Dai, O. Firat, M. Johnson, D. Lepikhin, A. Passos, S. Shakeri, E. Taropa, P. Bailey, Z. Chen, et al. Palm 2 technical report. *arXiv preprint arXiv:2305.10403*, 2023.

J. Austin, A. Odena, M. I. Nye, M. Bosma, H. Michalewski, D. Dohan, E. Jiang, C. J. Cai, M. Terry, Q. V. Le, and C. Sutton. Program synthesis with large language models. *CoRR*, abs/2108.07732, 2021. URL https://arxiv.org/abs/2108.07732.

Y. Bai, S. Kadavath, S. Kundu, A. Askell, J. Kernion, A. Jones, A. Chen, A. Goldie, A. Mirhoseini, C. McKinnon, C. Chen, C. Olsson, C. Olah, D. Hernandez, D. Drain, D. Ganguli, D. Li, E. Tran-Johnson, E. Perez, J. Kerr, J. Mueller, J. Ladish, J. Landau, K. Ndousse, K. Lukosuite, L. Lovitt, M. Sellitto, N. Elhage, N. Schiefer, N. Mercado, N. DasSarma, R. Lasenby, R. Larson, S. Ringer, S. Johnston, S. Kravec, S. E. Showk, S. Fort, T. Lanham, T. Telleen-Lawton, T. Conerly, T. Henighan, T. Hume, S. R. Bowman, Z. Hatfield-Dodds, B. Mann, D. Amodei, N. Joseph, S. McCandlish, T. Brown, and J. Kaplan. Constitutional ai: Harmlessness from ai feedback, 2022.

P. Barham, A. Chowdhery, J. Dean, S. Ghemawat, S. Hand, D. Hurt, M. Isard, H. Lim, R. Pang, S. Roy, B. Saeta, P. Schuh, R. Sepassi, L. E. Shafey, C. A. Thekkath, and Y. Wu. Pathways: Asynchronous distributed dataflow for ml, 2022.

Y. Bisk, R. Zellers, R. L. Bras, J. Gao, and Y. Choi. PIQA: reasoning about physical commonsense in natural language. *CoRR*, abs/1911.11641, 2019. URL http://arxiv.org/abs/1911.11641.

R. A. Bradley and M. E. Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39, 1952.

N. Carlini, D. Ippolito, M. Jagielski, K. Lee, F. Tramer, and C. Zhang. Quantifying memorization across neural language models. *arXiv preprint arXiv:2202.07646*, 2022.

M. Chen, J. Tworek, H. Jun, Q. Yuan, H. P. de Oliveira Pinto, J. Kaplan, H. Edwards, Y. Burda, N. Joseph, G. Brockman, A. Ray, R. Puri, G. Krueger, M. Petrov, H. Khlaaf, G. Sastry, P. Mishkin, B. Chan, S. Gray, N. Ryder, M. Pavlov, A. Power, L. Kaiser, M. Bavarian, C. Winter, P. Tillet, F. P. Such, D. Cummings, M. Plappert, F. Chantzis, E. Barnes, A. Herbert-Voss, W. H. Guss, A. Nichol, A. Paino, N. Tezak, J. Tang, I. Babuschkin, S. Balaji, S. Jain, W. Saunders, C. Hesse, A. N. Carr, J. Leike, J. Achiam, V. Misra, E. Morikawa, A. Radford, M. Knight, M. Brundage, M. Murati, K. Mayer, P. Welinder, B. McGrew, D. Amodei, S. McCandlish, I. Sutskever, and W. Zaremba. Evaluating large language models trained on code. *CoRR*, abs/2107.03374, 2021. URL https://arxiv.org/abs/2107.03374.

A. Chowdhery, S. Narang, J. Devlin, M. Bosma, G. Mishra, A. Roberts, P. Barham, H. W. Chung, C. Sutton, S. Gehrmann, P. Schuh, K. Shi, S. Tsvyashchenko, J. Maynez, A. Rao, P. Barnes, Y. Tay, N. Shazeer, V. Prabhakaran, E. Reif, N. Du, B. Hutchinson, R. Pope, J. Bradbury, J. Austin, M. Isard, G. Gur-Ari, P. Yin, T. Duke, A. Levskaya, S. Ghemawat, S. Dev, H. Michalewski, X. Garcia, V. Misra, K. Robinson, L. Fedus, D. Zhou, D. Ippolito, D. Luan, H. Lim, B. Zoph, A. Spiridonov, R. Sepassi, D. Dohan, S. Agrawal, M. Omernick, A. M. Dai, T. S. Pillai, M. Pellat, A. Lewkowycz, E. Moreira, R. Child, O. Polozov, K. Lee, Z. Zhou, X. Wang, B. Saeta, M. Diaz, O. Firat, M. Catasta, J. Wei, K. Meier-Hellstern, D. Eck, J. Dean, S. Petrov, and N. Fiedel. Palm: Scaling language modeling with pathways, 2022.

P. F. Christiano, J. Leike, T. Brown, M. Martic, S. Legg, and D. Amodei. Deep reinforcement learning from human preferences. *Advances in Neural Information Processing Systems*, 30, 2017.

C. Clark, K. Lee, M. Chang, T. Kwiatkowski, M. Collins, and K. Toutanova. Boolq: Exploring the surprising difficulty of natural yes/no questions. *CoRR*, abs/1905.10044, 2019. URL http://arxiv.org/abs/1905.10044.

P. Clark, I. Cowhey, O. Etzioni, T. Khot, A. Sabharwal, C. Schoenick, and O. Tafjord. Think you have solved question answering? try arc, the ai2 reasoning challenge, 2018.

K. Cobbe, V. Kosaraju, M. Bavarian, M. Chen, H. Jun, L. Kaiser, M. Plappert, J. Tworek, J. Hilton, R. Nakano, C. Hesse, and J. Schulman. Training verifiers to solve math word problems. *CoRR*, abs/2110.14168, 2021. URL https://arxiv.org/abs/2110.14168.

J. Dean, G. Corrado, R. Monga, K. Chen, M. Devin, M. Mao, M. a. Ranzato, A. Senior, P. Tucker, K. Yang, Q. Le, and A. Ng. Large scale distributed deep networks. In F. Pereira, C. Burges, L. Bottou, and K. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012. URL https://proceedings.neurips.cc/paper_files/paper/2012/file/6aca97005c68f1206823815f66102863-Paper.pdf.

J. Devlin, M. Chang, K. Lee, and K. Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805, 2018. URL http://arxiv.org/abs/1810.04805.

Gemini Team. Gemini: A family of highly capable multimodal models, 2023.

D. Hendrycks, C. Burns, S. Basart, A. Zou, M. Mazeika, D. Song, and J. Steinhardt. Measuring massive multitask language understanding. *CoRR*, abs/2009.03300, 2020. URL https://arxiv.org/abs/2009.03300.

D. Hendrycks, C. Burns, S. Kadavath, A. Arora, S. Basart, E. Tang, D. Song, and J. Steinhardt. Measuring mathematical problem solving with the math dataset. *NeurIPS*, 2021.

D. Ippolito, F. Tramèr, M. Nasr, C. Zhang, M. Jagielski, K. Lee, C. A. Choquette-Choo, and N. Carlini. Preventing verbatim memorization in language models gives a false sense of privacy. *arXiv preprint arXiv:2210.17546*, 2022.

A. Q. Jiang, A. Sablayrolles, A. Mensch, C. Bamford, D. S. Chaplot, D. de las Casas, F. Bressand, G. Lengyel, G. Lample, L. Saulnier, L. R. Lavaud, M.-A. Lachaux, P. Stock, T. L. Scao, T. Lavril, T. Wang, T. Lacroix, and W. E. Sayed. Mistral 7b, 2023.

M. Joshi, E. Choi, D. S. Weld, and L. Zettlemoyer. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. *CoRR*, abs/1705.03551, 2017. URL http://arxiv.org/abs/1705.03551.

K. Kavukcuoglu, P. Kohli, L. Ibrahim, D. Bloxwich, and S. Brown. How our principles helped define alphafold's release, 2022.

T. Kudo and J. Richardson. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In E. Blanco and W. Lu, editors, *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, Brussels, Belgium, Nov. 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-2012. URL https://aclanthology.org/D18-2012.

S. Kudugunta, I. Caswell, B. Zhang, X. Garcia, C. A. Choquette-Choo, K. Lee, D. Xin, A. Kusupati, R. Stella, A. Bapna, et al. Madlad-400: A multilingual and document-level large audited dataset. *arXiv preprint arXiv:2309.04662*, 2023.

T. Kwiatkowski, J. Palomaki, O. Redfield, M. Collins, A. Parikh, C. Alberti, D. Epstein, I. Polosukhin, J. Devlin, K. Lee, K. Toutanova, L. Jones, M. Kelcey, M.-W. Chang, A. M. Dai, J. Uszkoreit, Q. Le, and S. Petrov. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466, 2019. doi: 10.1162/tacl_a_00276. URL https://aclanthology.org/Q19-1026.

Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *nature*, 521(7553):436–444, 2015.

T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space. In Y. Bengio and Y. LeCun, editors, *1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings*, 2013. URL http://arxiv.org/abs/1301.3781.

M. Nasr, N. Carlini, J. Hayase, M. Jagielski, A. F. Cooper, D. Ippolito, C. A. Choquette-Choo, E. Wallace, F. Tramèr, and K. Lee. Scalable extraction of training data from (production) language models. *arXiv preprint arXiv:2311.17035*, 2023.

L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray, et al. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35, 2022.

L. Pacchiardi, A. J. Chan, S. Mindermann, I. Moscovitz, A. Y. Pan, Y. Gal, O. Evans, and J. Brauner. How to catch an ai liar: Lie detection in black-box llms by asking unrelated questions, 2023.

D. Paperno, G. Kruszewski, A. Lazaridou, Q. N. Pham, R. Bernardi, S. Pezzelle, M. Baroni, G. Boleda, and R. Fernández. The LAMBADA dataset: Word prediction requiring a broad discourse context. *CoRR*, abs/1606.06031, 2016. URL http://arxiv.org/abs/1606.06031.

C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *CoRR*, abs/1910.10683, 2019. URL http://arxiv.org/abs/1910.10683.

A. Roberts, H. W. Chung, A. Levskaya, G. Mishra, J. Bradbury, D. Andor, S. Narang, B. Lester, C. Gaffney, A. Mohiuddin, C. Hawthorne, A. Lewkowycz, A. Salcianu, M. van Zee, J. Austin, S. Goodman, L. B. Soares, H. Hu,

S. Tsvyashchenko, A. Chowdhery, J. Bastings, J. Bulian, X. Garcia, J. Ni, A. Chen, K. Kenealy, J. H. Clark, S. Lee, D. Garrette, J. Lee-Thorp, C. Raffel, N. Shazeer, M. Ritter, M. Bosma, A. Passos, J. Maitin-Shepard, N. Fiedel, M. Omernick, B. Saeta, R. Sepassi, A. Spiridonov, J. Newlan, and A. Gesmundo. Scaling up models and data with `t5x` and `seqio`, 2022.

A. Roberts, H. W. Chung, G. Mishra, A. Levskaya, J. Bradbury, D. Andor, S. Narang, B. Lester, C. Gaffney, A. Mohiuddin, et al. Scaling up models and data with t5x and seqio. *Journal of Machine Learning Research*, 24(377):1–8, 2023.

K. Sakaguchi, R. L. Bras, C. Bhagavatula, and Y. Choi. WINOGRANDE: an adversarial winograd schema challenge at scale. *CoRR*, abs/1907.10641, 2019. URL http://arxiv.org/abs/1907.10641.

M. Sap, H. Rashkin, D. Chen, R. L. Bras, and Y. Choi. Socialiqa: Commonsense reasoning about social interactions. *CoRR*, abs/1904.09728, 2019. URL http://arxiv.org/abs/1904.09728.

N. Shazeer. Fast transformer decoding: One write-head is all you need. *CoRR*, abs/1911.02150, 2019. URL http://arxiv.org/abs/1911.02150.

N. Shazeer. GLU variants improve transformer. *CoRR*, abs/2002.05202, 2020. URL https://arxiv.org/abs/2002.05202.

J. M. V. Skalse, N. H. R. Howe, D. Krasheninnikov, and D. Krueger. Defining and characterizing reward gaming. In *NeurIPS*, 2022.

J. Su, Y. Lu, S. Pan, B. Wen, and Y. Liu. Roformer: Enhanced transformer with rotary position embedding. *CoRR*, abs/2104.09864, 2021. URL https://arxiv.org/abs/2104.09864.

I. Sutskever, O. Vinyals, and Q. V. Le. Sequence to sequence learning with neural networks. *CoRR*, abs/1409.3215, 2014. URL http://arxiv.org/abs/1409.3215.

M. Suzgun, N. Scales, N. Schärli, S. Gehrmann, Y. Tay, H. W. Chung, A. Chowdhery, Q. V. Le, E. H. Chi, D. Zhou, and J. Wei. Challenging big-bench tasks and whether chain-of-thought can solve them, 2022.

A. Talmor, J. Herzig, N. Lourie, and J. Berant. Commonsenseqa: A question answering challenge targeting commonsense knowledge, 2019.

H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, A. Rodriguez, A. Joulin, E. Grave, and G. Lample. Llama: Open and efficient foundation language models, 2023a.

H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale, D. Bikel, L. Blecher, C. C. Ferrer, M. Chen, G. Cucurull, D. Esiobu, J. Fernandes, J. Fu, W. Fu, B. Fuller, C. Gao, V. Goswami, N. Goyal, A. Hartshorn, S. Hosseini, R. Hou, H. Inan, M. Kardas, V. Kerkez, M. Khabsa, I. Kloumann, A. Korenev, P. S. Koura, M.-A. Lachaux, T. Lavril, J. Lee, D. Liskovich, Y. Lu, Y. Mao, X. Martinet, T. Mihaylov, P. Mishra, I. Molybog, Y. Nie, A. Poulton, J. Reizenstein, R. Rungta, K. Saladi, A. Schelten, R. Silva, E. M. Smith, R. Subramanian, X. E. Tan, B. Tang, R. Taylor, A. Williams, J. X. Kuan, P. Xu, Z. Yan, I. Zarov, Y. Zhang, A. Fan, M. Kambadur, S. Narang, A. Rodriguez, R. Stojnic, S. Edunov, and T. Scialom. Llama 2: Open foundation and fine-tuned chat models, 2023b.

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need. *CoRR*, abs/1706.03762, 2017. URL http://arxiv.org/abs/1706.03762.

J. Wei, X. Wang, D. Schuurmans, M. Bosma, E. H. Chi, Q. Le, and D. Zhou. Chain of thought prompting elicits reasoning in large language models. *CoRR*, abs/2201.11903, 2022. URL https://arxiv.org/abs/2201.11903.

L. Weidinger, J. Mellor, M. Rauh, C. Griffin, J. Uesato, P. Huang, M. Cheng, M. Glaese, B. Balle, A. Kasirzadeh, Z. Kenton, S. Brown, W. Hawkins, T. Stepleton, C. Biles, A. Birhane,

J. Haas, L. Rimell, L. A. Hendricks, W. Isaac, S. Legassick, G. Irving, and I. Gabriel. Ethical and social risks of harm from language models. *CoRR*, abs/2112.04359, 2021. URL https://arxiv.org/abs/2112.04359.

R. J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8, 1992.

XLA. Xla: Optimizing compiler for tensorflow, 2019. URL https://www.tensorflow.org/xla.

Y. Xu, H. Lee, D. Chen, B. A. Hechtman, Y. Huang, R. Joshi, M. Krikun, D. Lepikhin, A. Ly, M. Maggioni, R. Pang, N. Shazeer, S. Wang, T. Wang, Y. Wu, and Z. Chen. GSPMD: general and scalable parallelization for ML computation graphs. *CoRR*, abs/2105.04663, 2021. URL https://arxiv.org/abs/2105.04663.

B. Zhang and R. Sennrich. Root mean square layer normalization. *CoRR*, abs/1910.07467, 2019. URL http://arxiv.org/abs/1910.07467.

L. Zheng, W.-L. Chiang, Y. Sheng, S. Zhuang, Z. Wu, Y. Zhuang, Z. Lin, Z. Li, D. Li, E. P. Xing, H. Zhang, J. E. Gonzalez, and I. Stoica. Judging llm-as-a-judge with mt-bench and chatbot arena, 2023.

W. Zhong, R. Cui, Y. Guo, Y. Liang, S. Lu, Y. Wang, A. Saied, W. Chen, and N. Duan. Agieval: A human-centric benchmark for evaluating foundation models, 2023.

A. Zou, L. Phan, S. Chen, J. Campbell, P. Guo, R. Ren, A. Pan, X. Yin, M. Mazeika, A.-K. Dombrowski, S. Goel, N. Li, M. J. Byun, Z. Wang, A. Mallen, S. Basart, S. Koyejo, D. Song, M. Fredrikson, J. Z. Kolter, and D. Hendrycks. Representation engineering: A top-down approach to ai transparency, 2023.