llm-jp-eval: 日本語大規模言語モデルの自動評価ツール

Namgi Han [1] 植田 暢大 [2*] 大嶽 匡俊 [1*] 勝又 智 [3*] 鎌田 啓輔 [4*] 清丸 寛一 [2*] 児玉 貴志 [2*]
菅原 朔 [5*] Bowen Chen [1*] 松田 寛 [6*] 宮尾 祐介 [1*] 村脇 有吾 [2*] 劉 弘毅 [7*]
[1] 東京大学 [2] 京都大学 [3] 株式会社レトリバ [4] Weights & Biases Japan
[5] 国立情報学研究所 [6] 株式会社リクルート [7] 株式会社 Citadel AI
{hng88,otake,bwchen,yusuke}@is.s.u-tokyo.ac.jp
{ueda,kiyomaru,kodama,murawaki}@nlp.ist.i.kyoto-u.ac.jp
satoru.katsumata@retrieva.jp keisuke.kamata@wandb.com
saku@nii.ac.jp hiroshi_matsuda@megagon.ai koki@citadel.co.jp

概要

日本語の大規模モデルが次々と発表される中，その自動評価が重要性を増している．本稿では，日本語の大規模言語モデルに対する評価ベンチマーク llm-jp-eval を提案する．llm-jp-eval は 8 カテゴリ，計 12 個の日本語の自然言語処理の公開評価データを用いて，言語モデルの生成結果を自動的に評価する．評価は全て生成問題に基づくもので，既存の評価データセットにプロンプト形式を適用して自動変換した問題に対する言語モデルの回答を評価する．本稿では llm-jp-eval を用いて様々な日本語大規模言語モデルを評価した結果を報告し，既存研究の知見に照らして議論する．

1 はじめに

ChatGPT をはじめとする大規模言語モデルの成功は全世界に大きな影響をもたらした．日本も例外でなく，様々な日本語の大規模言語モデルが発表され続けている．これにともない，大規模言語モデルの性能評価が重要性を増している．

海外では大規模言語モデルが発表される際に，テクニカルレポートで既存の評価データセットを網羅した評価結果を報告する傾向が見られる．[1)] また Big-Bench [1] をはじめとする，大規模言語モデルの能力を測るためのベンチマークも充実している．それに比べ，日本語では JGLUE [2] が構築されているとはいえ，他の大規模言語モデルに対する評価ベンチマークは少ない．

本研究では，既存研究で提案されている 12 個の日本語の評価データセットを用いて大規模言語モデルの性能を評価するベンチマーク llm-jp-eval を提案する．llm-jp-eval は，既存の評価データセットにプロンプト形式を適用することで全ての問題を生成問題に変換し，言語モデルが生成した回答に基づき評価を行う．llm-jp-eval を用いた評価を公開大規模モデルに適用した結果，パラメータの数と日本語の継続学習が評価スコアに影響を与えることを確認できた．llm-jp-eval は Apache License 2.0 のもとオープンソースのソフトウェアとして公開されている．[2)]

2 関連研究

大規模言語モデルの評価ベンチマークは英語・中国語を対象として多くのものが提案されている．Chang ら [12] によると現時点で 40 個以上のベンチマークが存在し，その数は今なお増え続けている．また，これらの評価ベンチマークは自然言語推論をはじめとした伝統的な自然言語処理の課題はもちろん，自動翻訳やコード生成などの生成問題から，社会的バイアスや信頼性などの安全性検証まで幅広くカバーしている．例えば大規模言語モデルの研究開発が活発化する前に評価ベンチマークとして提案されていた GLUE [13] は 9 つのタスクで構成され，生成問題や安全性検証などの課題は含まれていない．それに比べ，大規模言語モデルの代表的な評価ベンチマークである Big-Bench は 200 以上の課題で構成され，GLUE が扱っていない様々な言語理解能力を評価している．

日本語の大規模言語モデルは，2022 年 1 月に 1.3B

* 第 2 著者以降は姓名の五十音順

1) GPT-4 のテクニカルレポートが良い例である：https://cdn.openai.com/papers/gpt-4.pdf

2) この論文の内容は以下のページで公開されている v1.2.0 を元に作成された：https://github.com/llm-jp/llm-jp-eval

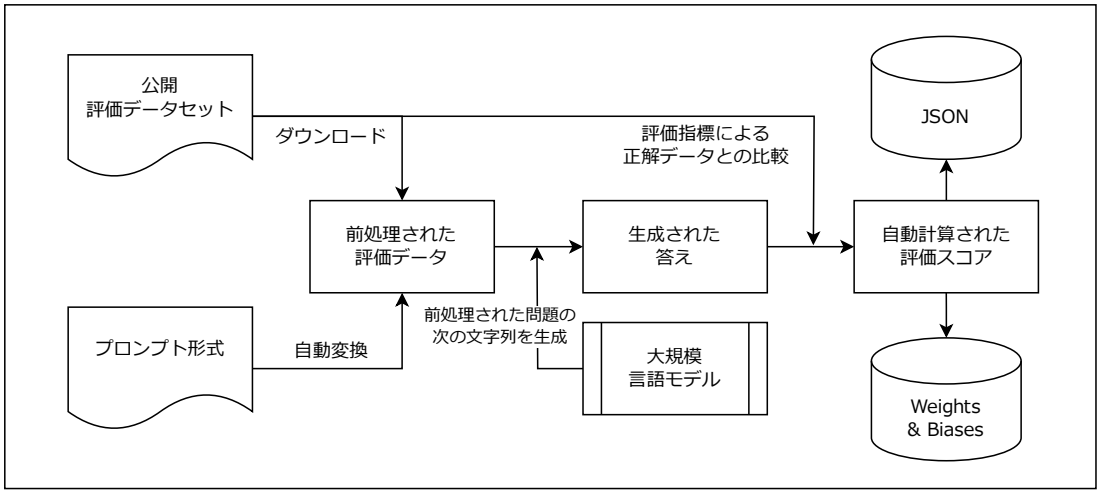

図 1　llm-jp-eval の評価フレームワーク．

表 1　llm-jp-eval が対応している評価データセット．

カテゴリー	名前	出処	ライセンス	評価指標
Natural Language Inference (NLI)	Jamp	[3]	CC BY-SA 4.0	Exact Match
	JaNLI	[4]	CC BY-SA 4.0	Exact Match
	JNLI	[2]	CC BY-SA 4.0	Exact Match
	JSeM	[5]	BSD 3-Clause	Exact Match
	JSICK	[6]	CC BY-SA 4.0	Exact Match
Question Answering (QA)	JEMHopQA	[7]	CC BY-SA 4.0	Char. F1
	NIILC	[8]	CC BY-SA 4.0	Char. F1
Reading Comprehension (RC)	JSQuAD	[2]	CC BY-SA 4.0	Char. F1
Multiple Choice question answering (MC)	JCommonsenseQA	[2]	CC BY-SA 4.0	Exact Match
Entity Linking (EL)	chABSA	[9]	CC BY 4.0	Set F1
Fundamental Analysis (FA)	Wikipedia Annotated Corpus	[10]	CC BY-SA 4.0	Set F1
Mathematical Reasoning (MR)	MAWPS	[11]	Apache-2.0	Exact Match
Semantic Textual Similarity (STS)	JSTS	[2]	CC BY-SA 4.0	Pearson/Spearman Coef.

パラメータの言語モデルを公開した rinna [14] をはじめ，様々なモデルが公開されるようになった．2023 年は特にその傾向が強く，ELYZA [15]，Japanese Stable LM [16]，LLM-jp[3]，OpenCALM[4]，PLaMo [17]，rinna の 3.6B と 4B パラメータのモデル[5]，stockmark[6]，Weblab[7]，Swallow [18] などが発表された．

しかし，日本語の大規模言語モデルの性能を検証するための評価ベンチマークの整備は追いついていない．日本語の大規模言語モデルの評価に用いられる評価ベンチマークとして，JGLUE [2] の他に JP Language Model Evaluation Harness[8] がある．JP Language Model Evaluation Harness は JGLUE を取り入れつつ，JAQKET [19]，JaQuAD [20]，JBLiMP [21] のような日本語の評価データセット，そして MGSM [22]，WikiLingua [23] XL-Sum [24]，XWinograd [25] のような多言語評価データセットをサポートしている．しかし，JP Language Model Evaluation Harness は課題によって，評価を行う際に評価対象の大規模言語モデルの生成結果ではなく，出力ラベルの対数尤度を使う場合がある．[9] そのため，純粋に大規模言語モデルの生成結果で性能を評価していないという限界があった．

大規模言語モデルの対数尤度ではなく，生成結果を評価する評価ベンチマークが求められる中，JGLUE に対する生成結果を評価した Nejumi リーダーボード[10] が提案された．しかし，Nejumi リー

3）https://huggingface.co/llm-jp
4）https://huggingface.co/cyberagent/open-calm-7b
5）https://huggingface.co/rinna/japanese-gpt-neox-3.6b
https://huggingface.co/rinna/bilingual-gpt-neox-4b
6）https://huggingface.co/stockmark/gpt-neox-japanese-1.4b
7）https://huggingface.co/matsuo-lab/weblab-10b
8）https://github.com/Stability-AI/lm-evaluation-harness/tree/jp-stable
9）https://note.com/wandb_jp/n/n2464e3d85c1a
10）https://wandb.ai/wandb/LLM_evaluation_Japan/reports/LLM-JGLUE---Vmlldzo0NTUzMDE2

```
以下は、タスクを説明する指示と、文脈のある入力の組み合わせです。要求を適切に満たす応答を書きなさい。

### 指示:
前提と仮説の関係をentailment、contradiction、neutralの中から回答してください。それ以外には何も含めないことを厳守してください。

制約：
- 前提から仮説が、時間関係上導出可能である場合はentailmentと出力
- 前提と仮説が両立しえない場合はcontradictionと出力
- そのいずれでもない場合はneutralと出力

### 入力:
前提：8月以来、ゾーイは雑誌に発表している。現在、10月である。
仮説：ゾーイは9月には雑誌に発表していた。

### 応答:
```

図 2　llm-jp-eval で自動変換された自然言語推論の問題の例.

ダーボードには JGLUE だけが対象という限界があり，後に JGLUE 以外の評価データセットを新たに取り込んだ Nejumi リーダーボード Neo[11] へと改良されたが，幅広い評価データセットを用いた評価は依然として必要とされている．

大規模言語モデルの出力を評価データセットが提供する正解と比較して評価するのではなく，オープンクエスチョンに対する大規模言語モデルの生成結果を GPT-4 のような相対的に強力な大規模言語モデルに評価させる手法も注目されている [26]．決まった回答を用意することが困難なこの系列の日本語の評価ベンチマークとして，Japanese MT-Bench[12]，Japanese Vicuna QA[13]，Rakuda ベンチマーク[14] などがある．しかし，これらの評価ベンチマークは特定の大規模言語モデルに依存しているため，回答の与え方，回答の冗長さ，評価を行うモデルと評価対象のモデルの類似性などのバイアスに影響された評価結果になる可能性がある [26].

3　llm-jp-eval

本研究では，大規模言語モデルの生成結果に基づき評価を行うベンチマーク llm-jp-eval を提案する．llm-jp-eval の評価フレームワークを図 1 に示す．

lllm-jp-eval は既存研究で提案された日本語の評価データセットを使用する．表 1 に llm-jp-eval が対応している評価データセットを示す．評価データセットには商用利用が可能なライセンスのもと公開されているものを選定している．llm-jp-eval は評価データを含んでいないため，評価データは公開先からダウンロードする必要があることに注意されたい．

llm-jp-eval は全ての評価データセットを生成問題として定式化し，回答として生成された文字列と正解の文字列を比較することで評価を行う．llm-jp-eval が対応している評価データセットには生成問題として設計されてないものが多く含まれる．そのため，評価データセットはデータセットごとに設計したプロンプト形式[15] を適用することで生成問題に変換する．変換された評価データセットの問題の具体例を図 2 に示す．llm-jp-eval はこのポリシーによって，大規模言語モデルが回答として生成した文字列を自動評価するという統一された手法を全ての評価データセットに適用する．

評価スコアは評価データセットに用意されている正解データ，大規模言語モデルの生成結果，そして評価データセットごとに決まっている評価指標によって計算される．また，llm-jp-eval はカテゴリーごとに平均スコアを取った結果も提供している．この結果は JSON ファイルとしてローカルに出力される上，Weights & Biases [28] を通じてクラウド上で管理する機能もサポートしている．

4　llm-jp-eval による評価例

本稿では日本語の公開大規模言語モデルの一部と，継続学習のベースモデルとして使われている海外の大規模言語モデルの評価結果を示す．大規模言語モデルは Hugging Face Hub[16] に公開されているものを用いた．ハイパーパラメータ[17] とプロンプト形式は同じもので統一した．評価は全て 4-shots で行った．各カテゴリーのスコアはそのカテゴリーに所属している評価データセットのスコアの平均で計算した．なお，平均スコア（AVR）の計算には STS を含めていない．これは STS が他のカテゴリーと異なり評価指標が相関係数で，-1 から 1 までの数値であるためである．

評価結果を表 2 に示す．その他のモデルを含むよ

11）https://wandb.me/nejumi
12）https://github.com/Stability-AI/FastChat/tree/jp-stable/fastchat/llm_judge
13）https://github.com/ku-nlp/ja-vicuna-qa-benchmark
14）https://yuzuai.jp/benchmark
15）基本的に Alpaca [27] のプロンプト形式に従う．
16）https://huggingface.co/models
17）公開されている llm-jp-eval のコードの初期値を使った．

表 2 llm-jp-eval の評価例．AVR は STS を除く評価スコアの平均．

モデル名	AVR	NLI	QA	RC	MC	EL	FA	MR	STS
cyberagent/open-calm-1b	0.148	0.269	0.213	0.222	0.217	0.087	0.023	0.006	-0.018
cyberagent/open-calm-3b	0.204	0.368	0.258	0.418	0.203	0.147	0.029	0.008	-0.022
cyberagent/open-calm-7b	0.224	0.256	0.366	0.564	0.198	0.159	0.015	0.008	0.000
llm-jp/llm-jp-1.3b-v1.0	0.253	0.310	0.304	0.557	0.205	0.304	0.072	0.018	0.000
llm-jp/llm-jp-13b-v1.0	0.343	0.349	0.468	0.721	0.206	0.340	0.189	0.130	-0.049
meta-llama/Llama-2-7b-hf	0.351	0.363	0.346	0.750	0.246	0.329	0.118	0.304	0.047
tokyotech-llm/Swallow-7b-hf	0.415	0.318	0.494	0.806	0.368	0.327	0.214	0.374	0.083
elyza/ELYZA-japanese-Llama-2-7b	0.433	0.401	0.421	0.791	0.509	0.351	0.097	0.462	0.130
stabilityai/japanese-stablelm-base-beta-7b	0.439	0.411	0.450	0.820	0.388	0.339	0.206	0.458	0.308
mistralai/Mistral-7B-v0.1	0.521	0.404	0.355	0.858	0.747	0.408	0.216	0.656	0.728
stabilityai/japanese-stablelm-base-gamma-7b	0.552	0.355	0.501	0.880	0.831	0.411	0.253	0.634	0.488

り詳細な評価結果は，Weights & Biases 上で公開しているリーダーボード[18]を参照されたい．

表 2 の上段はパラメータの違いによる評価スコアの違いを示す．OpenCALM と LLM-jp の両方で，パラメータの数が増えるごとに評価スコアも高くなる傾向が確認された．特に QA と RC でその傾向が強く，言語モデルが大規模であるほど与えられた問題に対する答えの生成能力が高くなるという既存の知見と合致する結果となった．

表 2 の下段は主に英語を対象としている海外の大規模言語モデルと，それに対して日本語の継続学習を行った日本語の大規模言語モデルの評価結果を示す．Swallow，ELYZA，Japanese Stable LM Beta は Llama2 に日本語のコーパスで継続学習を施した言語モデルである．Japanese Stable LM Gamma は Mistral を継続学習したモデルである．どちらの評価スコアも，日本語のコーパスで継続学習を行った言語モデルの方が高く，継続学習によって日本語の言語理解が改善されていることが分かる．また，こちらでも上と同じく QA と RC で評価スコアの向上が観測された．

しかし，パラメータの数の増加や継続学習の実施により，あらゆるタスクで一貫して性能が向上しているわけではない．NLI，FA，STS の評価スコアがその例である．これらのタスクはそもそも生成問題として設計されていないタスクという共通点はあるが，原因の詳細な分析は今後の課題とする．

5 おわりに

本研究では日本語の大規模言語モデルに対する評価ベンチマークである llm-jp-eval を構築した．既存のベンチマークと同じく，llm-jp-eval は公開評価データセットを評価対象として取り入れている．しかし，その評価データセットを全て生成問題とみなして評価しているという点で異なり，評価対象としているデータセットが全て商用利用可能なライセンスのもと公開されているため，研究はもちろん企業での大規模言語モデルの開発でも llm-jp-eval を取り込みやすいという利点がある[19]．また，llm-jp-eval による評価例から llm-jp-eval の評価スコアと大規模言語モデルに対する既存の知見を比較し，その結果を議論した．

英語を対象にした評価ベンチマークに比べると，llm-jp-eval 含め，日本語の評価ベンチマーク構築はまだ先が長い．まず足りないものとして，評価データセットの数と種類がある．Chang ら [12] は大規模言語モデルが評価されるべき能力として，伝統的な自然言語処理のタスクはもちろん，社会バイアスや毒性表現などに関わる倫理・信頼性，医療や応用タスクに関わるドメイン特化能力，理工学・社会科学のように実世界を理解する能力などをあげている．また，これらの能力を測るべく発表されている評価データセットも数多く報告している．しかし，日本語の評価データセットはまだ英語圏に比べて提案されているものが少なく，既存の評価データセットを束ねて評価ベンチマークとする動きも少ない．そのため，新たなデータセットの開発，英語圏の評価データセットの翻訳，多数の評価データセットを取り込む評価ベンチマークの提案などを，日本語の大規模言語モデルの評価に対する今後の課題にしたい．

18） https://wandb.me/llm-jp-leaderboard

19） 例えば前述した Nejumi リーダーボード Neo では llm-jp-eval も有効活用し，対応する日本語の評価データセットを増やしている．

謝辞

本研究の成果の一部は，データ活用社会創成プラットフォーム mdx を利用して得られたものです．

参考文献

[1] BIG bench authors. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. **Transactions on Machine Learning Research**, 2023.

[2] Kentaro Kurihara, Daisuke Kawahara, and Tomohide Shibata. JGLUE: Japanese general language understanding evaluation. In **Proceedings of the Thirteenth Language Resources and Evaluation Conference**, pp. 2957–2966, Marseille, France, June 2022. European Language Resources Association.

[3] Tomoki Sugimoto, Yasumasa Onoe, and Hitomi Yanaka. Jamp: Controlled Japanese temporal inference dataset for evaluating generalization capacity of language models. In **Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 4: Student Research Workshop)**, pp. 57–68, Toronto, Canada, July 2023. Association for Computational Linguistics.

[4] Hitomi Yanaka and Koji Mineshima. Assessing the generalization capacity of pre-trained language models through Japanese adversarial natural language inference. In **Proceedings of the 2021 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP (BlackboxNLP2021)**, 2021.

[5] 川添愛, 田中リベカ, 峯島宏次, 戸次大介. 形式意味論に基づく含意関係テストセット構築の方法論. 人工知能学会全国大会論文集 第 29 回 (2015), pp. 1K31–1K31. 一般社団法人 人工知能学会, 2015.

[6] Hitomi Yanaka and Koji Mineshima. Compositional evaluation on Japanese textual entailment and similarity. **Transactions of the Association for Computational Linguistics**, Vol. 10, pp. 1266–1284, 2022.

[7] 石井愛, 井之上直也, 関根聡. 根拠を説明可能な質問応答システムのための日本語マルチホップ QA データセット構築. 言語処理学会第 29 回年次大会論文集, 2023.

[8] 関根聡. 百科事典を対象とした質問応答システムの開発. 言語処理学会第９回年次大会, 2003.

[9] Takahiro Kubo and Hiroki Nakayama. chABSA: Aspect based sentiment analysis dataset in Japanese, 2018. https://github.com/chakki-works/chABSA-dataset/blob/master/doc/chabsa-aspect-based.pdf.

[10] 萩行正嗣, 河原大輔, 黒橋禎夫. 多様な文書の書き始めに対する意味関係タグ付きコーパスの構築とその分析. 自然言語処理, Vol. 21, No. 2, pp. 213–247, 2014.

[11] 堀尾海斗, 村田栄樹, 王昊, 井手竜也, 河原大輔, 天, 新里顕大, 中町礼文, 李聖哲, 佐藤敏紀. 日本語における chain-of-thought プロンプトの検証. 人工知能学会全国大会論文集, Vol. JSAI2023, pp. 3T1GS602–3T1GS602, 2023.

[12] Yupeng Chang, Xu Wang, Jindong Wang, Yuan Wu, Kaijie Zhu, Hao Chen, Linyi Yang, Xiaoyuan Yi, Cunxiang Wang, Yidong Wang, Wei Ye, Yue Zhang, Yi Chang, Philip S. Yu, Qiang Yang, and Xing Xie. A survey on evaluation of large language models. **arXiv preprint arXiv:2307.03109**, 2023.

[13] Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. GLUE: A multi-task benchmark and analysis platform for natural language understanding. 2019. In the Proceedings of ICLR.

[14] 趙天雨, 沢田慶. 日本語自然言語処理における事前学習モデルの公開. 人工知能学会研究会資料 言語・音声理解と対話処理研究会, Vol. 93, pp. 169–170, 2021.

[15] Akira Sasaki, Masato Hirakawa, Shintaro Horie, and Tomoaki Nakamura. ELYZA-japanese-Llama-2-7b, 2023.

[16] Meng Lee, Fujiki Nakamura, Makoto Shing, Paul McCann, Takuya Akiba, and Naoki Orii. Japanese StableLM Base Alpha 7B.

[17] Preferred Networks, Inc. PLaMo-13B, 2023.

[18] 藤井一喜, 中村泰士, Mengsay Loem, 飯田大貴, 大井聖也, 服部翔, 平井翔太, 水木栄, 横田理央, 岡崎直観. 継続事前学習による日本語に強い大規模言語モデルの構築. 言語処理学会第 30 回年次大会 (NLP2024), March 2024.

[19] 鈴木正敏, 鈴木潤, 松田耕史, 西田京介, 井之上直也. Jaqket: クイズを題材にした日本語 qa データセットの構築. 言語処理学会第 26 回年次大会, pp. 237–240, 2020.

[20] ByungHoon So, Kyuhong Byun, Kyungwon Kang, and Seongjin Cho. JaQuAD: Japanese Question Answering Dataset for Machine Reading Comprehension, 2022.

[21] Taiga Someya and Yohei Oseki. JBLiMP: Japanese benchmark of linguistic minimal pairs. In Andreas Vlachos and Isabelle Augenstein, editors, **Findings of the Association for Computational Linguistics: EACL 2023**, pp. 1581–1594, Dubrovnik, Croatia, May 2023. Association for Computational Linguistics.

[22] Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. **arXiv preprint arXiv:2110.14168**, 2021.

[23] Claire Cardie Faisal Ladhak, Esin Durmus and Kathleen McKeown. Wikilingua: A new benchmark dataset for multilingual abstractive summarization. In **Findings of EMNLP, 2020**, 2020.

[24] Tahmid Hasan, Abhik Bhattacharjee, Md. Saiful Islam, Kazi Mubasshir, Yuan-Fang Li, Yong-Bin Kang, M. Sohel Rahman, and Rifat Shahriyar. XL-sum: Large-scale multilingual abstractive summarization for 44 languages. In **Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021**, pp. 4693–4703, Online, August 2021. Association for Computational Linguistics.

[25] Niklas Muennighoff, Thomas Wang, Lintang Sutawika, Adam Roberts, Stella Biderman, Teven Le Scao, M Saiful Bari, Sheng Shen, Zheng-Xin Yong, Hailey Schoelkopf, Xiangru Tang, Dragomir Radev, Alham Fikri Aji, Khalid Almubarak, Samuel Albanie, Zaid Alyafeai, Albert Webson, Edward Raff, and Colin Raffel. Crosslingual generalization through multitask finetuning, 2022.

[26] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. Judging LLM-as-a-judge with MT-Bench and Chatbot Arena. **arXiv preprint arXiv:2306.05685**, 2023.

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

[28] Lukas Biewald. Experiment tracking with weights and biases, 2020. Software available from wandb.com.