

QWEN2 TECHNICAL REPORT

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, Jianxin Yang, Jin Xu, Jingren Zhou, Jinze Bai, Jinzheng He, Junyang Lin, Kai Dang, Keming Lu, Keqin Chen, Kexin Yang, Mei Li, Mingfeng Xue, Na Ni, Pei Zhang, Peng Wang, Ru Peng, Rui Men, Ruize Gao, Runji Lin, Shijie Wang, Shuai Bai, Sinan Tan, Tianhang Zhu, Tianhao Li, Tianyu Liu, Wenbin Ge, Xiaodong Deng, Xiaohuan Zhou, Xingzhang Ren, Xinyu Zhang, Xipin Wei, Xuancheng Ren, Xuejing Liu, Yang Fan, Yang Yao, Yichang Zhang, Yu Wan, Yunfei Chu, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, Zhifang Guo, and Zhihao Fan

Qwen Team, Alibaba Group*

ABSTRACT

This report introduces the Qwen2 series, the latest addition to our large language models and large multimodal models. We release a comprehensive suite of foundational and instruction-tuned language models, encompassing a parameter range from 0.5 to 72 billion, featuring dense models and a Mixture-of-Experts model. Qwen2 surpasses most prior open-weight models, including its predecessor Qwen1.5, and exhibits competitive performance relative to proprietary models across diverse benchmarks on language understanding, generation, multilingual proficiency, coding, mathematics, and reasoning.

The flagship model, Qwen2-72B, showcases remarkable performance: 84.2 on MMLU, 37.9 on GPQA, 64.6 on HumanEval, 89.5 on GSM8K, and 82.4 on BBH as a base language model. The instruction-tuned variant, Qwen2-72B-Instruct, attains 9.1 on MT-Bench, 48.1 on Arena-Hard, and 35.7 on LiveCodeBench. Moreover, Qwen2 demonstrates robust multilingual capabilities, proficient in approximately 30 languages, spanning English, Chinese, Spanish, French, German, Arabic, Russian, Korean, Japanese, Thai, Vietnamese, and more, underscoring its versatility and global reach.

To foster community innovation and accessibility, we have made the Qwen2 model weights openly available on Hugging Face¹ and ModelScope², and the supplementary materials including example code on GitHub³. These platforms also include resources for quantization, fine-tuning, and deployment, facilitating a wide range of applications and research endeavors.

*Authors are ordered alphabetically by the first name.

¹<https://huggingface.co/Qwen>

²<https://modelscope.cn/organization/qwen>

³<https://github.com/QwenLM/Qwen2>

CONTENTS

1	Introduction	3
2	Tokenizer & Model	3
2.1	Tokenizer	3
2.2	Model Architecture	4
2.2.1	Qwen2 Dense Model	4
2.2.2	Qwen2 Mixture-of-experts Model	4
2.2.3	Model Configuration	5
3	Pre-training	5
3.1	Pre-training Data	5
3.2	Long-context Training	6
4	Post-training	6
4.1	Post-training Data	6
4.1.1	Collaborative Data Annotation	7
4.1.2	Automated Data Synthesis	7
4.2	Supervised Fine-tuning	8
4.3	Reinforcement Learning from Human Feedback	8
5	Evaluation	8
5.1	Base Language Models	8
5.1.1	Core Capabilities	8
5.2	Instruction-tuned Model	12
5.2.1	Open Benchmark Evaluation	12
5.2.2	In-house Automatic Evaluation	14
5.2.3	Long Context Capabilities	15
5.2.4	Multilingual Evaluation	18
5.2.5	Safety & Responsibility	18
5.2.6	Contamination Analysis	19
6	Conclusion	20

1 INTRODUCTION

Following the emergence of ChatGPT (OpenAI, 2022), enthusiasm for large language models (LLMs) has escalated globally. The release of the Llama series (Touvron et al., 2023) has further ignited interests within the open-source community, particularly regarding GPT-level local LLMs. Recently, Claude-3 Opus (Anthropic, 2024) and GPT-4o (omni) (OpenAI, 2024), the updated model for ChatGPT, have ascended to the pinnacle of the Chatbot Arena (Chiang et al., 2024) in quick succession. This platform is well-regarded for its human evaluations of LLMs. Moreover, Llama-3 (AI@Meta, 2024) has emerged as the state-of-the-art open-weight model series, narrowing the performance gap with leading proprietary models and widely acknowledged as GPT-4-level. An increasing number of competitive LLMs are now pursuing advancements similar to those made by the GPT series from OpenAI. Many of these models, including Qwen (Bai et al., 2023a), Mistral (Jiang et al., 2023a), Gemma (Mesnard et al., 2024), etc., have been released in an open-weight manner.

Over recent months, we have successively introduced the Qwen series (Bai et al., 2023a) and progressed to Qwen1.5 (Qwen Team, 2024a). In the meantime, we have unveiled the vision-language model Qwen-VL (Bai et al., 2023b), and launched the audio-language model Qwen-Audio (Chu et al., 2023). In this work, we introduce the newest addition to the Qwen family of large language models and large multimodal models: **Qwen2**. Qwen2 is a series of LLMs, grounded in the Transformer architecture (Vaswani et al., 2017), trained using next-token prediction. The model series encompasses foundational, i.e., base language models, pre-trained but unaligned to human preferences, and instruction-tuned models, fine-tuned with single-turn and multi-turn instruction-following datasets suitable for chat and agent purposes. Our release comprises four dense models with parameter counts of 0.5 billion, 1.5 billion, 7 billion, and 72 billion, plus a Mixture-of-Experts (MoE) model with 57 billion parameters, of which 14 billion are activated for each token. The smaller models, specifically Qwen2-0.5B and Qwen2-1.5B, are designed for easy deployment on portable devices such as smartphones, earphones, and smart glasses. Conversely, the larger models cater to deployment across GPUs of varying scales.

All models were pre-trained on a high-quality, large-scale dataset comprising over 7 trillion tokens, covering a wide range of domains and languages. Compared to previous editions of Qwen, Qwen2 includes a broader spectrum of linguistic data, enhancing the quantity and quality of code and mathematics content. This enrichment is hypothesized to improve reasoning abilities of LLMs. Regarding post-training, all models underwent supervised fine-tuning and direct preference optimization (DPO, Rafailov et al., 2023), aligning them with human preferences through learning from human feedback. This process endows the models with the capability to follow instructions effectively.

모든 모델은 7T 이상의 토큰의 다양한 도메인과 언어로 학습됨

Post Training은 SFT, DPO로 학습

We have conducted a thorough evaluation of Qwen2, alongside a selection of baseline models including both open-weight and proprietary models accessible via API. Qwen2 outperforms competing models in evaluations of both fundamental language capabilities and instruction-tuned functionalities. Specifically, Qwen2-72B-Instruct, our instruction-tuned variant, scores 9.1 on MT-Bench (Zheng et al., 2023), 48.1 on Arena-Hard (Chiang et al., 2024), and 35.7 on LiveCodeBench (Jain et al., 2024). Meanwhile, Qwen2-72B, the base language model, achieves 84.2 on MMLU (Hendrycks et al., 2021a), 37.9 on GPQA (Rein et al., 2023), 64.6 on HumanEval (Chen et al., 2021), 89.5 on GSM8K (Cobbe et al., 2021), and 82.4 on BBH (Suzgun et al., 2023).

2 TOKENIZER & MODEL

This section introduces the tokenizer and model design of Qwen2. We detail the model architecture and configurations for different model sizes.

2.1 TOKENIZER

Following Qwen (Bai et al., 2023a), we employ the identical tokenizer based on byte-level byte-pair encoding. Notably, this tokenizer exhibits high encoding efficiency, as evidenced by its better compression rate relative to alternatives, facilitating the multilingual capabilities of Qwen2.

Qwen과 동일하게 Byte Level BPE 토큰라이저 사용

Models of all sizes employ a common vocabulary consisting of 151,643 regular tokens and 3 control tokens. For more information, please refer to Bai et al. (2023a). It should be noted that, owing to considerations in distributed training, the effective size for the embeddings is larger.

Vocab의 크기는 15만개

2.2 MODEL ARCHITECTURE

The Qwen2 series fundamentally constitute large language models based on the Transformer architecture, featuring self-attention with causal masks (Vaswani et al., 2017). Specifically, this series encompasses dense language models of 4 scales and a Mixture-of-Experts (MoE) model. We introduce the specifics of the dense models before delving into the MoE model's distinctive attributes.

4개의 Dense 모델과
MoE 모델 내놓음

2.2.1 QWEN2 DENSE MODEL

The architecture of the Qwen2 dense models comprises multiple Transformer layers, each equipped with causal attention mechanisms and feed-forward neural networks (FFNs). Key differences from Qwen are described below:

Grouped Query Attention We adopt Grouped Query Attention (GQA, Ainslie et al., 2023) instead of conventional multi-head attention (MHA). GQA optimizes KV cache usage during inference, significantly enhancing throughput. Detailed KV head configurations for various model sizes are reported in Section 2.2.3.

GQA
GQA를 사용하여 추론시 KV Cache가 최적화됨

Dual Chunk Attention with YARN To expand the context window of Qwen2, we implement Dual Chunk Attention (DCA, An et al., 2024), which segments long sequences into chunks of manageable lengths. If the input can be handled in a chunk, DCA produces the same result as the original attention. Otherwise, DCA facilitates effective capture of relative positional information between tokens within and across chunks, thereby improving long context performance. Moreover, we also employ YARN (Peng et al., 2023) to rescale the attention weights for better length extrapolation.

Dual Chunk
Long Input Sequence를 적절한 크기의 Chunk로 분할하여 처리함

만약 Input이 1개의 Chunk 안에서 처리 가능한 경우 기존 어텐션과 동일한 출력을 생성함.

그렇지 않은 경우 DCA는 Chunk 내 또는 Chunk간의 상대적인 Positional Information을 효과적으로 잡아냄

Moreover, we follow Qwen with the usage of SwiGLU (Dauphin et al., 2017) for activation, Rotary Positional Embeddings (RoPE, Su et al., 2024) for positional embedding, QKV bias (Su, 2023) for attention, RMSNorm (Jiang et al., 2023b) and pre-normalization for training stability.

YARN
Attention Weight를 Rescale하여 외삽 능력을 강화함

그외에도
- SwiGLU
- RoPE
- QKV Bias
- RMSNorm
등을 사용

2.2.2 QWEN2 MIXTURE-OF-EXPERTS MODEL

기존 Qwen1.5-MoE-A2.7B와 유사한 구조를 가짐

The architecture of Qwen2 MoE models closely mirrors that of Qwen1.5-MoE-A2.7B (Qwen Team, 2024c). As a substitute for the original FFN, the MoE FFN consists of n individual FFNs, each serving as an expert. Each token is directed to a specific expert E_i for computation based on probabilities assigned by a gated network G :

MoE모델은

n 개의 개별 FFN 네트워크를 Expert로 가짐

E_i : 개별 Expert

G : Expert를 Routing하는 Gate

$$\mathbf{p} = \text{softmax}(G(\mathbf{x})), \quad (1)$$

$$\mathbf{y} = \sum_{i \in \text{top}_k(\mathbf{p})} \mathbf{p}_i E_i(\mathbf{x}). \quad (2)$$

In the following, we present critical design considerations of Qwen2 MoE.

Expert Granularity The key structural difference between MoE models and dense models is that MoE layers incorporate multiple FFNs, each serving as an individual expert. Consequently, one straightforward strategy to transition from a dense architecture to an MoE architecture is to set the parameters of each expert equal to those of a single FFN from the original dense model. For example, transitioning from Mistral-7B (Jiang et al., 2023a) to Mixtral 8x7B (Jiang et al., 2024), involves activating two of the eight experts at a time. Differently, our model employs fine-grained experts (Dai et al., 2024), creating smaller-scale experts while activating a greater number of experts simultaneously. Given an equal total number of expert parameters and activated parameters, fine-grained experts offer a richer set of expert combinations. By leveraging these fine-grained experts, Qwen2 MoE facilitates more diverse and dynamic expert utilization, thereby enhancing overall performance and adaptability.

Dense => MoE로 변환하는 가장 편한 방법은 FFN을 Expert로 바꾸는 것임
예를 들어 Mistral의 경우, Mistral-7B의 FFN을 각 Expert로 바꿔서 7 x n 개의 MoE모델로 만드는 방법임

Fine-Grained Experts

예를 들면, 4B Expert 2개 대신 2B Expert 4개를 구성하는 방법 (DeepSeek MoE)
좀더 다양한 Expert를 구성할 수 있음

Expert Routing The design of expert routing mechanisms is crucial for enhancing the performance of MoE models. Recently, there has been a notable trend towards integrating both shared and routing-specific experts within MoE layers (Rajbhandari et al., 2022; Dai et al., 2024). We adopt this approach, as it facilitates the application of shared experts across various tasks while reserving others for selective use in specific routing scenarios. The introduction of shared and specialized experts offers a more adaptable and efficient method for developing MoE routing mechanisms.

Expert Routing
Routing Expert 외에 Shared Expert를 추가 (DeepSeek MoE)

Common Knowledge에 대한 것은 Shared Expert가 처리해줌
Shared Expert는 항상 활성화 됨

Table 1: **Architecture of Qwen2 dense and MoE models.** For MoE models, 57B-A14B denotes that the model has 57B parameters in total and for each token 14B parameters are active, the Intermediate size denotes that of each expert, and # Activated Experts excludes the shared experts.

Configuration	0.5B	1.5B	7B	72B	57B-A14B
Hidden Size	896	1,536	3,584	8,192	3,584
# Layers	24	28	28	80	28
# Query Heads	14	12	28	64	28
# KV Heads	2	2	4	8	4
Head Size	64	128	128	128	128
Intermediate Size	4,864	8,960	18,944	29,568	2,560
# Routed Experts	-	-	-	-	64
# Activated Experts	-	-	-	-	8
# Shared Experts	-	-	-	-	8
Embedding Tying	True	True	False	False	False
Vocabulary Size	151,646	151,646	151,646	151,646	151,646
# Trained Tokens	12T	7T	7T	7T	4.5T

MoE는 Dense모델의 Weight를 활용하여 초기화함

Expert Initialization We initialize the experts in a similar way to upcycling (Komatsuzaki et al., 2023), leveraging the weights of a dense model. In contrast, our approach emphasizes diversification among fine-grained experts to enhance the model’s representational breadth. Given the designated expert intermediate size h_E , the number of experts n , and the original FFN intermediate size h_{FFN} , the FFN is replicated $\lceil n \times h_E / h_{FFN} \rceil$ times. This replication ensures compatibility with the specified number of experts while accommodating any arbitrary expert intermediate size. To promote diversity within each FFN copy, parameters are shuffled along the intermediate dimension. This guarantees that each fine-grained expert exhibits unique characteristics, even across different FFN copies. Subsequently, these experts are extracted from the FFN copies, and the remaining dimensions are discarded. For each fine-grained expert, 50% of its parameters are randomly reinitialized. This process introduces additional stochasticity into expert initialization, potentially enhancing the model’s capacity for exploration during training.

예)
Dense의 Hidden = 80
각 Expert 크기 = 40
 $n = 8$
이면, $[40 \times 8 / 80] = 4$ 번 복제하여 초기화한다

2.2.3 MODEL CONFIGURATION

In the following, we provide the key configuration and information for the Qwen2 series.

The Qwen2 series consists of models of 5 sizes, which are Qwen2-0.5B, Qwen2-1.5B, Qwen2-7B, Qwen2-57B-A14B, and Qwen2-72B. Table 1 lists the hyper-parameters and important information, e.g., the number of pre-trained tokens. Particularly, Qwen2-57B-A14B is upscaled from Qwen2-7B. Notably, Qwen2 models demonstrate a substantially lower Key-Value (KV) size per token relative to Qwen1.5 models. This characteristic translates into a reduced memory footprint, particularly advantageous in long-context inference tasks.

Qwen1.5 대비 Qwen2는 KV Cache의 크기가 확연히 줄어들
따라서 Long-Context의 추론 과정에서 이점임

3 PRE-TRAINING

In the pre-training of Qwen2, our efforts were focused on refining the dataset and investigating methods to handle extended context lengths effectively.

Pretraining은 데이터 정제와 Context Length를 늘이는데에 주력함

3.1 PRE-TRAINING DATA

The pre-training of the Qwen2 models involves the development of a new, large-scale, high-quality multilingual dataset. This dataset represents an improvement over the corpora used in previous Qwen and Qwen1.5 models (Bai et al., 2023a; Qwen Team, 2024a), enhancing the scale, quality, and diversity of the pre-training data in several key areas:

Quality Enhancement The filtering algorithm has been refined with additional heuristic and model-based methods, including the use of the Qwen models to filter out low-quality data. Moreover, these models are utilized to synthesize high-quality pre-training data.

Quality Filtering은

- Heuristic

- Model-Based

Model-Based는 기존의 Qwen을 사용
또한 이 모델로 합성데이터 생성에 사용함

Multilingual, 수학, 코드 등
특히 30개국어의 다국어 데이터셋
으로 확장시킴

Data Expansion Compared to Qwen1.5 (Qwen Team, 2024a), we have collected a significantly larger volume of high-quality code, mathematics, and multilingual data, enhancing the model's capabilities in respective areas. This new dataset supports approximately 30 languages, such as English, Chinese, Spanish, French, German, Arabic, Russian, Korean, Japanese, Thai, and Vietnamese.

Distribution Improvement To ensure the model learns the distribution ^{~에 유사한} akin to human-like learning, we conduct experiments on scaled-down models to optimize the mixing of data from various sources and domains.

최적의 데이터 분포를 찾기 위해
작은 Scale의 모델에서 실험

Based on these enhancements, the pre-training data was expanded from 3 trillion tokens in Qwen1.5 (Qwen Team, 2024a) to 7 trillion tokens. An attempt to further relax the quality threshold resulted in a 12 trillion token dataset. However, the model trained on this dataset did not show a significant performance improvement over the 7 trillion token model. It is suspected that increasing the volume of data does not necessarily benefit model pre-training. Considering training costs, we opted to use the higher-quality 7 trillion token dataset for training larger models, leaving further exploration for future model iterations.

Pretraining Data 토큰은
3T -> 7T로 확장시킴

All Qwen2 dense models, excluding Qwen2-0.5B, were pre-trained on this large-scale dataset of over 7 trillion tokens. Qwen2-0.5B were pre-trained using the 12 trillion token dataset. The MoE model received an additional 4.5 trillion tokens of pre-training, in line with the principle of upcycling. Similar to previous Qwen models, high-quality multi-task instruction data is integrated into the Qwen2 pre-training process to enhance in-context learning and instruction-following abilities.

재활용품을 이용하여
기존 제품보다 더 나은
새 제품을 만드는 것

3.2 LONG-CONTEXT TRAINING

To enhance the long-context capability of Qwen2, we augmented the context length from 4,096 tokens to 32,768 tokens during the concluding phase of pre-training. This expansion was complemented by the introduction of a significantly increased volume of high-quality, lengthy data. In conjunction with these enhancements, we modified the base frequency of RoPE from 10,000 to 1,000,000 to optimize performance in long-context scenarios (Xiong et al., 2023).

Pretraining 마지막 단계에서
4096 => 32768로 토큰을 확장함

이와 함께 High quality의
Long Context를 증가시켜
학습에 반영함

To fully leverage the model's length extrapolation potential, we adopted the YARN mechanism (Peng et al., 2023) and the Dual Chunk Attention mechanism (An et al., 2024). These strategies enable the model to process sequences of up to 131,072 tokens while maintaining high performance, as evidenced by minimal perplexity degradation in preliminary experiments.

Long Context 시나리오에 반영하기
위해 RoPE의 Base Frequency를
1e4 -> 1e6로 확장시킴

외산능력 극대화를 위해 YARN과
Dual Chunk Attention을 도입
이를 통해 131072 토큰으로 확장

4 POST-TRAINING

Following ^{중심이 되는} extensive large-scale pre-training, we engage in a post-training phase for Qwen2. This process is ^{수반하다} pivotal in enhancing its proficiency across a broad spectrum of domains, including coding, mathematics, logical reasoning, instruction following, and multilingual comprehension. Moreover, it ensures that the generation from the models is in harmony with human values, making it helpful, honest, and harmless. Unlike traditional methods that heavily rely on extensive human supervision, our approach focuses on scalable alignment with minimal human annotation (Cao et al., 2024). Specifically, we investigate methods to acquire high-quality demonstration and preference data for Supervised Fine-Tuning (SFT) and Reinforcement Learning from Human Feedback (RLHF), aiming to minimize the need for human labeling while maximizing the quality and reliability of the data.

4.1 POST-TRAINING DATA

The post-training data primarily consists of two components: demonstration data $\mathcal{D} = \{(x_i, y_i)\}$ and preference data $\mathcal{P} = \{(x_i, y_i^+, y_i^-)\}$, where x_i represents the instruction, y_i represents a satisfactory response, and y_i^+ and y_i^- are two responses to x_i , with y_i^+ being the preferred choice over y_i^- . The set \mathcal{D} is utilized in SFT, whereas \mathcal{P} is employed in RLHF.

Posttraining 데이터셋은
2단계로 구성함

1) Collaborative Data Annotation
데이터셋에서 Ontology를 추출하여
Instruction을 확장

^{~로 이어지다} The construction of training data entails a two-step process: collaborative data annotation and automated data synthesis. First, we extract the data ontology from large-scale instruction corpora, leading to a broad and diverse set of high-quality instructions. These instructions are systematically enhanced to incorporate greater complexity. Through human annotation, we obtain the target response y_i and their positive and negative counterparts (y_i^+, y_i^-) . Subsequently, a variety of automated

2) Automated data synthesis
Instruction에 Preference를 추가

Ontology
특정 도메인에서 개념들과
개념들간의 관계를 체계적으로
정리한 지식 구조

Qwen2-0.5B는 12T로 학습
그 외는 7T로 학습
MoE는 추가로 4.5T 토큰으로 학습
(버려진 토큰 재활용)

YARN
Yet Another RoPE extension
Context가 길어지면 RoPE의
고주파 정보가 뭉개짐
Interpolation과 Attention
재조정 두 축으로 개선하는 방법

DCA
Dual Chunk Attention
RoPE는 4096 토큰이 한계임
Input을 Chunk (예 4096)으로
나눠서 분리하여 계산함
- Intra-Chunk :
같은 청크 안
- Inter-chunk :
서로 다른 chunk
- Successive-chunk:
바로 이웃 chunk
3가지 Chunk로 나눠서 계산

$$\text{Attn}(Q, K, V) = \text{Intra} + \text{Inter} + \text{Successive}$$

alignment strategies are employed to synthesize a substantial volume of artificially annotated data across the domains of code, mathematics, instruction-following, creation, role-playing, and safety.

4.1.1 COLLABORATIVE DATA ANNOTATION

Automatic Ontology Extraction The process initiates with the application of InsTag (Lu et al., 2024c), an open-set fine-grained tagger, to extract the underlying ontology from a large-scale instruction dataset. Subsequent manual refinement ensures the accuracy of the extracted ontology.

Automatic Ontology Extraction
InsTag라는 어플리케이션의
개방형 세부 태깅 도구를 활용하여
자동으로 온톨로지 추출

Instruction Selection Each instruction, with tags annotated, is evaluated for tag diversity, semantic richness, complexity, and intent completeness. Based on these criteria, we select a set of representative instructions (Dong et al., 2023).

Instruction Selection
태깅의 다양성, 의미가 풍부함,
복잡성, 의도의 완결성 등으로
Instruction 선별

Instruction Evolution To enrich the instruction dataset, a self-evolution strategy (Zhao et al., 2024) is employed, prompting the Qwen models to add constraints or requirements to existing instructions, thereby increasing their complexity and ensuring a diverse range of difficulty levels within the dataset.

Instruction Evolution
더 복잡하고 다양한 Instruction 생성

Human Annotation Multiple responses to an instruction are obtained using diverse generation strategies and Qwen models of different scales. Annotators rank these responses based on their preferences, ensuring the best response meets established criteria, yielding both demonstration and preference data.

Human Annotation
하나의 Instruction에 다양한
generate 전략과 모델 스케일에 따라
여러 응답을 생성함
그리고 Human Annotator에게
평가하게 함

4.1.2 AUTOMATED DATA SYNTHESIS

Maintaining the quality of annotations for responses to instructions presents significant challenges on a large scale, particularly those that require expertise, experience, carefulness, or patience. To address these challenges, we devised various automated alignment strategies to synthesize data at scale.

Rejection Sampling For mathematical or similar tasks with definitive final answers, rejection sampling (Yuan et al., 2023) is applied to improve the quality of solutions. Large language models (LLMs) are tasked to generate multiple responses, **namely** the reasoning paths, for each instruction. Paths that result in accurate conclusions and are considered reasonable by the model are preserved, serving as demonstration data. Preference data is generated by contrasting correct and incorrect paths.

Rejection Sampling
수학 등 명확한 답이 있는 경우,
여러 Response 중
Rejection Sampling을 통해
올바른 추론 경로를 생성하게 함

Execution Feedback For coding tasks, LLMs are employed to generate solutions and associated test cases. The efficacy of these solutions is evaluated by compiling and executing them against the test cases, thereby creating demonstration and preference data. This methodology is also applicable to assessing instruction following (Dong et al., 2024). For each instruction with constraints, e.g., length limit, the LLM is tasked to generate a Python verification function to ensure the response aligns with the instruction requirements.

Execution Feedback
코딩을 평가할 때 실행을 하듯이
Instruction을 실행하고 평가
예를 들면 글자수 100개 제한하면
len(string) < 100을 실행함

~에 맞추어 조정하다

Data Repurposing Creating skilled responses in literary writing tasks is challenging for annotators without specialized training. To tackle this problem, we aggregate high-quality literary works from the public domain and employ LLMs to develop instructions with varying levels of detail. These instructions, paired with the original works, serve as demonstration data. For example, to compile roleplay data with vivid and engaging responses, we source detailed character profiles from knowledge repositories such as Wikipedia and instruct LLMs to generate corresponding instructions and responses (Lu et al., 2024b). This process, similar to a reading comprehension task, ensures that the integrity of the character's profile is maintained.

Data Repurposing
문학 창작 같은 작업을 위함임
고품질 문학 작품을 수집하여
다양한 레벨의 Instruction을
생성하고, 원문과 함께 제공됨
(Few-shot 처럼)

Constitutional Feedback Constitutional AI refers to the process of guiding LLMs to generate responses based on predefined sets of principles (Bai et al., 2022). To ensure adherence to guidelines such as safety and values, a constitution dataset was compiled. This dataset delineates principles to be followed and those to be avoided. It was used to instruct LLMs to produce responses that either are aligned with or deviated from these guidelines, serving as a reference for demonstration and preference data.

Constitutional Feedback
헌법과 같이 기준 데이터 집합을
만들고 이에 따라 응답을 생성하도록
유도하는 방식임
기준 데이터를 만들고 이를 일탈시켜
Preference 데이터로 만들기도 함

4.2 SUPERVISED FINE-TUNING

5e5 Examples

We have assembled an extensive instruction dataset featuring more than 500,000 examples that cover skills such as instruction following, coding, mathematics, logical reasoning, role-playing, multilingualism, and safety. Our model was fine-tuned for two epochs with a sequence length of 32,768 tokens. To optimize learning, the learning rate was gradually decreased from 7×10^{-6} to 7×10^{-7} . To address overfitting, we applied a weight decay of 0.1 and gradients were clipped at a maximum value of 1.0.

4.3 REINFORCEMENT LEARNING FROM HUMAN FEEDBACK

Our training regime for RLHF comprises two sequential stages: offline and online training. In the offline training stage, we use a pre-compiled preference dataset \mathcal{P} to maximize the difference in likelihood between y_i^+ and y_i^- with Direct Preference Optimization (DPO, Rafailov et al., 2023). In the online training stage, the model iteratively refines its performance in real-time, leveraging reward models for immediate feedback. Specifically, we sample multiple responses from the current policy model, and the reward model selects the most and the least preferred responses, forming preference pairs that are used for DPO in each episode. Moreover, we employ Online Merging Optimizer (Lu et al., 2024a) to mitigate the alignment tax, i.e., the performance degradation associated with aligning model generation with human preferences.

Alignment Tax를 줄이기 위해
Online Merging Optimizer를 사용

5 EVALUATION

To thoroughly assess the Qwen2 models, consisting of both base and instruction-tuned models, we implement a comprehensive evaluation protocol. This protocol examines a range of competencies, including general knowledge understanding, language comprehension, generation, coding, mathematics, reasoning, and additional areas of expertise. Specifically, base models are assessed using established benchmark datasets for large language models (LLMs), with responses elicited through few-shot prompting, unless specified otherwise. For instruction-tuned models, in addition to benchmark evaluations, we prioritize human preference assessments.

5.1 BASE LANGUAGE MODELS

In this section, we illustrate the evaluation of the base language models of the Qwen2 series. Specifically, we evaluate the models on benchmark datasets for knowledge and basic capabilities and apply multilingual benchmark datasets to evaluate their support of languages. As there are multiple model sizes, we compare them with the state-of-the-art (SOTA) models of similar or larger sizes.

5.1.1 CORE CAPABILITIES

Benchmarks and Evaluation Protocol The common practice of evaluating the core capabilities of base language models is the implementation of benchmark dataset evaluation with few-shot or zero-shot prompting. The evaluation mainly focuses on the model performance of natural language understanding, general question answering, coding, mathematics, scientific knowledge, reasoning, etc. The datasets for evaluation include MMLU (Hendrycks et al., 2021a) (5-shot), MMLU-Pro (Wang et al., 2024) (5-shot), GPQA (Rein et al., 2023) (5shot), Theorem QA (Chen et al., 2023a) (5-shot), BBH (Suzgun et al., 2023) (3-shot), HellaSwag (Zellers et al., 2019) (10-shot), Winogrande (Sakaguchi et al., 2021) (5-shot), TruthfulQA (Lin et al., 2022a) (0-shot), ARC-C (Clark et al., 2018) (25-shot), HumanEval (Chen et al., 2021) (0-shot), MBPP (Austin et al., 2021) (0-shot), EvalPlus (Liu et al., 2023a) (0-shot), MultiPL-E (Cassano et al., 2023) (0-shot on Python, C++, Java, PHP, TypeScript, C#, Bash, and JavaScript), GSM8K (Cobbe et al., 2021) (5-shot), MATH (Hendrycks et al., 2021b) (4-shot), C-Eval (Huang et al., 2023) (5-shot), and CMMLU (Li et al., 2023) (5-shot). Multilingual datasets can be grouped into four categories: (a) Exam: M3Exam (5-shot, we only choose examples that require no image), IndoMMLU (Koto et al., 2023) (3-shot), ruMMLU (Fenogenova et al., 2024) (5-shot), and translated MMLU (Chen et al., 2023b) (5-shot on Arabic, Spanish, French, Portuguese, German, Italian, Japanese, and Korean); (b) Understanding: BELEBELE (Bandarkar et al., 2023) (5-shot), XCOPA (Ponti et al., 2020) (5-shot), XWinograd (Muennighoff et al., 2023) (5-shot), XStoryCloze (Lin et al., 2022b) (0-shot) and PAWS-X (Yang et al., 2019) (5-shot); (c)

Table 2: **Performance of the 70B+ models.** We compare Qwen2-72B with the baselines, including Mixtral-8x22B, Llama-3-70B, Qwen1.5-110B, and Qwen1.5-72B. For most datasets, Qwen2-72B demonstrates advantages over the baselines.

Datasets	Mixtral-8x22B	Llama-3-70B	Qwen1.5-72B	Qwen1.5-110B	Qwen2-72B
<i>English</i>					
MMLU	77.8	79.5	77.5	80.4	84.2
MMLU-Pro	49.5	52.8	45.8	49.4	55.6
GPQA	34.3	36.3	36.3	35.9	37.9
Theorem QA	35.9	32.3	29.3	34.9	43.1
BBH	78.9	81.0	65.5	74.8	82.4
HellaSwag	88.7	88.0	86.0	87.5	87.6
Winogrande	85.0	85.3	83.0	83.5	85.1
ARC-C	70.7	68.8	65.9	69.6	68.9
TruthfulQA	51.0	45.6	59.6	49.6	54.8
<i>Coding</i>					
HumanEval	46.3	48.2	46.3	54.3	64.6
MBPP	71.7	70.4	66.9	70.9	76.9
EvalPlus	54.1	54.8	52.9	57.7	65.4
MultiPL-E	46.7	46.3	41.8	52.7	59.6
<i>Mathematics</i>					
GSM8K	83.7	83.0	79.5	85.4	89.5
MATH	41.7	42.5	34.1	49.6	51.1
<i>Chinese</i>					
C-Eval	54.6	65.2	84.1	89.1	91.0
CMMLU	53.4	67.2	83.5	88.3	90.1
<i>Multilingual</i>					
Exam	63.5	70.0	66.4	75.6	76.6
Understanding	77.7	79.9	78.2	78.2	80.7
Mathematics	62.9	67.1	61.7	64.4	76.0
Translation	23.3	38.0	35.6	36.2	37.8

Mathematics: MGSM (Goyal et al., 2022) (8-shot CoT); and (d) Translation: Flores-101 (Goyal et al., 2022) (5-shot).

Qwen2-72B In terms of the largest model of Qwen2, we compare Qwen2-72B with competitive baseline open-weight models, including Mixtral-8x22B (Jiang et al., 2024), Llama-3-70B (AI@Meta, 2024), as well as Qwen1.5-72B (Qwen Team, 2024a) and Qwen1.5-110B (Qwen Team, 2024b). The results are reported in Table 2. Qwen2-72B outperforms Llama-3-70B in general **knowledge** understanding on both MMLU and MMLU-Pro, achieving accuracy improvements of 4.7 and 2.8, respectively. In **scientific** assessments, Qwen2-72B demonstrates superiority over Llama-3-70B with enhancements of 1.6 and 9.8 on GPQA and Theorem QA. Upon enrichment of **coding** data, Qwen2-72B exhibits a significant 18.3 and 10.0 percentage point advantage over Qwen1.5-72B in HumanEval and MBPP evaluations. Enhanced **mathematics**-related data allows Qwen2-72B to outperform Qwen1.5-72B by 10.0 and 17.0 percentage points in the GSM8K and MATH benchmarks. Qwen2-72B displays **reasoning** capabilities equivalent to Llama-3-70B, considering BBH, Winogrande, and ARC-C, attributable to its improved coding and mathematical data. In assessing language understanding in **Chinese**, Qwen2-72B significantly outperforms Mixtral-8x22B and Llama-3-70B, and also outperforms Qwen1.5-72B.

Qwen2-57B-A14B For the evaluation of the MoE model, Qwen2-57B-A14B is compared against baselines of similar sizes. These baselines include other MoE models, such as Mixtral-8x7B (Jiang et al., 2024) and Jamba (Lieber et al., 2024), and dense models, such as Yi-1.5-34B (Young et al., 2024)

Table 3: **Performance of the 30B+ dense models and 40B+ MoE models.** Qwen2-57B-A14B, an MoE model with a total of 57 billion parameters and 14 billion activated parameters, is designed to match the performance of 30 billion parameter dense models. This comparison includes dense model baselines: Yi-1.5-34B and Qwen1.5-32B, as well as MoE baselines: Mixtral-8x7B and Jamba. Results demonstrate that Qwen2-57B-A14B achieves competitive performance overall, with a notable superiority in coding and mathematics tasks.

Datasets	Jamba	Mixtral-8x7B	Yi-1.5-34B	Qwen1.5-32B	Qwen2-57B-A14B
Architecture	MoE	MoE	Dense	Dense	MoE
# Act Params	12B	12B	32B	34B	14B
# Params	52B	47B	32B	34B	57B
<i>English</i>					
MMLU	67.4	71.8	77.1	74.3	76.5
MMLU-Pro	-	41.0	48.3	44.0	43.0
GPQA	-	29.2	-	30.8	34.3
Theorem QA	-	23.2	-	28.8	33.5
BBH	45.4	50.3	76.4	66.8	67.0
HellaSwag	87.1	86.5	85.9	85.0	85.2
Winogrande	82.5	81.9	84.9	81.5	79.5
ARC-C	64.4	66.0	65.6	63.6	64.1
TruthfulQA	46.4	51.1	53.9	57.4	57.7
<i>Coding</i>					
HumanEval	29.3	37.2	46.3	43.3	53.0
MBPP	-	63.9	65.5	64.2	71.9
EvalPlus	-	46.4	51.9	50.4	57.2
MultiPL-E	-	39.0	39.5	38.5	49.8
<i>Mathematics</i>					
GSM8K	59.9	62.5	82.7	76.8	80.7
MATH	-	30.8	41.7	36.1	43.0
<i>Chinese</i>					
C-Eval	-	-	-	83.5	87.7
CMMLU	-	-	84.8	82.3	88.5
<i>Multilingual</i>					
Exam	-	56.1	58.3	61.6	65.5
Understanding	-	70.7	73.9	76.5	77.0
Mathematics	-	45.0	49.3	56.1	62.3
Translation	-	29.8	30.0	33.5	34.5

and Qwen1.5-32B (Qwen Team, 2024a), both of which have approximately 30 billion parameters. The results are shown in Table 3. We anticipate that Qwen2-57B-A14B, which activates 14 billion parameters, will match the performance of a 30 billion parameter dense equivalent Qwen2 model. Our evaluation reveals that Qwen2-57B-A14B performs comparably to Yi-1.5-34B in natural language understanding tasks. Moreover, it outperforms the baseline models in coding and mathematics tasks. Additionally, Qwen2-57B-A14B demonstrates robust Chinese language understanding capabilities, rivaling the larger Qwen2-72B model. In essence, Qwen2-57B-A14B is an efficient model that, while activating only 14 billion parameters per forward pass, maintains the performance level of a 30 billion parameter dense model.

Qwen2-7B The 7B model is widely utilized, as it enables the execution in 16-bit floating points on accelerators equipped with 16GB memory. Our focus is on comparing this model with other leading 7B models, including Llama-3-8B, which has recently demonstrated exceptional performance in the Chatbot Arena (Chiang et al., 2024). This comparison also includes Mistral-7B-v0.2 (Jiang et al., 2023a), Gemma-7B (Mesnard et al., 2024), and our predecessor, Qwen1.5-7B (Qwen Team, 2024a).

Table 4: **Performance of the 7B+ models.** We compare Qwen2-7B with previously released state-of-the-art 7B+ models including Mixtral-7B, Gemma-7B, Llama-3-8B, and our previous Qwen1.5-7B. Qwen2-7B demonstrates significant advantages over the baselines in most of the evaluation datasets.

Datasets	Mistral-7B	Gemma-7B	Llama-3-8B	Qwen1.5-7B	Qwen2-7B
<i>English</i>					
MMLU	64.2	64.6	66.6	61.0	70.3
MMLU-Pro	30.9	33.7	35.4	29.9	40.0
GPQA	24.7	25.7	25.8	26.7	31.8
Theorem QA	19.2	21.5	22.1	14.2	31.1
BBH	56.1	55.1	57.7	40.2	62.6
HellaSwag	83.2	82.2	82.1	78.5	80.7
Winogrande	78.4	79.0	77.4	71.3	77.0
ARC-C	60.0	61.1	59.3	54.2	60.6
TruthfulQA	42.2	44.8	44.0	51.1	54.2
<i>Coding</i>					
HumanEval	29.3	37.2	33.5	36.0	51.2
MBPP	51.1	50.6	53.9	51.6	65.9
Evalplus	36.4	39.6	40.3	40.0	54.2
MultiPL-E	29.4	29.7	22.6	28.1	46.3
<i>Mathematics</i>					
GSM8K	52.2	46.4	56.0	62.5	79.9
MATH	13.1	24.3	20.5	20.3	44.2
<i>Chinese</i>					
C-Eval	47.4	43.6	49.5	74.1	83.2
CMMLU	-	-	50.8	73.1	83.9
<i>Multilingual</i>					
Exam	47.1	42.7	52.3	47.7	59.2
Understanding	63.3	58.3	68.6	67.6	72.0
Mathematics	26.3	39.1	36.3	37.3	57.5
Translation	23.3	31.2	31.9	28.4	31.5

The results can be found in Table 4. Qwen2-7B demonstrates superior performance across most datasets compared to other models, particularly excelling in coding tasks, mathematics, and Chinese language tasks. It also shows strong performance in multilingual understanding and exams. This indicates that Qwen2-7B has been optimized for a wide range of language and logic-based tasks, showcasing its versatility and advanced capabilities.

Qwen2-1.5B & Qwen2-0.5B To evaluate the performance of our smaller models, specifically Qwen2-1.5B and Qwen2-0.5B, we compare them against established baselines: Phi-2 (Abdin et al., 2024), Gemma-2B (Mesnard et al., 2024), and Qwen1.5-1.8B (Qwen Team, 2024a). The results are given in Table 5. In language understanding, Qwen2-1.5B outperforms Phi-2, a model trained on textbook-like data. For coding tasks, Qwen2-0.5B matches the performance of Gemma-2B and Qwen1.5-1.8B, while Qwen2-1.5B surpasses these baselines, except for Phi-2. Both Qwen2 models exhibit superior performance in mathematics compared to their competitors. In terms of general reasoning, we find that Phi-2 generally outperforms all others, which to some extent reflects the significance of textbook data for reasoning capabilities. In TruthfulQA, Qwen2-1.5B performs the best, demonstrating that smaller models does not necessarily suffer from hallucination. In Chinese language understanding, both Qwen2 models outperform all the others, a trend consistent with larger models in their respective comparisons.

In general, the Qwen2 series demonstrates superior performance against the baselines across different model sizes. Notably, Qwen2-72B exhibits the highest performance among all Qwen2 models, underscoring the efficacy of model size scaling.

Table 5: **Performance of the smaller models.** We compare our Qwen2-0.5B and Qwen2-1.5B with the previous SOTA small models including Phi-2, Gemma-2B and Qwen1.5-1.8B. Qwen2-0.5B with a much smaller model size achieves competitive performance, and Qwen2-1.5B significantly outperforms Qwen2-0.5B.

Datasets	Phi-2	Gemma-2B	Qwen1.5-1.8B	Qwen2-0.5B	Qwen2-1.5B
# Non-Emb Params	2.5B	2.0B	1.2B	0.3B	1.2B
MMLU	52.7	42.3	46.8	45.4	56.5
MMLU-Pro	-	15.9	-	14.7	21.8
Theorem QA	-	-	-	8.9	15.0
BBH	43.4	35.2	24.2	28.4	37.2
HellaSwag	73.1	71.4	61.4	49.3	66.6
Winogrande	74.4	66.8	60.3	56.8	66.2
ARC-C	61.1	48.5	37.9	31.5	43.9
TruthfulQA	44.5	33.1	39.4	39.7	45.9
HumanEval	47.6	22.0	20.1	22.0	31.1
MBPP	55.0	29.2	18.0	22.0	37.4
GSM8K	57.2	17.7	38.4	36.5	58.5
MATH	3.5	11.8	10.1	10.7	21.7
C-Eval	23.4	28.0	59.7	58.2	70.6
CMMLU	24.2	-	57.8	55.1	70.3

5.2 INSTRUCTION-TUNED MODEL

To critically evaluate instruction-tuned models, we implement a multifaceted approach. Assessments of foundational skills and human preferences are conducted using open datasets and benchmarks. Our detailed in-house examinations further probe model competencies in key areas. A particular focus is placed on assessing long context capability. Safety measures include multilingual safety assessments and red teaming exercises. The following sections detail the evaluation methods and their outcomes.

5.2.1 OPEN BENCHMARK EVALUATION

To comprehensively evaluate the quality of instruction-tuned models, we compile automatic and human evaluation to assess the capabilities and human preference. For the evaluation of basic capabilities, we apply similar datasets in the pre-trained model evaluation, which target on natural language understanding, coding, mathematics, and reasoning. Specifically, we evaluate on MMLU, MMLU-Pro, GPQA, and Theorem QA for language understanding and knowledge, HumanEval, MBPP, MultiPL-E, and LiveCodeBench v1 (Jain et al., 2024) for coding, GSM8K and MATH for mathematics. Additionally, we assess the performance of human preference alignment and instruction following by evaluating on benchmarks including MT-Bench (Zheng et al., 2023), Arena-Hard (Li et al., 2024), AlignBench (Liu et al., 2023b), MixEval (Ni et al., 2024) whose results approximate those of Chatbot Arena, and IFEval (Zhou et al., 2023)⁴ for instruction following.

Qwen2-72B-Instruct We compare Qwen2-72B-Instruct against the instruction-tuned models including Mixtral-8x22B-Instruct, Llama-3-70B-Instruct, as well as Qwen1.5-72B-Chat. The results are presented in Table 6. It can be found that a strong base language model can help boost the downstream performance of the instruction-tuned model. Specifically, Qwen2-72B-Instruct outshines its peers in areas such as language understanding, coding, and mathematics, with the exception of GPQA and MBPP. Regarding human preference alignment and instruction following, Qwen2-72B has significant advantages over the baselines. We assume this achievement is attributed to both the high-quality pre-trained model and improvements in both data and training techniques for post-training.

⁴For simplicity, we report the results of the subset *strict-prompt*.

Table 6: **Performance of 70B+ instruction-tuned models.** We compare Qwen2-72B-Instruct with Mixtral-8x22B-Instruct, Llama-3-70B-Instruct, Qwen1.5-72B-Chat, and Qwen1.5-110B-Chat. “-Instruct” or “-Chat” is omitted in the table. Qwen2-72B-Instruct demonstrates advantages in core capabilities, and superior performance in human preference alignment.

Datasets	Mixtral-8x22B	Llama-3-70B	Qwen1.5-72B	Qwen1.5-110B	Qwen2-72B
<i>English</i>					
MLU	74.0	82.0	75.6	76.5	82.3
MLU-Pro	56.1	56.2	51.7	50.5	64.4
GPQA	49.7	41.9	39.4	32.8	42.4
Theorem QA	40.8	42.5	28.8	18.8	44.4
<i>Coding</i>					
HumanEval	73.8	81.7	71.3	74.4	86.0
MBPP	75.9	82.3	71.9	76.4	80.2
MultiPL-E	61.1	63.4	48.1	55.4	69.2
LiveCodeBench <i>v1</i>	21.8	29.3	17.9	25.3	35.7
<i>Mathematics</i>					
GSM8K	89.1	93.0	82.7	84.5	93.2
MATH	47.4	50.4	42.5	42.0	69.0
<i>Alignment</i>					
MT-Bench	8.66	8.95	8.61	8.88	9.12
MixEval	82.3	84.0	84.1	85.7	86.7
Arena-Hard	36.4	41.1	36.1	39.8	48.1
IFEval <i>strict-prompt</i>	67.1	77.3	55.8	57.5	77.6
AlignBench	-	7.42	7.28	7.87	8.27

Qwen2-57B-A14B-Instruct For medium-size models, we compare Qwen2-57B-A14B-Instruct with Mixtral-8x7B-Instruct, another MoE baseline, as well as the dense SOTA models with over 30 billion parameters, e.g., Yi-1.5-34B-Chat and Qwen1.5-32B-Chat. The results are provided in Table 7. Compared with Qwen1.5-32B-Chat, Qwen2-57B-A14B-Instruct reaches superior performance in almost all benchmarks, and compared with the 30B SOTA model Yi-1.5-34B-Chat, Qwen2-57B-A14B-Instruct has gained advantages in most evaluations except for those for mathematics. In terms of the evaluation for alignment, the advantages of Qwen2-57B-A14B-Instruct are notably evident.

Qwen2-7B-Instruct Within the spectrum of 7B to 9B models, we compare Qwen2-7B-Instruct with Llama-3-8B-Instruct, Yi-1.5-9B-Chat, GLM-4-9B-Chat, and Qwen1.5-7B-Chat. The results can be found in Table 8. Qwen2-7B-Instruct demonstrates substantial advancements compared to its predecessor, Qwen1.5-7B-Chat, across comprehensive evaluations, notably achieving higher scores in coding and mathematics-related tasks. Compared with the recent SOTA model, Llama-3-8B-Instruct, Qwen2-7B-Instruct demonstrates competitive performance and specifically it achieves superior performance in coding. Nonetheless, in terms of instruction following, Qwen2-7B-Instruct greatly falls behind the competitor. To address this limitation, we plan to augment the 7B model’s instruction-following ability by enhancing the quality of post-training data, ensuring a more robust understanding and execution of complex commands.

Qwen2-1.5B-Instruct & Qwen2-0.5B-Instruct In the context of smaller models, we compare Qwen2-0.5B-Instruct with Qwen1.5-0.5B-Chat, and Qwen2-1.5B-Instruct with Qwen1.5-1.8B-Chat. Notably, the complexity of certain datasets designed for larger models exceeds the capabilities of these smaller models; thus, our analysis focuses on a selected subset. As detailed in Table 9, the Qwen2 models demonstrate a marked advantage over their predecessors in both core capabilities and instruction-following tasks. The achievement mainly attributes to the scaling of pre-training data. Consequently, our results affirm that data scaling remains an effective strategy for enhancing model performance, even in the domain of sub-billion parameter models.

Table 7: **Performance of 30B+ dense and 40B+ MoE instruction-tuned models.** We compare Qwen2-57B-A14B-Instruct with the similar-size MoE model Mixtral-8x7B-Instruct, 30B dense models such as Yi-1.5-34B-Chat and Qwen1.5-32B-Chat. “-Instruct” or “-Chat” is omitted in the table. Qwen2-57B-A14B-Instruct is competitive with the recent SOTA 30B dense models, and significantly outcompetes the MoE baseline.

Datasets	Mixtral-8x7B	Yi-1.5-34B	Qwen1.5-32B	Qwen2-57B-A14B
Architecture	MoE	Dense	Dense	MoE
# Act Params	12B	32B	34B	14B
# Params	47B	32B	34B	57B
<i>English</i>				
MMLU	71.4	76.8	74.8	75.4
MMLU-Pro	43.3	52.3	46.4	52.8
GPQA	-	-	30.8	34.3
Theorem QA	-	-	30.9	33.1
<i>Coding</i>				
HumanEval	45.1	75.2	68.3	79.9
MBPP	59.5	74.6	67.9	70.9
MultiPL-E	-	-	50.7	66.4
LiveCodeBench v1	12.3	-	15.2	25.5
<i>Mathematics</i>				
GSM8K	65.7	90.2	83.6	85.3
MATH	30.7	50.1	42.4	49.1
<i>Alignment</i>				
MT-Bench	8.30	8.50	8.30	8.55
MixEval	70.0	81.7	81.0	82.3
IFEval <i>strict-prompt</i>	-	-	50.3	59.9
AlignBench	5.70	7.20	7.19	7.36

5.2.2 IN-HOUSE AUTOMATIC EVALUATION

Despite a number of open benchmark datasets for the evaluation, we believe that it is far from sufficient to fully comprehend the capabilities of LLMs. Specifically, we have made a series of in-house datasets that assess different capabilities of the models, e.g., knowledge understanding, text generation, coding, etc. The evaluation is in Chinese and English. The results are gathered in Table 10 and Table 11, respectively.

Chinese Evaluation For the evaluations in Chinese, we focus on comparing the performance of Qwen2 models with the Qwen1.5 counterparts. For the small models, Qwen2-1.5B-Instruct generally outperforms Qwen1.5-1.8B-Chat in almost all the evaluations even with fewer parameters. In terms of the comparison of 7B models, the advantages of Qwen2 are more significant. Noteworthy is Qwen2-72B’s superior performance to Qwen1.5-110B-Chat, despite the latter’s greatly more parameters. The MoE model displays superior performance across most domains relative to Qwen1.5-32B-Chat, excluding knowledge understanding. This discrepancy may be attributed to a short of pre-training tokens. In the near future, we are about to continue the pre-training of the MoE model to discover its scaling behaviors.

English Evaluation For English, we compare Qwen2 with both Qwen1.5 and Llama-3. Similarly, the small models of Qwen2 significantly outcompete the Qwen1.5 counterparts. However, in comparison with Llama-3-70B, Qwen2-72B-Instruct is falling behind by small margins especially in comprehension and coding. We assume both the amount of English tokens for pre-training and the quantity and diversity of data for post-training lead to the performance gap in English.

Table 8: **Performance of 7B+ instruction-tuned models.** We compare Qwen2-7B-Instruct with the recent SOTA models with 7-9 billion parameters, including Llama-3-8B-Instruct, Yi-1.5-9B-Chat, GLM-4-9B-Chat, and Qwen1.5-7B-Chat. “-Instruct” or “-Chat” is omitted in the table. Qwen2-7B-Instruct demonstrates competitive performance against Llama-3-8B-Instruct.

Datasets	Llama-3-8B	Yi-1.5-9B	GLM-4-9B	Qwen1.5-7B	Qwen2-7B
<i>English</i>					
MMLU	68.4	69.5	72.4	59.5	70.5
MMLU-Pro	41.0	-	-	29.1	44.1
GPQA	34.2	-	-	27.8	34.3
Theorem QA	23.0	-	-	14.1	25.3
<i>Coding</i>					
HumanEval	62.2	66.5	71.8	46.3	79.9
MBPP	67.9	-	-	48.9	67.2
MultiPL-E	48.5	-	-	27.2	59.1
LiveCodeBench <i>v1</i>	17.3	-	-	6.0	26.6
<i>Mathematics</i>					
GSM8K	79.6	84.8	79.6	60.3	85.7
MATH	30.0	47.7	50.6	23.2	52.9
<i>Alignment</i>					
MT-Bench	8.05	8.20	8.35	7.60	8.41
MixEval	75.0	74.2	-	71.4	76.5
IFEval <i>strict-prompt</i>	72.1	-	69.0	38.3	54.7
AlignBench	6.20	6.90	7.01	6.20	7.21

Table 9: **Performance of smaller instruction-tuned models.** We compare both Qwen2-0.5B-Instruct and Qwen2-1.5B-Instruct with Qwen1.5-0.5B-Chat and Qwen2-1.8B-Chat. “-Instruct” or “-Chat” is omitted in the table. Compared with the similar-size baselines, Qwen2 significant surpasses the performance of Qwen1.5.

Datasets	Qwen1.5-0.5B	Qwen2-0.5B	Qwen1.5-1.8B	Qwen2-1.5B
MMLU	35.0	37.9	43.7	52.4
HumanEval	10.4	29.9	27.4	47.0
MBPP	14.5	37.8	28.6	51.9
GSM8K	11.3	40.1	35.3	61.6
IFEval <i>strict-prompt</i>	14.6	20.0	16.8	29.0

5.2.3 LONG CONTEXT CAPABILITIES

Three methods to evaluate long context capabilities are employed: the Needle in a Haystack (NIAH, Kamradt, 2023), NeedleBench (OpenCompass Contributors, 2023), and LV-Eval (Yuan et al., 2024).

Needle in a Haystack This experiment assesses a model’s proficiency in pinpointing facts within voluminous texts. Texts with 8K, 16K, ..., 128K tokens in length were crafted, with facts strategically positioned at varying depths. Each depth interval, e.g., from 0% to 10%, encompassed two instances. For contexts over 32K, YARN (Peng et al., 2023) was applied in this evaluation. As illustrated in Figure 1, Qwen2-72B-Instruct exhibits exceptional accuracy in retrieving information from the entire 128K context. Coupled with its inherent strength, this model emerges as the optimal choice for processing extensive texts, assuming sufficient resources are accessible. Additionally, models within the same series showcases remarkable performance across different context lengths. Precisely, Qwen2-7B-Instruct achieves a high level of accuracy in handling contexts up to 128K tokens. Meanwhile, Qwen2-57B-A14B-Instruct manages contexts up to 64K tokens proficiently, and the two smaller models in the Qwen2 series could support contexts of 32K tokens.

Table 10: **Performances of Qwen2-Instruct models on our in-house Chinese automatic evaluation benchmark.** Scores of Qwen2 models surpassing their comparable-sized Qwen1.5 counterparts are in bold. Qwen2-57B-A14B-Instruct is compared with Qwen1.5-32B-Chat.

Models	Knowledge	Exam	Comprehension	Coding	Math	Reasoning	Avg.
<i>Proprietary LLMs</i>							
GPT-4o-2024-05-13	66.68	69.04	76.85	59.58	71.16	69.94	68.87
Qwen-Max-0428	76.65	74.80	73.66	49.48	66.01	70.84	68.57
<i>Qwen1.5 Series</i>							
Qwen1.5-0.5B-Chat	28.55	36.99	29.70	3.82	13.10	25.47	22.94
Qwen1.5-1.8B-Chat	30.31	44.98	44.81	6.86	29.85	34.61	31.90
Qwen1.5-4B-Chat	33.67	47.17	50.44	14.05	36.20	39.98	36.92
Qwen1.5-MoE-A2.7B-Chat	52.76	60.49	52.84	19.34	38.45	43.07	44.49
Qwen1.5-7B-Chat	56.77	59.36	55.50	18.85	46.41	48.77	47.61
Qwen1.5-14B-Chat	63.35	66.13	60.06	28.19	54.80	50.20	53.79
Qwen1.5-32B-Chat	68.63	67.59	64.67	35.28	60.62	62.87	59.94
Qwen1.5-72B-Chat	71.52	70.04	66.70	38.22	63.09	61.30	61.81
Qwen1.5-110B-Chat	76.26	74.00	71.25	44.25	64.92	64.47	65.86
<i>Qwen2 Series</i>							
Qwen2-0.5B-Instruct	28.18	38.09	35.90	9.40	21.20	25.61	26.40
Qwen2-1.5B-Instruct	35.46	51.93	44.70	14.05	34.58	35.94	36.11
Qwen2-7B-Instruct	61.54	66.66	59.63	34.74	60.99	58.22	56.96
Qwen2-57B-A14B-Instruct	64.15	73.67	67.52	40.66	63.90	59.89	61.63
Qwen2-72B-Instruct	76.19	75.65	74.72	49.53	70.80	70.59	69.58

Table 11: **Performances of Qwen2-Instruct models on our in-house English automatic evaluation benchmark.** Scores of Qwen2 models surpassing their comparable-sized Qwen1.5 and Llama-3 counterparts are in bold. Qwen2-57B-A14B-Instruct is compared with Qwen1.5-32B-Chat.

Models	Knowledge	Comprehension	Coding	Math	Avg.
<i>Proprietary LLMs</i>					
GPT-4o-2024-05-13	87.29	76.30	55.87	84.99	76.11
Qwen-Max-0428	80.73	71.63	48.76	79.12	70.06
<i>Qwen1.5 Series</i>					
Qwen1.5-0.5B-Chat	30.12	25.44	1.78	15.48	18.21
Qwen1.5-1.8B-Chat	40.37	41.87	4.99	29.71	29.23
Qwen1.5-4B-Chat	51.44	50.16	15.45	44.83	40.47
Qwen1.5-MoE-A2.7B-Chat	61.64	54.79	21.28	50.46	47.04
Qwen1.5-7B-Chat	64.86	58.61	20.79	54.24	49.62
Qwen1.5-14B-Chat	74.41	59.80	28.18	66.91	57.32
Qwen1.5-32B-Chat	76.38	64.70	37.39	73.04	62.88
Qwen1.5-72B-Chat	77.59	67.58	37.30	73.76	64.06
Qwen1.5-110B-Chat	78.29	70.17	44.12	78.87	67.86
<i>Llama-3 Series</i>					
Llama-3-8B-Instruct	71.01	64.71	42.56	65.82	61.03
Llama-3-70B-Instruct	83.06	76.31	57.18	79.70	74.06
<i>Qwen2 Series</i>					
Qwen2-0.5B-Instruct	43.19	29.57	6.95	31.52	27.81
Qwen2-1.5B-Instruct	56.03	45.08	17.61	50.44	42.29
Qwen2-7B-Instruct	73.75	63.09	36.41	75.67	62.23
Qwen2-57B-A14B-Instruct	76.80	67.92	42.37	77.04	66.03
Qwen2-72B-Instruct	83.00	73.58	53.03	82.15	72.94

Testing Qwen2-Instruct via “Needle in A HayStack”

Retrieve Facts from Given Documents across Context Lengths and Document Depth

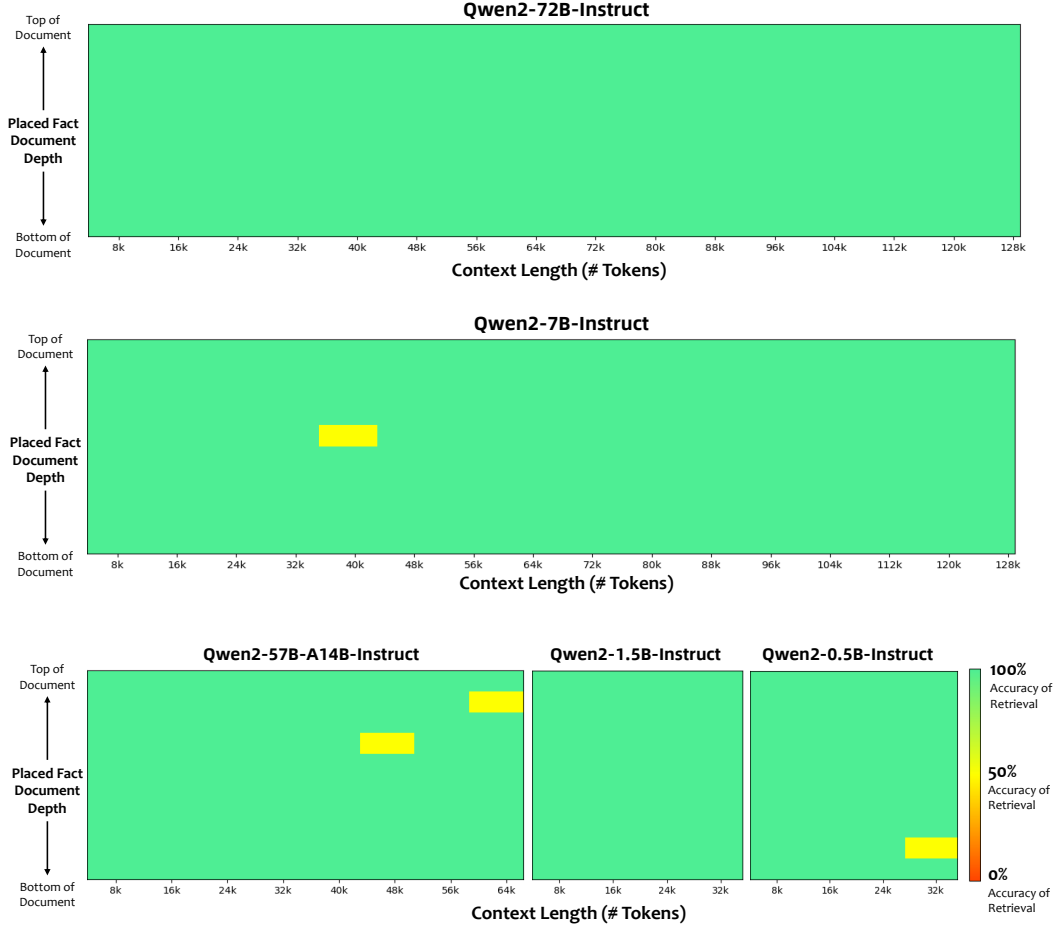


Figure 1: **Performance of Qwen2 instruction-tuned models on Needle in A Haystack Test.** All models that supports context lengths above 32k tokens integrates the YARN mechanism.

Table 12: **Performance of Qwen2-72B-Instruct and Qwen2-7B-Instruct on NeedleBench and LV-Eval.** +YARN+DCA does not change the model behavior within 32k tokens.

Datasets	NeedleBench				LV-Eval				
	8k	32k	128k	256k	16k	32k	64k	128k	256k
ChatGLM4-9B-1M	56.61	49.15	44.30	45.29	46.40	43.23	42.92	40.41	36.95
Qwen2-7B-Instruct + YARN + DCA	87.07	73.64	38.77 66.32	2.92 60.71	49.77	46.93	28.03 42.14	11.01 36.64	0.55 34.72
Qwen2-72B-Instruct + YARN + DCA	91.90	92.01	73.05 90.27	17.13 85.21	58.82	56.70	42.92 53.03	31.79 48.83	2.88 42.35

Table 13: **Performance of Qwen2-72B-Instruct and proprietary LLMs in multilingual human evaluation.** We compare Qwen2-72B-Instruct with GPT-3.5-Turbo-1106, GPT-4-Turbo-0409, GPT-4o-0513, Claude-3-Opus-0229. Scores range from 1 to 5. Overall, Qwen2-72B-Instruct performs substantially better than GPT-3.5-Turbo but there is progress to be made to be competitive with the proprietary models released in the last 6 months.

Language	GPT-3.5-Turbo	GPT-4-Turbo	GPT-4o	Claude-3-Opus	Qwen2-72B-Instruct
Arabic	2.52	3.44	3.55	4.15	3.86
French	3.47	4.19	4.16	4.23	4.01
Indonesian	3.56	4.09	4.39	4.40	3.83
Japanese	2.75	3.68	3.72	3.85	3.63
Korean	2.37	4.24	4.40	4.23	4.14
Portuguese	3.37	3.86	3.89	4.09	3.97
Russian	3.24	4.27	4.32	4.25	4.15
Spanish	4.07	4.08	4.26	4.31	4.10
Thai	3.38	4.11	4.09	4.01	3.75
Vietnamese	3.90	3.84	4.14	3.98	3.91
Average	3.16	3.98	4.09	4.15	3.93

NeedleBench NeedleBench ups the challenge on NIAH by including multiple facts (two to five) in passages, necessitating simultaneous identification and multi-hop reasoning. Table 12 reveals that the integration of YARN and DCA (An et al., 2024) notably improves Qwen2 models’ long-context abilities. Qwen2-7B-Instruct surpasses ChatGLM4-9B-1M (Zeng et al., 2024), which claims a 1M context length. Moreover, Qwen2-72B-Instruct demonstrates strong performance, with an accuracy reduction of just 6 points, compared to ChatGLM4-9B-1M, which shows a more pronounced decline of 11 points, particularly given its lower initial accuracy.

LV-Eval LV-Eval comprises 11 diverse QA datasets that demand comprehension of multiple pieces of evidence at once. To rectify the shortcomings of its original metric, which was excessively stringent and led to a high rate of false negatives, we adopt the keyword recall as the reported score. As shown in Table 12, integrating YARN and DCA substantially bolsters the long-context competencies of Qwen2 models on LV-Eval. Qwen2-7B-Instruct achieves parity with ChatGLM4-9B-1M, albeit with a more noticeable decline at extended contexts. Moreover, Qwen2-72B-Instruct demonstrates strong performance across all lengths, confirming its proficiency in handling long-context tasks.

5.2.4 MULTILINGUAL EVALUATION

For the multilingual evaluation, we implement a comprehensive human evaluation for the assessment of multilingual capabilities. Specifically, we design diverse test cases assessing different capabilities of large language models, and we have test cases that are in a number of languages. For the annotators, we invite one professional annotator for each language who majors in the language for the evaluation. For each test case, the annotator grades the response from model with a score from 1 to 5.

We report the results of our model and the baselines in the evaluation of different languages. From Table 13, it can be found that on average Qwen2-72B-Instruct significantly outperforms GPT-3.5-Turbo and it is competitive with GPT-4-Turbo and slightly falls behind Claude-3-Opus. This shows that our multilingual pre-training and instruction tuning data contribute to the multilingual capabilities of Qwen2-72B-Instruct and it is competitive with most state-of-the-art proprietary LLMs.

5.2.5 SAFETY & RESPONSIBILITY

LLMs with openly accessible weights effectively accelerate the development of the research as well as their applications. Moreover, we believe that it is crucial to build safe and responsible LLMs so that the effect of the misuse of AI technologies could be significantly alleviated.

We implement a multilingual safety evaluation that tests the LLMs in different languages. Specifically, we assess the safety performance of the models in the topics about illegal behaviors, fraud,

Multilingual 평가

- 다양한 언어의 Test Case 작성
- 각 언어별 전문 Annotator가 평가
- 각 Test에 대해 1~5점으로 평가

Safety 학습 평가

- Multilingual에서 평가 수행
- 불법, 사기, 음란물, 사생활 평가
- 탈옥에 취약한 프롬프트 수집하여 평가

Table 14: **Performance of models in safety evaluation.** We compare Qwen2-72B-Instruct with GPT-4 and Mixtral-8x22B-Instruct. The lower, the better. Qwen2-72B-Instruct rejected more prompts with risks than the competitors.

Risk Category	GPT-4	Mixtral-8x22B	Qwen2-72B-Instruct
Illegal	0.00	6.87	0.00
Fraud	3.40	8.49	2.41
Pornography	23.63	33.82	22.91
Privacy	3.37	15.03	2.47

Table 15: **Contamination Analysis.** The *contaminated* samples in this table are identified using a *strict criterion*: any test sample with a 13-gram overlap with the pre-training or post-training data is considered contaminated. We report the percentage of contaminated samples as well as the model performance on both the original and non-contaminated test sets.

Test set	Percent of Contamination	Qwen2-72B-Instruct			Qwen2-7B-Instruct		
		Original	Non-Contam.	Δ	Original	Non-Contam.	Δ
MMLU	11.2%	82.3	83.2	0.9	70.5	71.3	0.8
MMLU-Pro	11.6%	64.4	65.6	1.2	44.1	46.5	2.4
GPQA	1.0%	42.4	41.8	0.6	34.3	34.1	-0.2
HumanEval	75.0%	86.0	87.0	1.0	79.9	87.8	7.9
MBPP	29.6%	80.2	79.7	0.5	67.2	69.0	1.8
MultiPL-E	37.7%	69.2	69.2	0.0	59.1	58.9	-0.2
GSM8k	0.7%	93.2	92.8	-0.4	85.7	85.6	-0.1
Math	31.7%	69.0	74.6	5.6	52.9	57.6	4.7
IFEval	0.9%	77.6	77.4	-0.2	54.7	53.7	-1.0

pornography, and privacy. We have collected prompts prone to jail-breaking and use them to test whether the models can provide safe responses by rejection.

The results are presented in Table 14, where the proportion of harmful responses generated by the models are shown and the lower, the better. It can be observed that Qwen2-72B-Instruct performs better than the proprietary model, GPT-4, and significantly outperforms the open-weight model, Mixtral-8x22B-Instruct. However, we believe that there is still much room for our model to improve to be a safer and more responsible model, especially in terms of pornography, which is a conventionally difficult category to differentiate even for humans.

5.2.6 CONTAMINATION ANALYSIS

For large language models, what counts as contamination and how to run contamination analysis remain an active area of research (Ravaut et al., 2024; Golchin & Surdeanu, 2024; Sainz et al., 2023). In the following, we first introduce how we try to decontaminate the training corpora against the evaluation datasets, and then estimate the extent to which benchmark scores are influenced by the remaining contamination.

During the construction of the pre-training and post-training datasets, we exclude potentially contaminated data using n-gram matching. However, we found that this approach may lead to a high false negative rate, because there could be commonly used expressions, especially in mathematical and coding data. Therefore, we also applied another constraint based on the longest common subsequence (LCS). Specifically, we first remove all symbols and punctuation from both the test and training sequences and perform tokenization. For a training sequence s_t , we remove it if there is a test sequence s_e such that $|\text{LCS}(s_t, s_e)| \geq 13$ and $|\text{LCS}(s_t, s_e)| \geq 0.6 \times \min(|s_t|, |s_e|)$.

Decont할 때 n-gram 사용시
FN이 많이 발생함
(오염됐는데 안됐다고함)
이를 보완하기 위해 LCS 사용
(Longest Common Subsequence)

To assess the potential effects of leaking data on the test performance, we follow OpenAI (2023) to construct a *strict non-contaminated* test set to check if there is a significant performance degradation after *strict decontamination*. Specifically, we construct the non-contaminated test set by excluding any

sample which has 13-gram overlap with the pre-training or the post-training data (without constraint on LCS), and then compute the corresponding metric on the test set.

The results are presented in Table 15. Although some datasets exhibit a high percentage of contamination under the strict criterion, we noticed that most of the identified *contaminated* samples are false positives, primarily stemming from the mathematics and coding datasets. It is likely that certain code snippets and mathematical equations are so common that they do not provide any meaningful advantage in solving the test data. Furthermore, our analysis shows that the performance of the Qwen2 models remains consistent between the original and non-contaminated test data, suggesting that the potential issue of data contamination does not significantly impact the model’s performance.

6 CONCLUSION

This technical report has presented the Qwen2 series, a versatile suite of foundational and instruction-tuned language models, ranging from 0.5 to 72 billion parameters, including models of dense and Mixture-of-Experts architecture. Qwen2 outperforms previous open-weight models, notably its predecessor Qwen1.5, and displays competitive performance against proprietary models across a broad spectrum of benchmarks in language understanding, generation, multilingual capabilities, coding, mathematics, and reasoning. In this update, we have extra focus on long-context, multilingual, coding, mathematics capabilities and safety and responsibility. In a commitment to fostering innovation and accessibility within the community, we have made the Qwen2 model weights openly accessible, which enables researchers and developers to harness the full potential of Qwen2 in a variety of applications and research projects. Through these efforts, we aim to contribute to the advancement of AI technologies and their positive impact on society.

REFERENCES

- Marah Abdin, Jyoti Aneja, Sebastien Bubeck, Caio César Teodoro Mendes, Weizhu Chen, Allie Del Giorno, Ronen Eldan, Sivakanth Gopi, Suriya Gunasekar, Mojan Javaheripi, Piero Kauffmann, Yin Tat Lee, Yuanzhi Li, Anh Nguyen, Gustavo de Rosa, Olli Saarikivi, Adil Salim, Shital Shah, Michael Santacrose, Harkirat Singh Behl, Adam Taumann Kalai, Xin Wang, Rachel Ward, Philipp Witte, Cyril Zhang, and Yi Zhang. Phi-2: The surprising power of small language models, 2024. URL <https://www.microsoft.com/en-us/research/blog/phi-2-the-surprising-power-of-small-language-models/>.
- AI@Meta. Llama 3 model card, 2024. URL https://github.com/meta-llama/llama3/blob/main/MODEL_CARD.md.
- Joshua Ainslie, James Lee-Thorp, Michiel de Jong, Yury Zemlyanskiy, Federico Lebrón, and Sumit Sanghai. GQA: Training generalized multi-query Transformer models from multi-head checkpoints. In *EMNLP*, pp. 4895–4901. Association for Computational Linguistics, 2023.
- Chenxin An, Fei Huang, Jun Zhang, Shansan Gong, Xipeng Qiu, Chang Zhou, and Lingpeng Kong. Training-free long-context scaling of large language models. *CoRR*, abs/2402.17463, 2024.
- Anthropic. The Claude 3 model family: Opus, Sonnet, Haiku. Technical report, Anthropic, AI, 2024. URL https://www-cdn.anthropic.com/de8ba9b01c9ab7cbabf5c33b80b7bbcb618857627/Model_Card_Claude_3.pdf.
- Jacob Austin, Augustus Odena, Maxwell I. Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie J. Cai, Michael Terry, Quoc V. Le, and Charles Sutton. Program synthesis with large language models. *CoRR*, abs/2108.07732, 2021.
- Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, Binyuan Hui, Luo Ji, Mei Li, Junyang Lin, Runji Lin, Dayiheng Liu, Gao Liu, Chengqiang Lu, Keming Lu, Jianxin Ma, Rui Men, Xingzhang Ren, Xuancheng Ren, Chuanqi Tan, Sinan Tan, Jianhong Tu, Peng Wang, Shijie Wang, Wei Wang, Shengguang Wu, Benfeng Xu, Jin Xu, An Yang, Hao Yang, Jian Yang, Shusheng Yang, Yang Yao, Bowen Yu, Hongyi Yuan, Zheng Yuan, Jianwei Zhang, Xingxuan Zhang, Yichang Zhang, Zhenru Zhang, Chang Zhou, Jingren Zhou, Xiaohuan Zhou, and Tianhang Zhu. Qwen technical report. *CoRR*, abs/2309.16609, 2023a.
- Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-VL: A frontier large vision-language model with versatile abilities. *CoRR*, abs/2308.12966, 2023b.
- Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, Carol Chen, Catherine Olsson, Christopher Olah, Danny Hernandez, Dawn Drain, Deep Ganguli, Dustin Li, Eli Tran-Johnson, Ethan Perez, Jamie Kerr, Jared Mueller, Jeffrey Ladish, Joshua Landau, Kamal Ndousse, Kamile Lukosiute, Liane Lovitt, Michael Sellitto, Nelson Elhage, Nicholas Schiefer, Noemí Mercado, Nova DasSarma, Robert Lasenby, Robin Larson, Sam Ringer, Scott Johnston, Shauna Kravec, Sheer El Showk, Stanislav Fort, Tamera Lanham, Timothy Telleen-Lawton, Tom Conerly, Tom Henighan, Tristan Hume, Samuel R. Bowman, Zac Hatfield-Dodds, Ben Mann, Dario Amodei, Nicholas Joseph, Sam McCandlish, Tom Brown, and Jared Kaplan. Constitutional AI: Harmlessness from AI feedback. *CoRR*, abs/2212.08073, 2022.
- Lucas Bandarkar, Davis Liang, Benjamin Muller, Mikel Artetxe, Satya Narayan Shukla, Donald Husa, Naman Goyal, Abhinandan Krishnan, Luke Zettlemoyer, and Madian Khabsa. The Belebele benchmark: A parallel reading comprehension dataset in 122 language variants. *CoRR*, abs/2308.16884, 2023.
- Boxi Cao, Keming Lu, Xinyu Lu, Jiawei Chen, Mengjie Ren, Hao Xiang, Peilin Liu, Yaojie Lu, Ben He, Xianpei Han, Le Sun, Hongyu Lin, and Bowen Yu. Towards scalable automated alignment of LLMs: A survey. *CoRR*, abs/2406.01252, 2024.
- Federico Cassano, John Gouwar, Daniel Nguyen, Sydney Nguyen, Luna Phipps-Costin, Donald Pinckney, Ming-Ho Yee, Yangtian Zi, Carolyn Jane Anderson, Molly Q. Feldman, Arjun Guha, Michael Greenberg, and Abhinav Jangda. MultiPL-E: A scalable and polyglot approach to benchmarking neural code generation. *IEEE Trans. Software Eng.*, 49(7):3675–3691, 2023.

- Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Pondé de Oliveira Pinto, Jared Kaplan, Harrison Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder, Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mohammad Bavarian, Clemens Winter, Philippe Tillet, Felipe Petroski Such, Dave Cummings, Matthias Plappert, Fotios Chantzis, Elizabeth Barnes, Ariel Herbert-Voss, William Hebgen Guss, Alex Nichol, Alex Paino, Nikolas Tezak, Jie Tang, Igor Babuschkin, Suchir Balaji, Shantanu Jain, William Saunders, Christopher Hesse, Andrew N. Carr, Jan Leike, Joshua Achiam, Vedant Misra, Evan Morikawa, Alec Radford, Matthew Knight, Miles Brundage, Mira Murati, Katie Mayer, Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba. Evaluating large language models trained on code. *CoRR*, abs/2107.03374, 2021.
- Wenhu Chen, Ming Yin, Max Ku, Pan Lu, Yixin Wan, Xueguang Ma, Jianyu Xu, Xinyi Wang, and Tony Xia. TheoremQA: A theorem-driven question answering dataset. In *EMNLP*, pp. 7889–7901. Association for Computational Linguistics, 2023a.
- Zhihong Chen, Shuo Yan, Juhao Liang, Feng Jiang, Xiangbo Wu, Fei Yu, Guiming Hardy Chen, Junying Chen, Hongbo Zhang, Li Jianquan, Wan Xiang, and Benyou Wang. Multilingual-SIFT: Multilingual supervised instruction fine-tuning, 2023b. URL <https://github.com/FreedomIntelligence/MultilingualSIFT>.
- Wei-Lin Chiang, Lianmin Zheng, Ying Sheng, Anastasios Nikolas Angelopoulos, Tianle Li, Dacheng Li, Hao Zhang, Banghua Zhu, Michael I. Jordan, Joseph E. Gonzalez, and Ion Stoica. Chatbot arena: An open platform for evaluating LLMs by human preference. *CoRR*, abs/2403.04132, 2024.
- Yunfei Chu, Jin Xu, Xiaohuan Zhou, Qian Yang, Shiliang Zhang, Zhijie Yan, Chang Zhou, and Jingren Zhou. Qwen-Audio: Advancing universal audio understanding via unified large-scale audio-language models. *CoRR*, abs/2311.07919, 2023.
- Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? Try ARC, the AI2 reasoning challenge. *CoRR*, abs/1803.05457, 2018.
- 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. *CoRR*, abs/2110.14168, 2021.
- Damai Dai, Chengqi Deng, Chenggang Zhao, R. X. Xu, Huazuo Gao, Deli Chen, Jiashi Li, Wangding Zeng, Xingkai Yu, Y. Wu, Zhenda Xie, Y. K. Li, Panpan Huang, Fuli Luo, Chong Ruan, Zhifang Sui, and Wenfeng Liang. DeepSeekMoE: Towards ultimate expert specialization in mixture-of-experts language models. *CoRR*, abs/2401.06066, 2024.
- Yann N. Dauphin, Angela Fan, Michael Auli, and David Grangier. Language modeling with gated convolutional networks. In *ICML*, volume 70 of *Proceedings of Machine Learning Research*, pp. 933–941. PMLR, 2017.
- Guanting Dong, Hongyi Yuan, Keming Lu, Chengpeng Li, Mingfeng Xue, Dayiheng Liu, Wei Wang, Zheng Yuan, Chang Zhou, and Jingren Zhou. How abilities in large language models are affected by supervised fine-tuning data composition. *CoRR*, abs/2310.05492, 2023.
- Guanting Dong, Keming Lu, Chengpeng Li, Tingyu Xia, Bowen Yu, Chang Zhou, and Jingren Zhou. Self-play with execution feedback: Improving instruction-following capabilities of large language models. *CoRR*, abs/2406.13542, 2024.
- Alena Fenogenova, Artem Chervyakov, Nikita Martynov, Anastasia Kozlova, Maria Tikhonova, Albina Akhmetgareeva, Anton A. Emelyanov, Denis Shevelev, Pavel Lebedev, Leonid Sinev, Ulyana Isaeva, Katerina Kolomeytseva, Daniil Moskovskiy, Elizaveta Goncharova, Nikita Savushkin, Polina Mikhailova, Denis Dimitrov, Alexander Panchenko, and Sergey Markov. MERA: A comprehensive LLM evaluation in russian. *CoRR*, abs/2401.04531, 2024.
- Shahriar Golchin and Mihai Surdeanu. Time travel in llms: Tracing data contamination in large language models. In *ICLR*. OpenReview.net, 2024.

- Naman Goyal, Cynthia Gao, Vishrav Chaudhary, Peng-Jen Chen, Guillaume Wenzek, Da Ju, Sanjana Krishnan, Marc’Aurelio Ranzato, Francisco Guzmán, and Angela Fan. The Flores-101 evaluation benchmark for low-resource and multilingual machine translation. *Trans. Assoc. Comput. Linguistics*, 10:522–538, 2022.
- Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *ICLR. OpenReview.net*, 2021a.
- Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the MATH dataset. In *NeurIPS Datasets and Benchmarks*, 2021b.
- Yuzhen Huang, Yuzhuo Bai, Zhihao Zhu, Junlei Zhang, Jinghan Zhang, Tangjun Su, Junteng Liu, Chuancheng Lv, Yikai Zhang, Jiayi Lei, Yao Fu, Maosong Sun, and Junxian He. C-Eval: A multi-level multi-discipline chinese evaluation suite for foundation models. In *NeurIPS*, 2023.
- Naman Jain, King Han, Alex Gu, Wen-Ding Li, Fanjia Yan, Tianjun Zhang, Sida Wang, Armando Solar-Lezama, Koushik Sen, and Ion Stoica. LiveCodeBench: Holistic and contamination free evaluation of large language models for code. *CoRR*, abs/2403.07974, 2024.
- Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de Las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, L  lio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timoth  e Lacroix, and William El Sayed. Mistral 7B. *CoRR*, abs/2310.06825, 2023a.
- Albert Q. Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de Las Casas, Emma Bou Hanna, Florian Bressand, Gianna Lengyel, Guillaume Bour, Guillaume Lample, L  lio Renard Lavaud, Lucile Saulnier, Marie-Anne Lachaux, Pierre Stock, Sandeep Subramanian, Sophia Yang, Szymon Antoniak, Teven Le Scao, Th  ophile Gervet, Thibaut Lavril, Thomas Wang, Timoth  e Lacroix, and William El Sayed. Mixtral of experts. *CoRR*, abs/2401.04088, 2024.
- Zixuan Jiang, Jiaqi Gu, Hanqing Zhu, and David Z. Pan. Pre-RMSNorm and Pre-CRMSNorm Transformers: Equivalent and efficient pre-LN Transformers. *CoRR*, abs/2305.14858, 2023b.
- Gregory Kamradt. Needle in a haystack - pressure testing LLMs, 2023. URL https://github.com/gkamradt/LLMTest_NeedleInAHaystack.
- Aran Komatsuzaki, Joan Puigcerver, James Lee-Thorp, Carlos Riquelme Ruiz, Basil Mustafa, Joshua Ainslie, Yi Tay, Mostafa Dehghani, and Neil Houlsby. Sparse upcycling: Training mixture-of-experts from dense checkpoints. In *ICLR. OpenReview.net*, 2023.
- Fajri Koto, Nurul Aisyah, Haonan Li, and Timothy Baldwin. Large language models only pass primary school exams in Indonesia: A comprehensive test on IndoMMLU. In *EMNLP*, pp. 12359–12374. Association for Computational Linguistics, 2023.
- Haonan Li, Yixuan Zhang, Fajri Koto, Yifei Yang, Hai Zhao, Yeyun Gong, Nan Duan, and Timothy Baldwin. CMMLU: Measuring massive multitask language understanding in Chinese. *CoRR*, abs/2306.09212, 2023.
- Tianle Li, Wei-Lin Chiang, Evan Frick, Lisa Dunlap, Tianhao Wu, Banghua Zhu, Joseph E. Gonzalez, and Ion Stoica. From crowdsourced data to high-quality benchmarks: Arena-Hard and BenchBuilder pipeline. *CoRR*, abs/2406.11939, 2024.
- Opher Lieber, Barak Lenz, Hofit Bata, Gal Cohen, Jhonathan Osin, Itay Dalmedigos, Erez Safahi, Shaked Meirom, Yonatan Belinkov, Shai Shalev-Shwartz, Omri Abend, Raz Alon, Tomer Asida, Amir Bergman, Roman Glozman, Michael Gokhman, Avashalom Manevich, Nir Ratner, Noam Rozen, Erez Shwartz, Mor Zusman, and Yoav Shoham. Jamba: A hybrid Transformer-Mamba language model. *CoRR*, abs/2403.19887, 2024.
- Stephanie Lin, Jacob Hilton, and Owain Evans. TruthfulQA: Measuring how models mimic human falsehoods. In *ACL (1)*, pp. 3214–3252. Association for Computational Linguistics, 2022a.

- Xi Victoria Lin, Todor Mihaylov, Mikel Artetxe, Tianlu Wang, Shuohui Chen, Daniel Simig, Myle Ott, Naman Goyal, Shruti Bhosale, Jingfei Du, Ramakanth Pasunuru, Sam Shleifer, Punit Singh Koura, Vishrav Chaudhary, Brian O’Horo, Jeff Wang, Luke Zettlemoyer, Zornitsa Kozareva, Mona T. Diab, Veselin Stoyanov, and Xian Li. Few-shot learning with multilingual generative language models. In *EMNLP*, pp. 9019–9052. Association for Computational Linguistics, 2022b.
- Jiawei Liu, Chunqiu Steven Xia, Yuyao Wang, and Lingming Zhang. Is your code generated by ChatGPT really correct? Rigorous evaluation of large language models for code generation. In *NeurIPS*, 2023a.
- Xiao Liu, Xuanyu Lei, Shengyuan Wang, Yue Huang, Zhuoer Feng, Bosi Wen, Jiale Cheng, Pei Ke, Yifan Xu, Weng Lam Tam, Xiaohan Zhang, Lichao Sun, Hongning Wang, Jing Zhang, Minlie Huang, Yuxiao Dong, and Jie Tang. AlignBench: Benchmarking Chinese alignment of large language models. *CoRR*, abs/2311.18743, 2023b.
- Keming Lu, Bowen Yu, Fei Huang, Yang Fan, Runji Lin, and Chang Zhou. Online merging optimizers for boosting rewards and mitigating tax in alignment. *CoRR*, abs/2405.17931, 2024a.
- Keming Lu, Bowen Yu, Chang Zhou, and Jingren Zhou. Large language models are superpositions of all characters: Attaining arbitrary role-play via self-alignment. *CoRR*, abs/2401.12474, 2024b.
- Keming Lu, Hongyi Yuan, Zheng Yuan, Runji Lin, Junyang Lin, Chuanqi Tan, Chang Zhou, and Jingren Zhou. #InsTag: Instruction tagging for analyzing supervised fine-tuning of large language models. In *ICLR*. OpenReview.net, 2024c.
- Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, Pouya Tafti, Léonard Hussenot, Pier Giuseppe Sessa, 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 A. Choquette-Choo, 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, 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, Tris Warkentin, Ludovic Peran, Minh Giang, Clément Farabet, Oriol Vinyals, Jeff Dean, Koray Kavukcuoglu, Demis Hassabis, Zoubin Ghahramani, Douglas Eck, Joelle Barral, Fernando Pereira, Eli Collins, Armand Joulin, Noah Fiedel, Evan Senter, Alek Andreev, and Kathleen Kenealy. Gemma: Open models based on Gemini research and technology. *CoRR*, abs/2403.08295, 2024.
- 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. In *ACL (1)*, pp. 15991–16111. Association for Computational Linguistics, 2023.
- Jinjie Ni, Fuzhao Xue, Xiang Yue, Yuntian Deng, Mahir Shah, Kabir Jain, Graham Neubig, and Yang You. MixEval: Deriving wisdom of the crowd from LLM benchmark mixtures. *CoRR*, abs/2406.06565, 2024.
- OpenAI. Introducing ChatGPT, 2022. URL <https://openai.com/index/chatgpt/>.
- OpenAI. GPT4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.
- OpenAI. Hello GPT-4o, 2024. URL <https://openai.com/index/hello-gpt-4o/>.

- OpenCompass Contributors. OpenCompass: A universal evaluation platform for foundation models, 2023. URL <https://github.com/open-compass/opencompass>.
- Bowen Peng, Jeffrey Quesnelle, Honglu Fan, and Enrico Shippole. YaRN: Efficient context window extension of large language models. *CoRR*, abs/2309.00071, 2023.
- Edoardo Maria Ponti, Goran Glavas, Olga Majewska, Qianchu Liu, Ivan Vulic, and Anna Korhonen. XCOPA: A multilingual dataset for causal commonsense reasoning. In *EMNLP (1)*, pp. 2362–2376. Association for Computational Linguistics, 2020.
- Qwen Team. Introducing Qwen1.5, 2024a. URL <https://qwenlm.github.io/blog/qwen1.5/>.
- Qwen Team. Qwen1.5-110B: The first 100B+ model of the Qwen1.5 series, 2024b. URL <https://qwenlm.github.io/blog/qwen1.5-110b/>.
- Qwen Team. Qwen1.5-MoE: Matching 7B model performance with 1/3 activated parameters, 2024c. URL <https://qwenlm.github.io/blog/qwen-moe/>.
- Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D. Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In *NeurIPS*, 2023.
- Samyam Rajbhandari, Conglong Li, Zhewei Yao, Minjia Zhang, Reza Yazdani Aminabadi, Ammar Ahmad Awan, Jeff Rasley, and Yuxiong He. DeepSpeed-MoE: Advancing mixture-of-experts inference and training to power next-generation AI scale. In *ICML*, volume 162 of *Proceedings of Machine Learning Research*, pp. 18332–18346. PMLR, 2022.
- Mathieu Ravaut, Bosheng Ding, Fangkai Jiao, Hailin Chen, Xingxuan Li, Ruochen Zhao, Chengwei Qin, Caiming Xiong, and Shafiq Joty. How much are LLMs contaminated? A comprehensive survey and the llmsanitize library. *CoRR*, abs/2404.00699, 2024.
- David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R. Bowman. GPQA: A graduate-level Google-proof Q&A benchmark. *CoRR*, abs/2311.12022, 2023.
- Oscar Sainz, Jon Ander Campos, Iker García-Ferrero, Julen Etxaniz, Oier Lopez de Lacalle, and Eneko Agirre. NLP evaluation in trouble: On the need to measure LLM data contamination for each benchmark. In *EMNLP (Findings)*, pp. 10776–10787. Association for Computational Linguistics, 2023.
- Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. WinoGrande: An adversarial winograd schema challenge at scale. *Commun. ACM*, 64(9):99–106, 2021.
- Jianlin Su. The magical effect of the Bias term: RoPE + Bias = better length extrapolation, 2023. URL <https://spaces.ac.cn/archives/9577>.
- Jianlin Su, Murtadha H. M. Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. Roformer: Enhanced Transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024.
- Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc V. Le, Ed H. Chi, Denny Zhou, and Jason Wei. Challenging BIG-Bench tasks and whether chain-of-thought can solve them. In *ACL (Findings)*, pp. 13003–13051. Association for Computational Linguistics, 2023.
- Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurélien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. LLaMA: Open and efficient foundation language models. *CoRR*, abs/2302.13971, 2023.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NIPS*, pp. 5998–6008, 2017.

- Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyang Jiang, Tianle Li, Max Ku, Kai Wang, Alex Zhuang, Rongqi Fan, Xiang Yue, and Wenhui Chen. MMLU-Pro: A more robust and challenging multi-task language understanding benchmark. *CoRR*, abs/2406.01574, 2024.
- Wenhui Xiong, Jingyu Liu, Igor Molybog, Hejia Zhang, Prajjwal Bhargava, Rui Hou, Louis Martin, Rashmi Rungta, Karthik Abinav Sankararaman, Barlas Oguz, Madihan Khabisa, Han Fang, Yashar Mehdad, Sharan Narang, Kshitiz Malik, Angela Fan, Shruti Bhosale, Sergey Edunov, Mike Lewis, Sinong Wang, and Hao Ma. Effective long-context scaling of foundation models. *CoRR*, abs/2309.16039, 2023.
- Yinfei Yang, Yuan Zhang, Chris Tar, and Jason Baldridge. PAWS-X: A cross-lingual adversarial dataset for paraphrase identification. In *EMNLP/IJCNLP (1)*, pp. 3685–3690. Association for Computational Linguistics, 2019.
- Alex Young, Bei Chen, Chao Li, Chengen Huang, Ge Zhang, Guanwei Zhang, Heng Li, Jiangcheng Zhu, Jianqun Chen, Jing Chang, Kaidong Yu, Peng Liu, Qiang Liu, Shawn Yue, Senbin Yang, Shiming Yang, Tao Yu, Wen Xie, Wenhao Huang, Xiaohui Hu, Xiaoyi Ren, Xinyao Niu, Pengcheng Nie, Yuchi Xu, Yudong Liu, Yue Wang, Yuxuan Cai, Zhenyu Gu, Zhiyuan Liu, and Zonghong Dai. Yi: Open foundation models by 01.AI. *CoRR*, abs/2403.04652, 2024.
- Tao Yuan, Xuefei Ning, Dong Zhou, Zhijie Yang, Shiyao Li, Minghui Zhuang, Zheyue Tan, Zhuyu Yao, Dahua Lin, Boxun Li, Guohao Dai, Shengen Yan, and Yu Wang. LV-Eval: A balanced long-context benchmark with 5 length levels up to 256K. *CoRR*, abs/2402.05136, 2024.
- Zheng Yuan, Hongyi Yuan, Chengpeng Li, Guanting Dong, Chuanqi Tan, and Chang Zhou. Scaling relationship on learning mathematical reasoning with large language models. *CoRR*, abs/2308.01825, 2023.
- Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. Hellaswag: Can a machine really finish your sentence? In *ACL (1)*, pp. 4791–4800. Association for Computational Linguistics, 2019.
- Aohan Zeng, Bin Xu, Bowen Wang, Chenhui Zhang, Da Yin, Diego Rojas, Guanyu Feng, Hanlin Zhao, Hanyu Lai, Hao Yu, Hongning Wang, Jiadao Sun, Jiajie Zhang, Jiale Cheng, Jiayi Gui, Jie Tang, Jing Zhang, Juanzi Li, Lei Zhao, Lindong Wu, Lucien Zhong, Mingdao Liu, Minlie Huang, Peng Zhang, Qinkai Zheng, Rui Lu, Shuaiqi Duan, Shudan Zhang, Shulin Cao, Shuxun Yang, Weng Lam Tam, Wenyi Zhao, Xiao Liu, Xiao Xia, Xiaohan Zhang, Xiaotao Gu, Xin Lv, Xinghan Liu, Xinyi Liu, Xinyue Yang, Xixuan Song, Xunkai Zhang, Yifan An, Yifan Xu, Yilin Niu, Yuantao Yang, Yueyan Li, Yushi Bai, Yuxiao Dong, Zehan Qi, Zhaoyu Wang, Zhen Yang, Zhengxiao Du, Zhenyu Hou, and Zihan Wang. ChatGLM: A family of large language models from GLM-130B to GLM-4 all tools. *CoRR*, abs/2406.12793, 2024.
- Yingxiu Zhao, Bowen Yu, Binyuan Hui, Haiyang Yu, Minghao Li, Fei Huang, Nevin L. Zhang, and Yongbin Li. Tree-Instruct: A preliminary study of the intrinsic relationship between complexity and alignment. In *LREC/COLING*, pp. 16776–16789. ELRA and ICCL, 2024.
- 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. Judging LLM-as-a-judge with MT-Bench and Chatbot Arena. In *NeurIPS*, 2023.
- Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. Instruction-following evaluation for large language models. *CoRR*, abs/2311.07911, 2023.