# Exploring the Memory Ability of Large Language Models

Anonymous ACL submission

## Abstract

Memory capability is a critical aspect of large language models (LLMs). However, the disparity in memory ability between small and large LLMs remains unclear. In this paper, we present a novel investigation into the memory capabilities of both small and large LLMs, introducing an innovative knowledge-based dataset, enriched with frequency annotations. Derived from Wikidata5M, this dataset quantifies fact frequency by counting the co-occurrences of head and tail entities in Wikipedia and Baidu Baike documents. Building upon this, we constructed a fact-based question-answering dataset called KDF, and evaluated the memory performance of state-of-the-art pre-trained base model families. Our comprehensive experiments demonstrate that large LLMs exhibit robust memory capabilities, retaining most facts even when they occur infrequently. Conversely, small LLMs are limited to recalling only a subset of high-frequency facts, struggling significantly with low-frequency information. Our study not only illuminates the memory discrepancies between different scales of LLMs but also offers a valuable resource and methodology for future research in LLMs.

## 1 Introduction

Large language models (LLMs) have become the focus in the past few years. They can handle many fact-based NLP tasks without further fine-tuning(Petroni et al., 2019). This phenomenon suggests that LLMs are capable of recalling facts from their pretraining data. There are two main directions in the development of LLMs: small LLMs and large LLMs. Small LLMs, such as Phi-2(Li et al., 2023), Qwen1.5-1.8B (QwenTeam, 2024a) and so on, usually have fewer than 4 billion parameters. In contrast, large LLMs like Llama 3-70B, Qwen1.5-110B, which have significantly more parameters. While both small and large LLMs are valuable in their respective contexts, the distinction between their memory capabilities remains unclear. Moreover, in the past year, especially after the born of ChatGPT[1], a large number of new LLMs have emerged. The newly released models like Llama 3 (AI@Meta, 2024) are much well trained than previous models. Exploring the memory ability of these models are necessary. Understanding the differences in how these models retain and recall information from their pretraining data is crucial for optimizing their application in various NLP tasks.

Some previous work like Mallen et al. (2022), Kandpal et al. (2023), Carlini et al. (2022) and Sun et al. (2024) show that the memory ability is strongly related to how many times a fact has appeared in pre-training data. However, counting the exact frequency is challenging. Yu et al. (2023) sort the entities according to their frequency of occurrence in Wikipedia (Jin et al., 2019), which is used to identify high/low frequency knowledge. However, entities with high frequency in Wikipeida doesn't mean they would co-occor with high frequency. Mallen et al. (2022) uses the Wikipedia monthly page views as an approximation. Similarly, Sun et al. (2024) approximate the frequency with traffic (such as views and votes) and density (such as the number of facts about the entity). Although the views are much easier to acquire, there is still a distance between views and frequency. Kandpal et al. (2023) first run entity linking on pre-training dataset. Then they extract and link entities from downstream question answer pairs. Finally, they count the co-occur documents of question entity and answer entity as the fact frequency. The datasets they consider including The Pile (Gao et al., 2021), ROOTS(en) (Laurençon et al., 2023), WikiPedia (Lee et al., 2019) and so on.

In this paper, we undertake a comprehensive investigation into the memory capabilities of several newly released model families, including Llama

[1] https://openai.com/blog/chatgpt

2 family (Touvron et al., 2023), Llama 3 family (AI@Meta, 2024), Qwen1.5 family (QwenTeam, 2024a), Qwen2 family (QwenTeam, 2024b), and Yi family (01.AI et al., 2024). Models within a family typically share the same pretraining data distribution, which facilitates a systematic comparison of the memory capabilities between small and large LLMs. To this end, we introduce **KDF**, a novel **K**nowledge fact **D**ataset with **F**requency annotations. KDF is derived from Wikidata5M (Wang et al., 2019), and we meticulously count the co-occurrences of head and tail entities in both the Wikipedia and Baidu Baike datasets to establish frequency metrics. Given the prevalence of Wikipedia and Baidu Baike as foundational fact-based datasets, we assume they have been extensively utilized in the training processes of the model families under consideration.

Our approach not only allows for an accurate assessment of memory retention across different model scales but also provides insights into the nuances of fact recall capabilities, paving the way for future advancements in LLM architecture and training methodologies.

## 2 Related Work

**Memorization Ability** One of the seminal works in evaluating the factual and commonsense knowledge of language models is the LAMA (LAnguage Model Analysis) probe introduced by (Petroni et al., 2019). LAMA provides a set of knowledge sources composed of facts, formatted as either subject-relation-object triples or question-answer pairs. These facts are converted into cloze statements, which are used to query the language model for missing tokens. The evaluation metric is based on how highly the model ranks the ground truth token against other words in a fixed candidate vocabulary. KoLA (Yu et al., 2023) emulates human cognitive processes to develop a four-level classification of knowledge-related abilities, with the lowest level focusing on knowledge memorization. Frequency is defined based on the occurrence of entities in Wikipedia. It examines the correlation between memorization and training frequency by creating high-frequency and low-frequency test sets, by selecting 100 entities from the top 2,000 and from the least frequent entities, respectively. Unlike KoLA, we define frequency based on the co-occurrence of head and tail entities, and we further refine the frequency intervals into more granular categories.

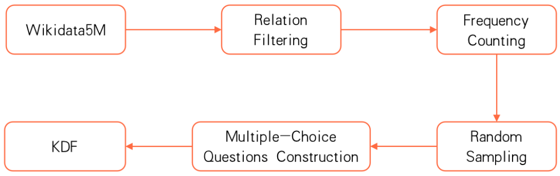



Figure 1: The progress of building KDF.

**Knowledge Frequency** The definition of knowledge frequency is not unique. Some studies use popularity as a proxy for frequency. PopQA(Mallen et al., 2022) use Wikipedia page views as a measure of popularity and convert knowledge triples from Wikidata, with diverse levels of popularity, into natural language questions, anchored to the original entities and relationship types. (Sun et al., 2024) proposed Head-to-Tail, which uses two ways to approximate popularity: traffic and density. When there is traffic information, such as views and votes, they conveniently use traffic to measure the popularity; otherwise, they use density as a proxy, such as the number of facts or authored works about the entity. Since our focus is on pre-trained base models, current popularity data may not be applicable to earlier versions of these models. Additionally, as we are primarily concerned with factual knowledge, popularity data tends to introduce significant noise.

We adopt an alternative method for obtaining frequency, specifically using the co-occurrence counts of entities in the training dataset as a proxy for frequency. (Kandpal et al., 2023) studied the relationship between the knowledge memorized by large language models and the information in pre-training datasets scraped from the web. It starts by identifying the salient entities within a question and its set of ground-truth answer variations. Next, relevant pre-training documents are identified by searching for instances where the key entities from the question and the answer co-occur. Our method for determining knowledge frequency is similar, but instead of using existing QA datasets, we construct knowledge based on Wikidata knowledge graph triplets. This approach is more direct and avoids the potential inaccuracies associated with entity extraction.

## 3 KDF

Investigating the frequency of knowledge occurrences in pre-training corpora presents several significant challenges.

Firstly, the manifestation of knowledge is inherently diverse. A single piece of knowledge can be conveyed through various expressions, and an entity may be known by different aliases or names. To accurately locate specific knowledge within unstructured pre-training data, it is crucial to employ techniques that structure the knowledge into a standardized format. To tackle this, we opted to use knowledge graph, representing knowledge in the form of triples, where each triple consists of a *(head entity, relation, tail entity)*. We consider each triple to represent a distinct piece of knowledge. We defined the frequency of knowledge occurrences as the number of times the head and tail entities co-occur in the retrieval corpus. This structured representation serves as the foundation for constructing our prompt questions.

Another key challenge is about the pre-training data. Extracting factual knowledge from unstructured pre-training data is inherently difficult due to the immense size of these datasets and the lack of transparency regarding the sources and composition of the pre-training material used by most models, making comprehensive searches impractical. Pre-training datasets like Common Crawl, which are derived from web data, are often unstructured and contain substantial noise. Conducting searches without appropriate filtering would inevitably result in inaccurate frequency statistics. To address these issues, we restricted our search scope to Wikipedia and Baidu Baike, as these two corpora are the most widely used factual knowledge bases in the English and Chinese domains respectively, and they maintain relatively clean data.

We propose KDF, as shown in Figure 1, a knowledge-based question-answering benchmark, which is designed to evaluate the performance of large language models (LLMs) across knowledge of varying frequencies, and ensure a more accurate representation of knowledge distribution in the pre-training corpora. Since we focus on factual knowledge, we use the triplets provied by ikidata5M(Wang et al., 2019), which is a high-quality subset of Wikidata containing about 5M entities, 20M triplets, and aligned entity descriptions.

We filtered out relations with the following characteristics: 1) the relation contains too few triples, 2) it is highly subjective or ambiguous (e.g., "topic", "symptom"), 3) the relation encompasses too few tail entities. These selected relations cover multiple domains such as literature and art, geography, business, and politics.

| Frequency Range | Number |
|---|---|
| 0 | 872 |
| $[1, 10)$ | 646 |
| $[10, 100)$ | 577 |
| $[100, inf)$ | 674 |

Table 1: Frequency distribution of KDF

We obtain the head and tail entities of the triples encompassed by these relations. We get the entity name from Wikidata dumps and we only use its Chinese name. We then search for the co-occurrence frequencies of each triple's head and tail entities in both Baidu Baike and Wikipedia. This co-occurrence frequency serves as a proxy for the frequency of the knowledge represented by each triple(Elsahar et al., 2018). For instance, consider the triple (英国, 首都, 伦敦). We take the head entity "英国" and the tail entity "伦敦" and calculate the number of documents that mention both entities.

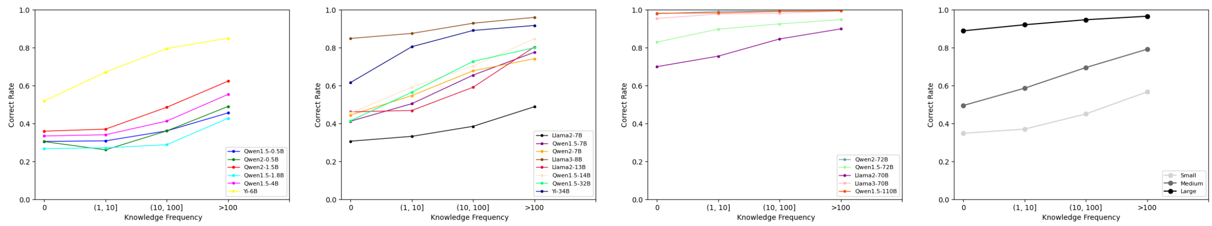
Then, we randomly select some triples from each frequency range as our candidate triples. The frequency ranges including: 0, $[1, 10)$, $[10, 100)$ and $[100, inf)$. We refer frequency $\leq 10$ as low frequency and $> 100$ as high frequency.

We aim to have the model predict the tail entity given the head entity and the relation. To ensure natural phrasing, we use a template-based approach to generate the questions. For each triple, we use the sentence pattern "The [relation] of [head] is [masked]," where the [masked] represents the tail entity that the model needs to predict. Given the numerous aliases for entities, we use a multiple-choice format to facilitate post-processing, requiring the model to choose the correct answer from four options (A, B, C, and D). To generate distractors for each question, we randomly sample from all tail entities under the current relation, ensuring that the sampled options do not form a valid triple with the given head entity.

Finally, there are 2964 triples, including 17 distinct relations. The frequency distribution is shown in Table 1. The relation and it's template is shown in Appendix Table 3.

## 4 Experiments

We evaluated the memory capability of models with different parameter sizes on our newly proposed benchmark, focusing on knowledge across various frequency ranges. Our evaluation concen-




(a) (b) (c) (d)

Figure 2: Overall results. (a), (b), and (c) show the results of small, medium, and large-scale models, respectively. (d) presents the average performance of them.

trated on state-of-the-art pre-trained base model families, covering a wide range of parameter sizes, including Llama 2 (7B, 13B, 70B), Llama 3 (8B, 70B), Qwen1.5 (0.5B, 1.8B, 4B, 7B, 14B, 32B, 72B, 110B), Qwen2(0.5B, 1.5B, 7B, 72B), and Yi (6B, 34B).

We evaluating these base models in 5-shot format. An example is shown in Appendix Figure 3. We using the logits of A, B, C and D from the first generated token, and using the one with maximum value as the predict result. This result is stable for different runs. We use accuracy as our evaluation metric.

### 4.1 Overall Results

As shown in Figure 2, the scores of small LLMs like Qwen2-1.5B, Qwen1.5-4B rising with increasing frequency. As each question in KDF has four options, a random baseline could acquire acc of 0.25. Small LLMs are little better than random baseline in low frequency range, which means they can barely remember low frequency knowledge.

For the middle sized models like Yi-6B, Llama 2-7B, Llama 2-13B, Qwen1.5-32B and so on, they perform better than small LLMs but their performance trend is similar to small LLMs.

The large LLMs like Qwen1.5-110B performs well even if the frequency is low. They perform even better on the high frequency knowledge. This phenomenon demonstrate that large LLMs have good memory, they could remember facts from pretraining data even with low frequency.

Scores of each model are shown in Appendix Table 2.

### 4.2 Discussion

The models in one model family may not necessary to be pretrained with the same amount of data, could this factor cause the difference? For example, Qwen(Bai et al., 2023) report that Qwen-1.8B was trained with 2.2T tokens, Qwen-7B was trained with 2.4T tokens, and Qwen-14B was trained with 3T tokens. Although Qwen1.5 and Qwen2 didn't reveal the details, we could assume that they are different. However, Yi, Llama 2 and Llama 3 family report the details of their pretraining data, models in these families are trained with the same amount of data. With the model size as the only difference, there is a significant difference of their memory ability.

Why our conclusion is different to Kola(Yu et al., 2023)? They found that many models perform worse on high frequency knowledge[2]. They first find the highest/lowest frequency entities according to their occurrence in Wikipedia. Then, they randomly select 100 entities with highest/lowest entities to construct triples, which named as high/low frequency knowledge. However, entities with high frequency in Wikipeida doesn't mean they co-occor with high frequency. Therefore, the "high frequency knowledge" may contains low frequency facts, which lead to lower scores.

Compared with Llama 2, which pretrained with 2T tokens, Llama 3 was pretrained with 15T tokens. In our experiment, Llama 3 remember much more knowledge than Llama 2. The more the pretraining data, the better the model was trained as a language model. And more pretraining data means the model potentially trained a knowledge fact more times.

## 5 Conclusions

We investigate the memory ability of some newly released model families like Llama 3 and Qwen 2. Our experiments find that large LLMs has strong memory ability. Small LLMs, on the contrary, can only remember part of the high frequency facts, not to mention low frequency facts.

[2]See Table 2 in Yu et al. (2023). Models like GPT-4, GPT-3.5-turbo acquire lower score in 1-1 (high frequency knowledge), compared with 1-2 (low frequency knowledge).

## 6 Limitations

It's very difficult to count the frequency of a fact in pretraining data due to the diversity of natural language expression and immense size of pretraining data. As an approximation, we count the co-occur of the entity pair in Wikipeida and Baidu Baike as the fact's proxy frequency. However, this counting method may underestimate the frequency of a fact. As shown in Figure 2, models could acquire scores when the fact's frequency is 0. It doesn't means that the model could learn something that has never been shown in the pretraining data. It just means that there are no document in Wikipedia and Baidu Baike that contains the identical entity names. There could be some entity alias that we didn't consider in our method.

Another limitation of this work is we assume each model has trained on Wikipedia and Baidu Baike. But models like Qwen1.5 family and Qwen2 didn't report details of their pretraining data. Our assumption may not hold.

## References


01.AI, : 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. 2024. Yi: Open foundation models by 01.ai. *Preprint*, arXiv:2403.04652.

AI@Meta. 2024. Llama 3 model card.

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. 2023. Qwen technical report. *Preprint*, arXiv:2309.16609.

Nicholas Carlini, Daphne Ippolito, Matthew Jagielski, Katherine Lee, Florian Tramer, and Chiyuan Zhang. 2022. Quantifying memorization across neural language models. *arXiv preprint arXiv:2202.07646*.

Hady Elsahar, Pavlos Vougiouklis, Arslen Remaci, Christophe Gravier, Jonathon Hare, Frederique Laforest, and Elena Simperl. 2018. T-rex: A large scale alignment of natural language with knowledge base triples. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.

Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, Shawn Presser, and Connor Leahy. 2021. The pile: An 800gb dataset of diverse text for language modeling. *CoRR*, abs/2101.00027.

Hailong Jin, Chengjiang Li, Jing Zhang, Lei Hou, Juanzi Li, and Peng Zhang. 2019. Xlore2: large-scale cross-lingual knowledge graph construction and application. *Data Intelligence*, 1(1):77–98.

Nikhil Kandpal, Haikang Deng, Adam Roberts, Eric Wallace, and Colin Raffel. 2023. Large language models struggle to learn long-tail knowledge. In *International Conference on Machine Learning*, pages 15696–15707. PMLR.

Hugo Laurençon, Lucile Saulnier, Thomas Wang, Christopher Akiki, Albert Villanova del Moral, Teven Le Scao, Leandro Von Werra, Chenghao Mou, Eduardo González Ponferrada, Huu Nguyen, Jörg Frohberg, Mario Šaško, Quentin Lhoest, Angelina McMillan-Major, Gerard Dupont, Stella Biderman, Anna Rogers, Loubna Ben allal, Francesco De Toni, Giada Pistilli, Olivier Nguyen, Somaieh Nikpoor, Maraim Masoud, Pierre Colombo, Javier de la Rosa, Paulo Villegas, Tristan Thrush, Shayne Longpre, Sebastian Nagel, Leon Weber, Manuel Muñoz, Jian Zhu, Daniel Van Strien, Zaid Alyafeai, Khalid Almubarak, Minh Chien Vu, Itziar Gonzalez-Dios, Aitor Soroa, Kyle Lo, Manan Dey, Pedro Ortiz Suarez, Aaron Gokaslan, Shamik Bose, David Adelani, Long Phan, Hieu Tran, Ian Yu, Suhas Pai, Jenny Chim, Violette Lepercq, Suzana Ilic, Margaret Mitchell, Sasha Alexandra Luccioni, and Yacine Jernite. 2023. The bigscience roots corpus: A 1.6tb composite multilingual dataset. *Preprint*, arXiv:2303.03915.

Kenton Lee, Ming-Wei Chang, and Kristina Toutanova. 2019. Latent retrieval for weakly supervised open domain question answering. *CoRR*, abs/1906.00300.

Yuanzhi Li, Sébastien Bubeck, Ronen Eldan, Allie Del Giorno, Suriya Gunasekar, and Yin Tat Lee. 2023. Textbooks are all you need ii: phi-1.5 technical report. *arXiv preprint arXiv:2309.05463*.

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. 2022. When not to trust language models: Investigating effectiveness of parametric and non-parametric memories. *arXiv preprint arXiv:2212.10511*.

Fabio Petroni, Tim Rocktäschel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, Alexander H Miller, and Sebastian Riedel. 2019. Language models as knowledge bases? *arXiv preprint arXiv:1909.01066*.

QwenTeam. 2024a. Introducing qwen1.5.

QwenTeam. 2024b. Qwen2 technical report.

Kai Sun, Yifan Ethan Xu, Hanwen Zha, Yue Liu, and Xin Luna Dong. 2024. Head-to-tail: How knowledgeable are large language models (llm),aka will llms replace knowledge graphs.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models. *Preprint*, arXiv:2307.09288.

Xiaozhi Wang, Tianyu Gao, Zhaocheng Zhu, Zhiyuan Liu, Juanzi Li, and Jian Tang. 2019. KEPLER: A unified model for knowledge embedding and pre-trained language representation. *CoRR*, abs/1911.06136.

Jifan Yu, Xiaozhi Wang, Shangqing Tu, Shulin Cao, Daniel Zhang-Li, Xin Lv, Hao Peng, Zijun Yao, Xiaohan Zhang, Hanming Li, et al. 2023. Kola: Carefully benchmarking world knowledge of large language models. *arXiv preprint arXiv:2306.09296*.

# A Appendix

| Model | 0 | [1,10) | [10,100) | [100,inf) | all |
|---|---|---|---|---|---|
| Qwen1.5-0.5B | 0.305 | 0.308 | 0.360 | 0.455 | 0.351 |
| Qwen2-0.5B | 0.304 | 0.260 | 0.360 | 0.490 | 0.345 |
| Qwen2-1.5B | 0.359 | 0.370 | 0.485 | 0.623 | 0.447 |
| Qwen1.5-1.8B | 0.267 | 0.271 | 0.288 | 0.427 | 0.309 |
| Qwen1.5-4B | 0.334 | 0.340 | 0.412 | 0.553 | 0.401 |
| Yi-6B | 0.518 | 0.671 | 0.795 | 0.850 | 0.691 |
| Llama2-7B | 0.306 | 0.332 | 0.385 | 0.488 | 0.370 |
| Qwen1.5-7B | 0.412 | 0.504 | 0.655 | 0.776 | 0.568 |
| Qwen2-7B | 0.444 | 0.547 | 0.678 | 0.742 | 0.586 |
| Llama3-8B | 0.849 | 0.875 | 0.929 | 0.960 | 0.897 |
| Llama2-13B | 0.461 | 0.468 | 0.591 | 0.804 | 0.567 |
| Qwen1.5-14B | 0.455 | 0.591 | 0.704 | 0.847 | 0.631 |
| Qwen1.5-32B | 0.414 | 0.565 | 0.728 | 0.800 | 0.606 |
| Yi-34B | 0.616 | 0.805 | 0.891 | 0.917 | 0.792 |
| Qwen2-72B | 0.979 | 0.990 | 0.993 | 0.994 | 0.989 |
| Qwen1.5-72B | 0.829 | 0.898 | 0.925 | 0.948 | 0.894 |
| Llama2-70B | 0.700 | 0.755 | 0.846 | 0.899 | 0.789 |
| Llama3-70B | 0.954 | 0.977 | 0.981 | 0.994 | 0.975 |
| Qwen1.5-110B | 0.982 | 0.983 | 0.991 | 0.994 | 0.987 |

Table 2: Accuracy of all models across different frequency intervals and their overall accuracy. All values are presented as percentages with three decimal places.

| **Relation** | | **Count** | **Template** |
|---|---|---|---|
| **English** | **Chinese** | | |
| genre | 类型 | 824 | [头实体]的类型是[尾实体]。 |
| cast member | 演员 | 820 | [头实体]的演员是[尾实体]。 |
| member of | 成员属于 | 274 | [头实体]属于[尾实体]的成员。 |
| capital | 行政中心 | 190 | [头实体]的行政中心是[尾实体]。 |
| director | 导演 | 167 | [头实体]的导演是[尾实体]。 |
| author | 作者 | 119 | [头实体]的作者是[尾实体]。 |
| discoverer or inventor | 发现者或发明者 | 86 | [头实体]的发现者或发明者是[尾实体]。 |
| composer | 作曲者 | 67 | [头实体]的作曲者是[尾实体]。 |
| present in work | 登场作品 | 67 | [头实体]是中[尾实体]的人物。 |
| producer | 制作人 | 64 | [头实体]的制作人是[尾实体]。 |
| political ideology | 政治意识形态 | 59 | [头实体]的政治意识形态是[尾实体]。 |
| publisher | 出版者 | 51 | [头实体]的出版者是[尾实体]。 |
| developer | 开发者 | 45 | [头实体]的开发者是[尾实体]。 |
| production company | 制作商 | 43 | [头实体]的制作商是[尾实体]。 |
| is the study of | 研究对象 | 35 | [头实体]的研究对象是[尾实体]。 |
| creator | 创作作者 | 32 | [头实体]的创作作者是[尾实体]。 |
| residence | 居住地 | 26 | [头实体]的创作作者是[尾实体]。 |

Table 3: Name of the relationships, number of corresponding data items, and template.

```
《雪岭过江龙》的演员是______。
A:  安吉·迪金森
B: 艾德·毕夏普
C: 阿图罗·格茨
D: 汉娜·博奇森纽斯
回答:A

小行星9051的发现者或发明者是______。
A: 法兰兹·安东·梅斯梅尔
B: 威廉·赫歇尔
C:  上田清二
D: 北京天文台
回答:C

《波斯王子: 遗忘之沙》的出版者是______。
A: 米高梅互动娱乐公司
B:  育碧
C: 普罗米修斯出版社
D: 影子经纪人
回答:B

《我要做警察》的制作商是______。
A: BBC新闻
B: 全景电影发行公司
C: 梦工厂经典影业公司
D:  传奇电影公司
回答:D

匈牙利人民共和国属于______的成员。
A: DOWN TOWN
B:  华沙条约组织
C: 印度斯坦共和协会
D: Infinite
回答:B

巴西的行政中心是______。
A: 萨尔塞罗区
B: 沃伦顿 (北开普省)
C: 塔威塔威
D:  里约热内卢
回答:
```

Figure 3: An example of 5-shot format.