Physics of Language Models: Part 3.1, Knowledge Storage and Extraction

Zeyuan Allen-Zhu [1] **Yuanzhi Li** [2]

Abstract

Large language models (LLMs) can store a vast amount of world knowledge, often extractable via question-answering (e.g., “What is Abraham Lincoln’s birthday?”). However, do they answer such questions based on exposure to similar questions during training (i.e., cheating), or by genuinely learning to extract knowledge from sources like Wikipedia? In this paper, we investigate this issue using a controlled biography dataset. We find a strong correlation between the model’s ability to extract knowledge and various *diversity measures* of the training data. **Essentially**, for knowledge to be reliably extracted, it must be sufficiently augmented (e.g., through paraphrasing, sentence shuffling) *during pretraining*. Without such augmentation, knowledge may be memorized but not extractable, leading to 0% accuracy, regardless of subsequent instruction fine-tuning. To understand why this occurs, we employ (nearly) linear probing to demonstrate a strong connection between the observed correlation and *how the model internally encodes knowledge* — whether it is linearly encoded in the hidden embeddings of entity names or distributed across other token embeddings in the training text. **This paper provides several key recommendations for LLM pretraining in the industry: (1) rewrite the pretraining data — using small, auxiliary models — to provide knowledge augmentation, and (2) incorporate more instruction-finetuning data into the pretraining stage before it becomes too late.**

1. Introduction

Knowledge is crucial for human cognition and communication, allowing us to comprehend and utilize information. For humans, this often involves memorization, the process of storing and retrieving information in the brain. For example, after reading a biography of Abraham Lincoln, we can memorize the information and later answer questions like “Where was Lincoln born?” or “What is Lincoln’s birthday?” Memorization enables us to extract and manipulate knowledge from the sentences we read or hear, recognize the entities, relations, and facts expressed in the text, and apply logical and causal reasoning to infer new information or answer queries (Anderson & Milson, 1989; Baddeley, 1997; Craik & Jennings, 1992; Zlotnik & Vansintjan, 2019).

In this paper, we explore **how transformer-based language models** memorize knowledge during training and extract it during inference. This is distinct from in-context learning or RAG (Lewis et al., 2020), where the model is given a paragraph during inference and immediately answers questions about it. We focus on *factual knowledge* (e.g., knowledge graph) that a language model needs to memorize from the training corpus, encode in its weights, and extract later during inference.

We stress that *memorizing* all sentences in the training data **does not** ensure that the model can *extract or manipulate* the factual knowledge from the sentences during inference. Language models can reproduce the exact input during inference, but this doesn’t necessarily mean they can use these sentences to answer factual questions related to them. Hence, we differentiate between “memorization of knowledge” in language models and traditional memorization in machine learning, which merely means the model can fit the exact training data, but doesn’t imply the model can **extract the knowledge flexibly** from the data after training.

For example, if the training data includes Lincoln’s biography, the model can memorize and reproduce the sentence “Abraham Lincoln was born in Hodgenville, K.Y.” when given the prompt “Abraham Lincoln was born in”, but it might not be able to answer the question “Which city was Abraham Lincoln born in?” Therefore, a key question is:

How do language models memorize knowledge during

An extended video of this paper is available at `https://youtu.be/YSHzKmEianc`. Full and future editions of the paper can be found on `https://arxiv.org/abs/2309.14316`. [1]Meta / FAIR Labs, USA [2]MBZUAI, UAE. Correspondence to: Zeyuan Allen-Zhu <zeyuanallenzhu@meta.com>, Yuanzhi Li <Yuanzhi.Li@mbzuai.ac.ae>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

training, and extract it later to answer questions or perform logical reasoning during inference?

Previous works have demonstrated that language models can "memorize" a lot of knowledge by probing the model to answer questions related to different entities and attributes, see (Omar et al., 2023; Singhal et al., 2022; Sun et al., 2023) and the citations therein. However, these studies use models trained on internet data, leaving it **unclear** whether the model answers questions like "Which city was Abraham Lincoln born in?" by *extracting knowledge* from Lincoln's biography (our focus) or if it encountered a similar (or same!) question during training and simply memorized the answer (traditional memorization).

Given the challenges of conducting controlled experiments with internet data, we propose studying this question using well-controlled, synthetically generated data,[1] examining the models' mathematical properties that characterize their knowledge representation and extraction. We construct a synthetic dataset of $100k$ biographies, including their birthday, birth city, major of study, etc. We also use LLaMA (Touvron et al., 2023) to rewrite them to make them close to real-life biography styles. We pretrain the language model on the biography dataset of all the $100k$ people. We ask:[2]

After pretraining a language model on the biography dataset, can the model be finetuned to answer questions like "Where is the birth city of [name]", and if so, how does the model achieve so?

After pretraining the model on the entire biography, we fine-tune it using question and answer (QA) pairs from a p fraction of individuals. We then test its ability to *out-of-distribution* answer QAs about the remaining $1-p$ fraction. This approach ensures that the model (1) is exposed to sufficient data to comprehend the QAs and (2) does not encounter the same questions during training. The paper is structured as follows:

1. Before diving into the pretrain-finetune process, in Section 3, we first demonstrate that pretraining a model on all biographies *plus* QAs for a p fraction of individuals together enables it to (apply knowledge to) answer questions about the remaining $1-p$ fraction. We call this process *mixed training*. We observe in mixed training, the model *first uses QAs* to encode knowledge about the p fraction, then correlates this encoded knowledge with the biography to infer generalization to the remaining $1-p$ fraction. This learning process deviates from typical human learning and is also less frequently used in practical LLM pretrain.[3]
2. In Section 4, we pretrain the model only on biographies and then finetune it on QAs for a p fraction of individuals. We observe that the model struggles to answer questions about the remaining $1-p$ fraction, *irrespective of model size, pre-train time, or finetune parameters*. However, accuracy significantly improves with *knowledge augmentations* like varying writing styles or sentence shuffling. Even if this augmentation is applied to a subset of individuals, what we call celebrities, test accuracy for others also increases significantly. The mere inclusion of celebrity data in pre-training enhances the model's knowledge extraction for minorities. A key contribution of our study is to **establish this strong link** between knowledge augmentation in pre-training data and the model's improved knowledge extraction capabilities after finetuning.
3. In Section B, **as another main contribution**, we use (nearly) linear probing techniques to show that knowledge augmentation pushes the model to encode a person's knowledge almost linearly in the model's hidden embedding of the person's name tokens. Without augmentation, the model encodes the person's knowledge across all biography words/tokens, making knowledge extraction nearly impossible no matter how one finetunes it. In sum:

   no knowledge augmentation in pretrain data
   ⟺attribute is **not** entirely stored on person's names
   when the model memorizes the pretrain data
   ⟺knowledge cannot be extracted via instruction finetune

   and conversely

   knowledge augmented in pretrain data
   ⟺attribute is **nearly** entirely stored on person's names
   ⟺knowledge can be extracted via instruction finetune

4. In Appendix C, we show that *encoder-only models akin to BERT*, whether mixed-trained or pre-trained and then fine-tuned, cannot extract a person's knowledge after finetuning, regardless of the knowledge augmentation, unless the knowledge is a single word or multiple but independent words (like birth month, day, and year).

[1] One could suggest filtering the data to eliminate such questions and retraining the model. However, this doesn't rule out the presence of similar sentences "Which city did Abraham Lincoln grow up in?", more complex ones in French, or grammatically incorrect versions like "Where Abraham Lincoln birth in?" in the data.

[2] We leave the follow-up question to study *logical reasoning or manipulation* on knowledge to a separate paper (Allen-Zhu & Li, 2023).

[3] For humans, arguably, we first learn from textbooks and then answer exam questions.

Practical Implications. Our controlled study offers key recommendations for LLM training at an industrial scale:

- We emphasize the **importance of pre-training data rewriting (augmentation)**, particularly for rare but critical data. Addressing this during fine-tuning is often too late. Without rewriting, a model may accurately recite knowledge data word by word, but the way it embeds this knowledge into its weights may impede retrieval when prompted differently, resulting in a *total waste of model capacity*.

  Tools such as LLaMA-7B or even *smaller* auxiliary models are adequate for this rewriting task. These "rewrite models" do not need to possess the knowledge themselves. As demonstrated, simple sentence-level shuffling or translations can already enhance performance. Generally, we suggest including prompts that encourage sentence shuffling when using such rewrite models.

  Data rewriting is a form of data augmentation, but also distinct from traditional methods (e.g., dropout, masking, cropping, jittering, flipping) and their associated distillation techniques (like contrastive learning). While traditional augmentations promote the learning of generalizable features over pure memorization, data rewriting — what we call knowledge augmentation — helps language models to memorize knowledge in a more accessible format for downstream tasks. Without such augmentation, the accuracy even for the simplest knowledge extraction task, could be near zero.

- We also demonstrate the advantages of **including more instruction-finetuned data during pre-training**. Our mixed training experiments show that postponing all QA-like data to the fine-tuning phase is suboptimal. Introducing QA-like data earlier in pre-training enables the model to *encode knowledge more effectively*.

Related works. We compare to prior works in Appendix A. At a high level, question answering (QA) is a common method to probe knowledge encoded in language models pretrained on the internet data, and linear probing is a recognized method to examine how a model encodes knowledge, see (Aspillaga et al., 2021; Conneau et al., 2018; Dai et al., 2021; Li et al., 2021; Sun et al., 2023) and many others. **However**, our contribution is that, via *controlled experiments*, we discover that such encoding is only possible when knowledge is augmented on the pre-train level. It is vital to do controlled experiments because for prior works that study knowledge for models pretrained on *internet data*, it's unclear if the model answers QAs by flexibly extracting knowledge from the source or by simply memorizing exact/similar questions from training.

2. Preliminaries

In this paper, we analyze synthetic human biography datasets and near-real datasets generated by LLaMa (Touvron et al., 2023; Zhou et al., 2023). Detailed descriptions are in the appendix, with a brief overview here.

BIO dataset bioS. The synthetic dataset, bioS, generates profiles for $N = 100,000$ individuals.[4] Each individual's details are randomly and *independently* selected from a uniform distribution. The birth dates offer $200 \times 12 \times 28$ possibilities, while other categories offer $100 \sim 1,000$ choices. We also add a "company city" attribute which *depends* on the employer's headquarters location. We ensure uniqueness in each individual's full name.

We generate a six-sentence biographical text entry for each individual, highlighting six distinct aspects. For diversity, each sentence is randomly chosen from approximately 50 distinct templates. In the basic configuration, we generate a single biographical entry for each person, maintaining a consistent order for the six sentences. We use "bioS single" to denote this basic configuration. See an example entry below:

> Anya Briar Forger was born on October 2, 1996. She spent her early years in Princeton, NJ. She received mentorship and guidance from faculty members at Massachusetts Institute of Technology. She completed her education with a focus on Communications. She had a professional role at Meta Platforms. She was employed in Menlo Park, CA.

(2.1)

We also explore 3 types of knowledge augmentations: (1) multi$M$, generating M biography entries for an individual using varied templates, (2) fullname, substituting he/she/they with the person's full name; and (3) permute, shuffling the six sentences randomly. Examples are given in Section 4.2.

BIO dataset bioR. We examine a "close-to-real" dataset produced by LLaMA (Touvron et al., 2023; Zhou et al., 2023). For the set of $N = 100,000$ individuals, we provide an instructive prompt to LLaMA to generate a biographical entry. Here's an example:

> Anya Briar Forger is a renowned social media strategist and community manager. She is currently working as a Marketing Manager at Meta Platforms. She completed her graduation from MIT with a degree in Communications. She was born on 2nd October 1996 in Princeton, NJ and was brought up in the same city. She later moved to Menlo Park in California to be a part of Facebook's team. She is an avid reader and loves traveling.

We diversified our instructive prompts by drawing from a pool of templates and employed rejection sampling to guarantee the inclusion of all six attributes. In the basic configuration, we produce a single biographical entry for each per-

[4] We have a follow-up to push this to $N = 20,000,000$ and similar results hold (Allen-Zhu & Li, 2024).

son (denoted as "bioR single"). For comparison, we also consider multi$M$ augmentation which generates M entries per person and the fullname augmentation. Additional examples can be found in Appendix D.

QA dataset. This paper explores the effectiveness of a trained language model in retaining knowledge from BIO data. As discussed in the introduction, memorization *is more than just predicting the next token* when given exact sentences from BIO. It includes the model's ability to truly **extract knowledge from the BIO**. We assess this knowledge extraction using a question and answer (QA) framework. For each individual, we pose six questions targeting their six unique attributes:

1. What is the birth date of Anya Briar Forger? Answer: October 2, 1996.
2. What is the birth city of Anya Briar Forger? Answer: Princeton, NJ.
3. Which university did Anya Briar Forger study? Answer: Massachusetts Institute of Technology.
4. What major did Anya Briar Forger study? Answer: Communications.
5. Which company did Anya Briar Forger work for? Answer: Meta Platforms.
6. Where did Anya Briar Forger work? Answer: Menlo Park, CA.

For each question, we use it as a prompt for the model to generate a response. QA accuracy is measured by the proportion of answers that exactly match the correct response.[5]

Model architectures. In this ICML version we stick to the GPT2 architecture.[6] The standard GPT2-small architecture comprises 12 layers with 12 heads and 768 dimensions (Radford et al., 2019). Due to GPT2's limitations from its absolute positional embedding, we use its modern *rotary positional embedding* variant (Black et al., 2022; Su et al., 2021), referred to as GPT2 for brevity. We retain the GPT2 small architecture (124M) for pre-training on the bioS data, but use a larger 12-layer, 20-head, 1280-dim GPT (302M) for the bioR data to accommodate its increased complexity. Only in Figure 2 when presenting a negative result, we tried a 12-layer 32-head 2048-dim GPT2 (682M). The default GPT2 tokenizer is used, which converts simple words into single tokens, but names and most other attributes into tokens of varying lengths.

Training. We investigate two types of autoregressive training, detailed in Appendix E.

PRETRAIN + INSTRUCTION FINETUNE. Here, we pretrain the language model ***from scratch*** on the BIO data, randomly sampling and concatenating them into 512-token sentences, separated by a standard `<EOS>` token. The model is then fine-tuned using half of the QA data and evaluated on the remaining half, mirroring the typical instruction finetune process.

MIX TRAINING. In mix training, we train the model ***from scratch*** on all BIO data and half of the QA data. BIO and QA entries are randomly sampled without requiring them to be from the same individual. We use a parameter QA_r to control the QA data amount, primarily setting $\mathsf{QA}_r = 0.8$ (a $2 : 8$ BIO to QA entry ratio). The model's generation accuracy is evaluated using the remaining QA data.[7]

LoRA + full finetune. In full finetuning a pretrained model is tuned for a downstream task such as QAs. LoRA finetuning (Hu et al., 2021) improves upon this by freezing all pretrained model parameters and adding low-rank updates to a subset of the weight matrices for fine-tuning. We apply a low-rank update to the query/value matrices of the transformer model (suggested by (Hu et al., 2021)) and the embedding layer to account for input data distribution shifts. *Full finetuning is also included* when presenting negative results.

3. Mix Training

Mix training involves training the model using BIO data for *all* individuals and QAs for half of them. The group of individuals whose QAs are included in the training set is referred to as *in-distribution* or $\mathcal{P}_{\mathsf{train}}$. The model's generative accuracy is then tested on the QAs from the remaining individuals ($\mathcal{P}_{\mathsf{test}}$) to assess its out-of-distribution generalization capability.

As shown in Figure 1(a), a mix-trained model exhibits strong out-of-distribution generalization, answering most QAs with mean accuracies of 86.6% for bioS and 77.7% for bioR. This indicates that the model can extract and utilize knowledge from the BIO data, addressing queries about an individual's attributes even when no QA about that person was used in training; only their BIO entry was provided. However, our detailed analysis reveals that the model employs a somewhat unconventional method to extract knowledge through mix training.

3.1. Model's Abnormal Learning Behavior

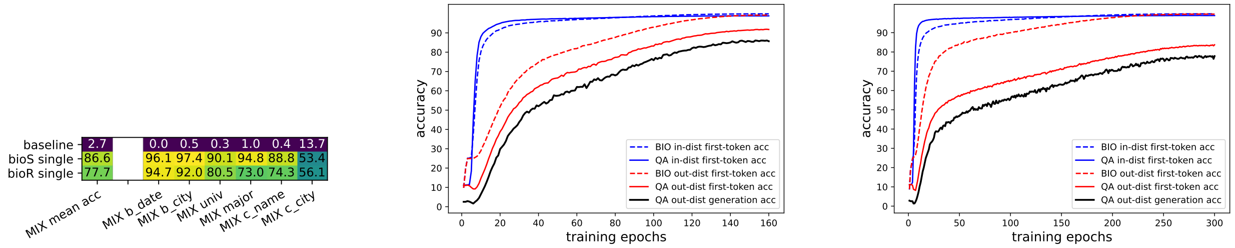
We examine the model's mixed training for knowledge storage and extraction by monitoring its accuracies on the BIO/QA data and for $\mathcal{P}_{\mathsf{train}}/\mathcal{P}_{\mathsf{test}}$ separately. Specifically,[8]

[5] We disregard partial matches or synonyms, emphasizing the model's precision in knowledge extraction.

[6] For others, see our arxiv version at `https://arxiv.org/abs/2309.14316`. Since this paper appeared, Jiang et al. (2024) confirms our results also apply to the pretrained LLaMA-7B model.

[7] See Appendix F for a comparison of how QA_r affects performance. We used beam=4 without sampling throughout this paper; results are almost identical if disabling beam.

[8] Interested readers may consider "whole-attribute" accuracies instead of "first-token" accuracies. They are similar, so we omit them here.

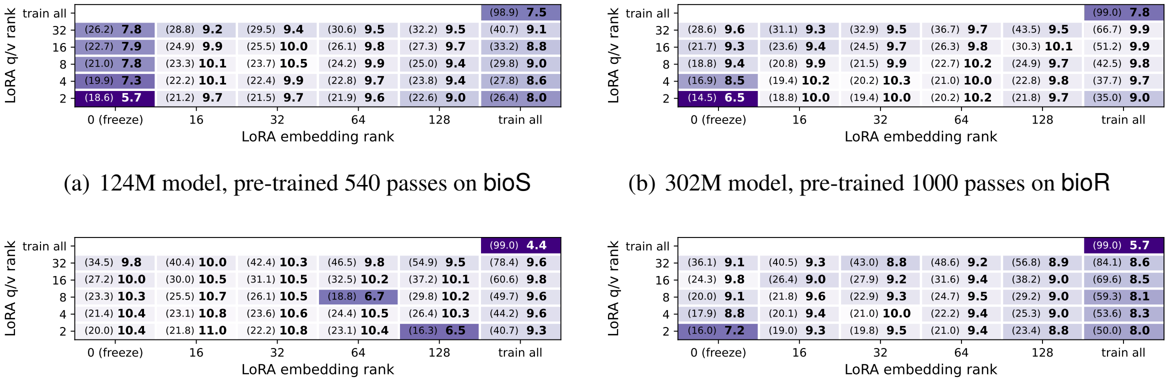


(a) QA out-dist accuracies

(b) training behavior on bioS dataset

(c) training behavior on bioR dataset

Figure 1: Accuracies and loss curves for mix training. b_date,b_city,c_name,c_city stand for birth date, birth city, company name, company city, and mean acc stands for the mean accuracy of the six attributes. Baseline is majority-guessing (c_city has large accuracy because many companies are based in NYC).



(a) 124M model, pre-trained 540 passes on bioS

(b) 302M model, pre-trained 1000 passes on bioR

(c) 682M model, pre-trained 1350 passes on bioS

(d) 682M model, pre-trained 1350 passes on bioR

Figure 2: BIO pretrain + QA finetune (train acc) / **test acc**. Bold number indicates QA generation accuracy on $\mathcal{P}_{\text{test}}$, and the smaller number in bracket represents QA (first-token) accuracy on $\mathcal{P}_{\text{train}}$. For **LoRA fine-tune** we consider a rank $r = 2, 4, 8, 16, 32$ update on the query/value (q/v) matrices and a rank $r' = 0, 16, 32, 64, 128$ update on the word embedding matrix. **Full finetune** is included in the upper-right corners (train all / train all). More details are in Appendix G.

- BIO first-token accuracy: we track the model's next-token-prediction accuracy on the first token of each of the six attributes (birthdate, birthcity, etc.) in the BIO data, separately for $\mathcal{P}_{\text{train}}/\mathcal{P}_{\text{test}}$. This measures the model's BIO data memorization performance. (Despite all individuals' BIO data appearing in training, we still separately track them for $\mathcal{P}_{\text{train}}/\mathcal{P}_{\text{test}}$.)
- QA first-token accuracy: we track the model's next-token-prediction accuracy on the first answer token in the QA data, separately for $\mathcal{P}_{\text{train}}/\mathcal{P}_{\text{test}}$. This loosely estimates the model's QA generation performance.
- QA generation accuracy: we track the model's whole-attribute generation accuracy on $\mathcal{P}_{\text{test}}$.

From Figure 1(b) and 1(c), we find that the model employs an unconventional learning strategy.

- Initially, the model uses the QA data from the training set to encode knowledge for people in $\mathcal{P}_{\text{train}}$, as indicated by the rapid increase in QA in-dist accuracy. This also aids in memorizing in-dist BIO data, as shown by the subsequent rise of the BIO in-dist accuracy.
- The model then gradually aligns the encoded knowledge with the BIO data to learn to extract knowledge and generalize it to $\mathcal{P}_{\text{test}}$. Notably, it takes a while before the BIO out-dist accuracy catches up, followed by an increase in the QA out-dist accuracy.

This is akin to the "study to pass the test" approach in schools, where students prepare using past exam questions and textbooks for answers. While this may yield high scores, it doesn't reflect the natural progression of human knowledge acquisition. **To address this**, we explore a more challenging scenario in the next section where the model is pretrained on the BIO data without exposure to the questions.

Remark 3.1. In mixed training, we selected $\text{QA}_r = 0.8$, maintaining a $8 : 2$ QA to BIO ratio as outlined in Section 2. We found that a higher QA ratio during training improved out-of-distribution QA accuracy (Figure 10 in Appendix F), further supporting our observation of the model's abnormal behavior: it first learns knowledge from QA and then associates it with BIO. For comparison, LLaMA was trained using only 2% of tokens from StackExchange (Touvron et al., 2023).

4. BIO Pretrain + QA Instruction Finetune

We now examine a scenario where the model is pre-trained solely on the BIO data of all individuals. It is then finetuned using QAs from half of these individuals, denoted as $\mathcal{P}_{\mathsf{train}}$, without further use of biographies. The model's generalization is evaluated on questions related to the remaining half, denoted as $\mathcal{P}_{\mathsf{test}}$, whose BIO/QA data were not used during fine-tuning. This process mirrors human knowledge acquisition, where learning from textbooks is applied to later answer exam questions.

4.1. Model May Fail to Extract Knowledge After Pretraining on BIO data

We first pretrain on the basic bioS and bioR datasets, each containing a single biography per person. The QA finetune generalization accuracies (on $\mathcal{P}_{\mathsf{test}}$) are reported in Figure 2, using both full and LoRA finetuning (Hu et al., 2021). The model's QA finetune training accuracy on $\mathcal{P}_{\mathsf{train}}$ is also included for comparison.

Despite a 99+% first-token accuracy during pretraining, the model exhibits near-zero QA accuracy on $\mathcal{P}_{\mathsf{test}}$ for all finetuning parameters. This suggests that while the model can memorize the BIO data token-by-token, it struggles to extract the underlying knowledge. Full-finetuning achieves high *in-distribution* QA finetune accuracy (nearly perfect on $\mathcal{P}_{\mathsf{train}}$), indicating it can memorize the QAs for individuals in the finetuning set. However, it is largely ineffective for QAs concerning individuals in $\mathcal{P}_{\mathsf{test}}$. In sum, we observe:

$$\begin{aligned} & \text{perfect BIO token memorization} \\ & + \text{perfect QA answers for half the people} \\ \not\Longrightarrow\ & \text{correct QA answers for the other half.} \\ & \qquad (\textit{knowledge extraction does not come for free}) \end{aligned}$$

This holds true even when the model size is approximately 7000 times larger than $N = 100k$, the number of individuals, each individual is exposed 1350 times during pretraining, and numerous finetune parameters have been explored.[9] Despite memorizing all knowledge from the BIO data during pretraining, the model encodes it in a disorganized manner within the transformer, preventing knowledge extraction during finetuning.[10]

Figure 2 seems to contradict the success of large models like GPT3.5, trained on internet data such as Common Crawl and known for effective knowledge extraction upon fine-tuning. Why is this? Analyzing the test accuracy breakdown for the six attributes on the bioS data (Figure 3, the "bioS single" row), we see that QA fine-tuning in fact achieves a 33% generalization accuracy on the "birthdate" attribute but fares poorly on others. This is because our bioS single data consistently places birthdate as the first attribute after a person's name, unlike internet data which presents information variably, often repeating it with diverse wordings and orderings. The following subsection on knowledge augmentation supports this hypothesis.

4.2. Knowledge Augmentation

We explore how knowledge augmentation enhances a model's capacity to store and efficiently extract knowledge from training data. We focus on three augmentations: adding multiplicity, introducing permutations, and repeating full names, typically found in internet data. The original datasets without augmentation are referred to as bioS single and bioR single.

- MULTIPLICITY. We denote the method of creating M distinct biography entries for each individual, using varied language but retaining the same information, as multi$M$.[11] An example of adding multiplicity to the biography in (2.1) is:

  Anya Briar Forger came into this world on *October 2, 1996*. She originated from *Princeton, NJ*. She pursued advanced coursework at *Massachusetts Institute of Technology*. She dedicated her studies to *Communications*. She developed her career at *Meta Platforms*. She gained work experience in *Menlo Park, CA*.

  Remark. As a special case, we also experimented with translation (e.g., English to French) to increase sentence diversity, which proved beneficial for the model's knowledge extraction, but we have not included these details in this paper for clarity.

- PERMUTATION. We denote adding random permuta-

[9] In our follow-up work (Allen-Zhu & Li, 2024), we pushes the model size to 1B and N to 20M and confirmed the same holds.

[10] This is not a direct result of catastrophic forgetting, a common issue during heavy fine-tuning where the model forgets the pretraining data. Even with LoRA fine-tuning, which introduces minimal low-rank updates to model weights while preserving the pretrained model, test accuracy only slightly improves.

[11] For bioS data, each of the six sentences is selected from around 50 templates, with a new template resampled for each sentence in the M entries. For bioR data, we recreate the biography using LLaMA for each of the M entries.

Observation. Knowledge augmentation in pretraining data improves model generalization to out-of-distribution QAs after finetuning. Accuracy increases with more augmentations introduced; while mixed training is minimally impacted by knowledge augmentation.

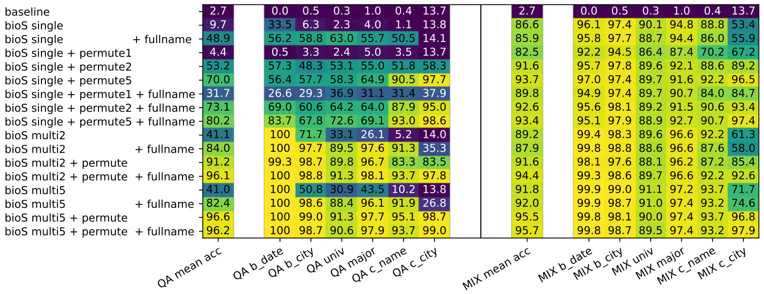


Figure 3: Comparison of BIO pretraining + QA finetuning (left) versus their mixed training counterparts (right) under various knowledge augmentations on the bioS data. Displayed values indicate QA generation accuracies for six attributes in $\mathcal{P}_{\mathsf{test}}$. Refer to Figure 12 for bioR data and Appendix G for more details.

tions to the biography sentences as permute.[12] For instance, the example above can be permuted as follows:

> *Anya Briar Forger* originated from *Princeton, NJ*. She dedicated her studies to *Communications*. She gained work experience in *Menlo Park, CA*. She developed her career at *Meta Platforms*. She came into this world on *October 2, 1996*. She pursued advanced coursework at *Massachusetts Institute of Technology*.

- FULLNAME. We denote the augmentation where all pronouns or partial names in bioS/bioR are replaced with the person's full name as fullname. [13] An example of this augmentation is:

> *Anya Briar Forger* originated from *Princeton, NJ*. Anya Briar Forger dedicated her studies to *Communications*. Anya Briar Forger gained work experience in *Menlo Park, CA*. Anya Briar Forger developed her career at *Meta Platforms*. Anya Briar Forger came into this world on *October 2, 1996*. Anya Briar Forger pursued advanced coursework at *Massachusetts Institute of Technology*.

Results. In Figure 3, we present our results for the bioS dataset. (Parallel results for the bioR dataset are in Figure 12.) We implemented each knowledge augmentation individually and in combinations, then compared the model's QA finetune accuracy on $\mathcal{P}_{\mathsf{test}}$ using LoRA. The model architecture and training parameters remained consistent, but the pre-training datasets varied based on the applied augmentations. Further details are in Appendix G.

We find that adding multiplicity, permutations, or repeating full names all improve the model's ability to memorize the person's information during pretraining, making knowledge extraction easier later.[14] Notably, pretraining on a dataset where each individual has five diverse biography entries (i.e., different wording, different sentence shuffling) boosts the QA fine-tune accuracy (on $\mathcal{P}_{\mathsf{test}}$) from 9.7% to 96.6%. Moreover, such accuracy increases as data multiplicity or permutation number increases, highlighting the model's improved ability to store and extract knowledge when presented with repeated information during pretraining.

One might infer that exposing the model to varied expressions of identical knowledge encourages it to focus on the underlying logical structure of the information, rather than its superficial presentation. This could foster a more direct link between an individual's name and their attributes. We will introduce probing techniques to substantiate this hypothesis in Section B.

4.3. Celebrity Can Help Minority

The previous subsection highlighted the significant benefits of knowledge augmentation. However, in practice, we may not have augmented data for all individuals. This subsection explores whether partially augmenting data can improve knowledge extraction for non-augmented data. In our biography dataset, the augmented subset is akin to a "celebrity" group with plentiful online biographical information, potentially included in the fine-tuning dataset as well. The non-augmented subset is comparable to a "minority" group with limited biographical data.

For comparison, we introduce an additional set of $N = 100,000$ individuals, the celebrity group $\mathcal{P}_{\mathsf{cel}}$, while the original N individuals form the minority group $\mathcal{P}_{\mathsf{min}}$. We test both synthetic bioS and more realistic bioR data. For bioS, the celebrity group's biographies use the

[12] For bioS single, we denote random permutation of the same six sentences P times as permute$P$. For bioS multi$M$, we denote random permutation of each of the M biography entries as permute. The bioR data, generated by LLaMA, already has some randomness in sentence ordering, so no extra permutations are added.

[13] In the synthetic bioS dataset, a person's full name is presented only once, at the start of the initial sentence, with subsequent sentences using solely pronouns. For the LLaMa-generated bioR data, typically, the person's full name appears once at the start; later sentences use either pronouns or parts of the name, such as the first or last name.

[14] An exception is when permutation is directly added to the single data without multiplicity (see "bioS single + permute1"), this hurts the QA performance as it makes knowledge extraction harder.

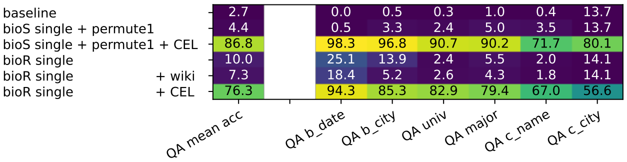


Figure 4: QA finetune accuracy on the *minority group* with vs. without celebrity data in the pretraining process. Experiment details are in Appendix J, where we also include additional experiments in Figure 17.

multi5+permute augmentation, simulating varied expressions found on internet. For bioR, the celebrity group uses the multi5 augmentation, generating their biographies five times using LLaMA.

The language model is pretrained on the combined set $\mathcal{P}_{\mathsf{cel}} \cup \mathcal{P}_{\mathsf{min}}$ biographies and then fine-tuned using QAs from the celebrity group $\mathcal{P}_{\mathsf{cel}}$. We evaluate the model's QA accuracy on the $\mathcal{P}_{\mathsf{min}}$ group.[15] Our results are presented in Figure 4.

Results. In the synthetic bioS case, introducing celebrity data boosts the minority group's QA accuracy from 4.4% to 86.8%. This is significant because:

- the minority group's BIO pretrain data *remains unchanged* in both cases, with $\mathcal{P}_{\mathsf{min}}$ using bioS single+permute1 for biographies, and
- the minority group's QA data *is not used* during finetuning.

This highlights that **simply including celebrity data during pretraining** significantly improves the model's ability to store and extract knowledge from the minority group. Similarly, in the more realistic bioR case, introducing celebrity data also increases the minority group's QA accuracy from 10.0% to 76.3%. We believe this strongly suggests that this phenomenon *also occurs in real-world scenarios*. We will introduce probing techniques to validate the above findings in Section B.

Remark 4.1. Using the bioR dataset, we find the positive impact of celebrity data is *not universal*. Substituting it with the WikiBook dataset improves the model's English comprehension, yet it still struggles with biographical knowledge extraction. This suggests that only celebrity data of *similar form* truly aids knowledge extraction for minority groups. In Figure 17 in Appendix J, we further investigate different celebrity data types and instances of minor format differences between minority and celebrity knowledge.

5. Knowledge Probes on the BIO Pretrained Model

We investigate how a pretrained language model on BIO data encodes knowledge in its hidden representations using two probing techniques: position-based probing (P-probing) and query-based probing (Q-probing). Both techniques employ simple (nearly-linear) probes to extract a person's attributes from the model's hidden representations. Detailed findings are in Appendix B.

In P-probing, we input biography entries into the pretrained model and train a linear classifier on the last hidden layer to predict six target attributes. To accommodate varied data lengths, we identify six *special token positions* preceding the first occurrences of the six attributes in each biography entry. We use the transformer's last hidden layer at these positions to (linearly) predict the six target attributes (Figure 7).[16] Our results (Figure 5) show that *increased knowledge augmentation* in the pretrain data improves *P-probing prediction accuracies from earlier token positions*. In the basic bioS single setup, P-probing accuracy remains low until the token immediately preceding the target attribute. This suggests the model memorizes BIO data but encodes knowledge in a complex manner, revealing a person's attribute **only after encountering all prior attributes**. This **prevents knowledge extraction during QA finetuning**, particularly when only the person's name is given. In Appendix B, we use a Venn diagram to precisely illustrate which attribute is stored after observing another, further confirming this finding.

In Q-probing, we focus on the knowledge directly linked to a person's name. We evaluate input sentences *containing only* the person's full name and train a linear classifier on the last layer's hidden states to predict the person's six attributes.[17] Our results (Figure 6 in Appendix B.2) show that the knowledge-extraction accuracy *is directly linked to* whether the knowledge is (nearly-)linearly stored on the person's name in the pretrained model. This is independent of the finetune parameters, suggesting the model *does not utilize contextual or global information from the biographies to extract knowledge about the individual.*

6. Conclusion

This study explores the ability of pre-trained language models to store and extract knowledge during inference us-

[15] Other fine-tuning variations, such as QA fine-tuning with half of $\mathcal{P}_{\mathsf{min}}$ as training and half as testing, show negligible differences.

[16] For each target attribute prediction task, we freeze the pretrained network but add a trainable rank-2 update on the embedding layer to account for the task change.

[17] We freeze all transformer layers (acquired through pretraining), except the embedding layer, to which we apply a rank-16 update. This adjustment is arguably the minimal change necessary since we are tackling a notably different input distribution.

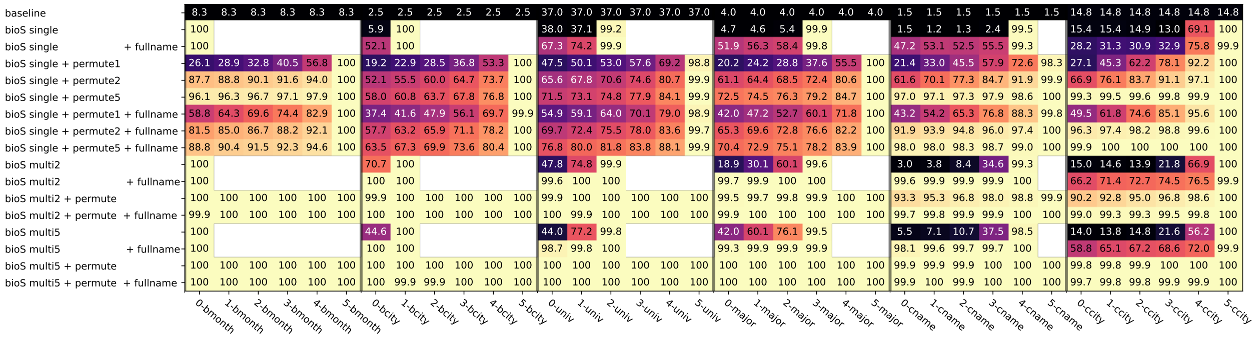


Figure 5: P-probing accuracies for various pretrained models on bioS data. Each **row** represents a pretrained model using a different knowledge augmentation, and each **column** labeled "i-$field$" shows the accuracy of predicting the *first token* of $field$ from position i. Details are in Section B and Appendix H (where we also include experiments for the bioR data and for predicting the full-attribute $field$.)

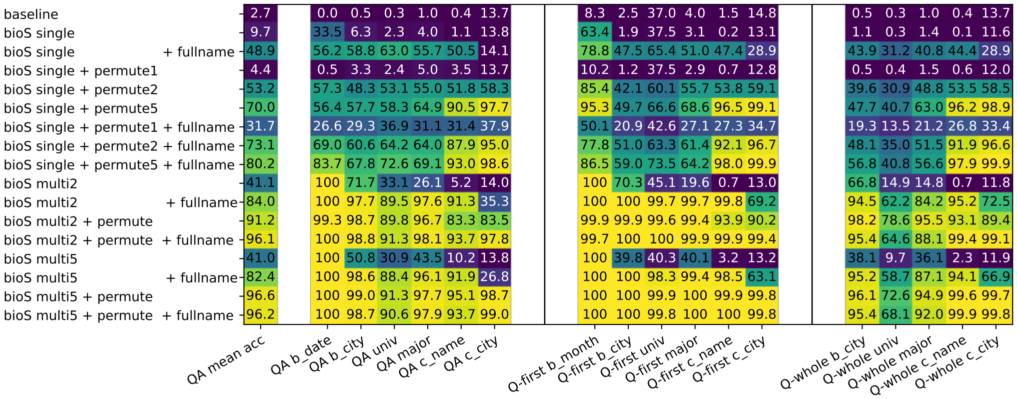


Figure 6: Q-probing accuracies. Each **row** denotes a pretrained model with its specific knowledge augmentation. The *left block* reiterates QA finetune accuracies from Figure 3. The *middle block* showcases Q-probing accuracies on the first-token prediction for the six attributes, and the *right block* focuses on Q-probing for the whole-attribute prediction. (Further details for bioR and more are in Appendix H. Note: For birth date, first token predicts the whole birth month; we do not have whole-attribute prediction for it since it has too many choices.)

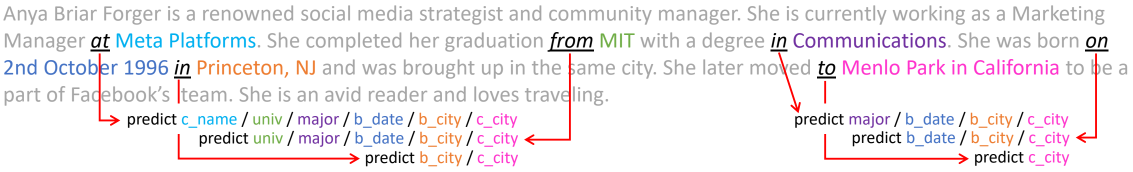

Anya Briar Forger is a renowned social media strategist and community manager. She is currently working as a Marketing Manager at Meta Platforms. She completed her graduation from MIT with a degree in Communications. She was born on 2nd October 1996 in Princeton, NJ and was brought up in the same city. She later moved to Menlo Park in California to be a part of Facebook's team. She is an avid reader and loves traveling.
predict c_name / univ / major / b_date / b_city / c_city
predict univ / major / b_date / b_city / c_city
predict b_city / c_city
predict major / b_date / b_city / c_city
predict b_date / b_city / c_city
predict c_city

Figure 7: Illustration of the P-probing. Underscore prepositions are the *special token positions* where we prob. The task is to predict all attributes following these positions. Given the attribute ordering, there can be up to 6 × 6 = 36 tasks across all data.

ing question-answering tasks. We created a controlled biography dataset and utilized probing techniques to examine the effect of knowledge augmentation on the storage and extractability of knowledge in pre-trained transformers. Synthetic data offers increased control over model training and fine-tuning inputs, which is crucial for understanding the influence of different data sources on the **internal mechanisms** of transformers. This could be a significant future direction for unraveling the complexities of transformers.

Where is the Appendix?

We omit the appendix in this ICML version to encourage readers to consult our full paper, which includes additional results and future editions, at `https://arxiv.org/abs/2309.14316`. An extended video presentation is available at `https://youtu.be/YSHzKmEianc`.

Acknowledgements

We would like to thank Lin Xiao, Chunting Zhou, Tianyi Peng, Xiaodong Liu, and Zhijie Zhou for many helpful conversations. We would like to extend special thanks to Ian Clark, Gourab De, Anmol Mann, and Max Pfeifer from W&B, as well as Nabib Ahmed, Giri Anantharaman, Lucca Bertoncini, Henry Estela, Liao Hu, Caleb Ho, Wil Johnson, Apostolos Kokolis, and Shubho Sengupta from Meta FAIR NextSys; without their invaluable support, the experiments in this paper would not have been possible.

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

References

Allen-Zhu, Z. and Li, Y. Physics of Language Models: Part 3.2, Knowledge Manipulation. *ArXiv e-prints*, abs/2309.14402, September 2023. Full version available at http://arxiv.org/abs/2309.14402.

Allen-Zhu, Z. and Li, Y. Physics of Language Models: Part 3.3, Knowledge Capacity Scaling Laws. *ArXiv e-prints*, abs/2404.05405, April 2024. Full version available at http://arxiv.org/abs/2404.05405.

Anderson, J. R. and Milson, R. Human memory: An adaptive perspective. *Psychological Review*, 96(4):703, 1989.

Aspillaga, C., Mendoza, M., and Soto, A. Inspecting the concept knowledge graph encoded by modern language models. *arXiv preprint arXiv:2105.13471*, 2021.

Baddeley, A. D. *Human memory: Theory and practice*. psychology press, 1997.

Berglund, L., Stickland, A. C., Balesni, M., Kaufmann, M., Tong, M., Korbak, T., Kokotajlo, D., and Evans, O. Taken out of context: On measuring situational awareness in llms. *arXiv preprint arXiv:2309.00667*, 2023.

Black, S., Biderman, S., Hallahan, E., Anthony, Q., Gao, L., Golding, L., He, H., Leahy, C., McDonell, K., Phang, J., Pieler, M., Prashanth, U. S., Purohit, S., Reynolds, L., Tow, J., Wang, B., and Weinbach, S. GPT-NeoX-20B: An open-source autoregressive language model. In *Proceedings of the ACL Workshop on Challenges & Perspectives in Creating Large Language Models*, 2022. URL https://arxiv.org/abs/2204.06745.

Cai, H., Chen, H., Song, Y., Zhang, C., Zhao, X., and Yin, D. Data manipulation: Towards effective instance learning for neural dialogue generation via learning to augment and reweight. *arXiv preprint arXiv:2004.02594*, 2020.

Choi, B., Lee, Y., Kyung, Y., and Kim, E. Albert with knowledge graph encoder utilizing semantic similarity for commonsense question answering. *arXiv preprint arXiv:2211.07065*, 2022.

Conneau, A., Kruszewski, G., Lample, G., Barrault, L., and Baroni, M. What you can cram into a single vector: Probing sentence embeddings for linguistic properties. *arXiv preprint arXiv:1805.01070*, 2018.

Craik, F. I. and Jennings, J. M. Human memory. 1992.

Dai, D., Dong, L., Hao, Y., Sui, Z., Chang, B., and Wei, F. Knowledge neurons in pretrained transformers. *arXiv preprint arXiv:2104.08696*, 2021.

Eldan, R. and Li, Y. Tinystories: How small can language models be and still speak coherent english? *arXiv preprint arXiv:2305.07759*, 2023.

Geva, M., Schuster, R., Berant, J., and Levy, O. Transformer feed-forward layers are key-value memories. *arXiv preprint arXiv:2012.14913*, 2020.

He, P., Liu, X., Gao, J., and Chen, W. Deberta: Decoding-enhanced bert with disentangled attention. *arXiv preprint arXiv:2006.03654*, 2020.

Hernandez, E., Li, B. Z., and Andreas, J. Measuring and manipulating knowledge representations in language models. *arXiv preprint arXiv:2304.00740*, 2023.

Hu, E. J., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W., et al. LoRA: Low-Rank Adaptation of Large Language Models. In *ICLR*, 2021.

Jiang, Z., Sun, Z., Shi, W., Rodriguez, P., Zhou, C., Neubig, G., Lin, X. V., Yih, W.-t., and Iyer, S. Instruction-tuned language models are better knowledge learners. *arXiv preprint arXiv:2402.12847*, 2024.

Kenton, J. D. M.-W. C. and Toutanova, L. K. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, pp. 4171–4186, 2019.

Kobayashi, S. Contextual augmentation: Data augmentation by words with paradigmatic relations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pp. 452–457, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-2072. URL https://aclanthology.org/N18-2072.

Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W.-t., Rocktäschel, T., Riedel, S., and Kiela, D. Retrieval-augmented generation for knowledge-intensive nlp tasks. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 9459–9474. Curran Associates, Inc., 2020. URL `https://proceedings.neurips.cc/paper_files/paper/2020/file/6b493230205f780e1bc26945df7481e5-Paper.pdf`.

Li, B. Z., Nye, M., and Andreas, J. Implicit representations of meaning in neural language models. *arXiv preprint arXiv:2106.00737*, 2021.

Liu, X., Wang, Y., Ji, J., Cheng, H., Zhu, X., Awa, E., He, P., Chen, W., Poon, H., Cao, G., and Gao, J. The microsoft toolkit of multi-task deep neural networks for natural language understanding. *arXiv preprint arXiv:2002.07972*, 2020.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V. Roberta: A robustly optimized bert pretraining approach. *ArXiv e-prints*, abs/1907.11692, July 2019.

Meng, K., Bau, D., Andonian, A., and Belinkov, Y. Locating and editing factual associations in gpt. *Advances in Neural Information Processing Systems*, 35:17359–17372, 2022.

Naseem, T., Ravishankar, S., Mihindukulasooriya, N., Abdelaziz, I., Lee, Y.-S., Kapanipathi, P., Roukos, S., Gliozzo, A., and Gray, A. A semantics-aware transformer model of relation linking for knowledge base question answering. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pp. 256–262, Online, August 2021. Association for Computational Linguistics.

Omar, R., Mangukiya, O., Kalnis, P., and Mansour, E. Chatgpt versus traditional question answering for knowledge graphs: Current status and future directions towards knowledge graph chatbots. *arXiv preprint arXiv:2302.06466*, 2023.

Peng, H., Wang, X., Hu, S., Jin, H., Hou, L., Li, J., Liu, Z., and Liu, Q. Copen: Probing conceptual knowledge in pre-trained language models. *arXiv preprint arXiv:2211.04079*, 2022.

Petroni, F., Rocktäschel, T., Lewis, P., Bakhtin, A., Wu, Y., Miller, A. H., and Riedel, S. Language models as knowledge bases? *arXiv preprint arXiv:1909.01066*, 2019.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I., et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

Richardson, K. and Sabharwal, A. What does my QA model know? devising controlled probes using expert knowledge. *Transactions of the Association for Computational Linguistics*, 8:572–588, 2020. doi: 10.1162/tacl_a_00331. URL `https://aclanthology.org/2020.tacl-1.37`.

Singhal, K., Azizi, S., Tu, T., Mahdavi, S. S., Wei, J., Chung, H. W., Scales, N., Tanwani, A., Cole-Lewis, H., Pfohl, S., et al. Large language models encode clinical knowledge. *arXiv preprint arXiv:2212.13138*, 2022.

Su, J., Lu, Y., Pan, S., Wen, B., and Liu, Y. Roformer: Enhanced transformer with rotary position embedding, 2021.

Sun, K., Xu, Y. E., Zha, H., Liu, Y., and Dong, X. L. Head-to-tail: How knowledgeable are large language models (llm)? aka will llms replace knowledge graphs? *arXiv preprint arXiv:2308.10168*, 2023.

Sushil, M., Suster, S., and Daelemans, W. Are we there yet? exploring clinical domain knowledge of BERT models. In *Proceedings of the 20th Workshop on Biomedical Language Processing*, pp. 41–53, Online, June 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.bionlp-1.5. URL `https://aclanthology.org/2021.bionlp-1.5`.

Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.-A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

Zhou, C., Liu, P., Xu, P., Iyer, S., Sun, J., Mao, Y., Ma, X., Efrat, A., Yu, P., Yu, L., et al. Lima: Less is more for alignment. *arXiv preprint arXiv:2305.11206*, 2023.

Zhu, Y., Kiros, R., Zemel, R., Salakhutdinov, R., Urtasun, R., Torralba, A., and Fidler, S. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, December 2015.

Zlotnik, G. and Vansintjan, A. Memory: An extended definition. *Frontiers in psychology*, 10:2523, 2019. doi: 10.3389/fpsyg.2019.02523.