Is It Smaller Than a Tennis Ball? Language Models Play the Game of Twenty Questions

Maxime De Bruyn, Ehsan Lotfi, Jeska Buhmann, Walter Daelemans
CLiPS Research Center
University of Antwerp, Belgium
maxime.debruyn@uantwerpen.be

Abstract

Researchers often use games to analyze the abilities of Artificial Intelligence models. In this work, we use the game of Twenty Questions to study the world knowledge of language models. Despite its simplicity for humans, this game requires a broad knowledge of the world to answer yes/no questions. We evaluate several language models on this task and find that only the largest model has enough world knowledge to play it well, although it still has difficulties with the shape and size of objects. We also present a new method to improve the knowledge of smaller models by leveraging external information from the web. Finally, we release our dataset and Twentle, a website to interactively test the knowledge of language models by playing Twenty Questions.

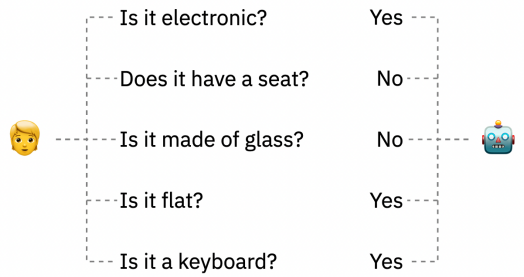

Figure 1: Example Twenty Questions game: a human must discover the hidden entity (a keyboard) by asking yes/no questions to the language model. In this case, the model needs to know about the shape, composition, and purpose of a keyboard to correctly answer all questions. While trivial for humans, our results show that this is not the case for most language models, except for GPT-3, which displays fantastic world knowledge on all questions except size-related questions.

1 Introduction

Generative language models achieve strong performance on multiple NLP tasks by using an unsupervised training objective: predicting the next token in a string of text (Brown et al., 2020; Chowdhery et al., 2022; Zhang et al., 2022).

Despite the simple training objective, these models capture a significant amount of world knowledge (Roberts et al., 2020; Jiang et al., 2020; Talmor et al., 2020). However, we can quickly uncover some limitations by asking simple questions. For example, GPT-3 (Brown et al., 2020) is more likely to complete the following sentence *question: is a kettle smaller than a tennis ball? answer:* ___ with *yes* than *no*. While trivial for a human, GPT-3 has trouble comparing the size of a kettle and a tennis ball.

We can use the *let's think step by step* method to look into the chain of reasoning of GPT-3 (Kojima et al., 2022): *question: is a kettle smaller than a tennis ball? answer: let's think step by step. [...] a tennis ball is about 6 inches in diameter [...] a typical kettle is about 8-10 inches tall and has a diameter of about 4-5 inches. So, a kettle is smaller than a tennis ball.* According to this example, GPT-3 predicts that a tennis ball is twice its actual size, leading to the wrong conclusion that a kettle is smaller than a tennis ball.

In this work, we try to analyze the world knowledge of language models through the game of Twenty Questions. We collected a dataset of 2000+ questions and tried to understand the strength and weaknesses of language models by classifying questions into nine categories of knowledge (usage, size & shape, appearance).

Our results show that GPT-3, a 175 billion parameters language model, can play Twenty Questions thanks to a consistent world knowledge on all categories identified, except for size & shape questions (e.g., *is it bigger than a foot*). Unfortunately, we also show that smaller models do not display the same consistency. However, leveraging the web improved the knowledgeability of T0 by 10% and brought it to a level competitive with GPT-3, despite having 16 times fewer parameters.

Proceedings of the Fifth BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP, pages 80 - 90

Our contributions are the following:

- We release the first dataset consisting of Twenty Questions games.
- We show that very large language models have a consistent world knowledge, while smaller models do not.
- We provide a method to improve the knowledgeability of smaller models using background information from the web.

We publicly release our dataset on HuggingFace (Wolf et al., 2020).[1] We also present *Twentle*, a website to interactively test the world knowledge of language model by playing the game of Twenty Questions.

2 Related Work

Although analyzing the capabilities of language models through the game of Twenty Questions is new, researching the amount of general knowledge and common sense of language models is not.

Unfortunately, the knowledge stored by language models is not symbolic. Therefore, we cannot look into the model and inspect its knowledge. Instead, previous work relied on multiple proxy tasks.

One option is to use regular reading comprehension datasets in a closed-book format. Roberts et al. (2020) follow this approach. They evaluate how much knowledge can be stored inside the weights of a text-to-text T5 model (Raffel et al., 2020). The authors repurposed three reading comprehension datasets to closed-book question answering: Web Questions (Berant et al., 2013), Trivia QA (Joshi et al., 2017) and Natural Questions (Kwiatkowski et al., 2019). They concluded that T5 performs on par with specialized machine comprehension models. GPT-3 (Brown et al., 2020) was also evaluated on the same closed-book question-answering datasets. The largest model (175B parameters) achieved state-of-the-art results on TriviaQA despite not being trained for the task.

Unfortunately, it has been demonstrated later by Lewis et al. (2021) that the datasets used by Roberts et al. (2020) and Brown et al. (2020) suffer from a considerable overlap between the training and test set, invalidating the authors' conclusion based on these datasets. Furthermore, when the overlap between the training and test set is removed, the performance of BART (Lewis et al., 2020a) diminishes from 26.7% to 0.8% on TriviaQA (Joshi et al., 2017), suggesting that the model is unable to generalize to previously unseen questions.

To overcome the previously mentioned overlap problem, Wang et al. (2021) repurposed SQuAD (Rajpurkar et al., 2016), a popular reading comprehension dataset, as a closed-book question answering dataset. They evaluated the performance of BART on this new dataset and concluded that it was still challenging for generative models to perform closed-book question answering.

Another approach is to look at how a language model fills in blanks (i.e., masking). One can estimate what the language model knows by carefully analyzing the model's suggestion. This is the approach followed by Petroni et al. (2019). The authors introduce a new dataset LAMA to test the factual and commonsense knowledge in language models. It provides a set of cloze tasks, e.g., *ravens can* _____ with the associated answer *fly*.

The *oLMpic Games* (Talmor et al., 2020) tests the symbolic reasoning of language models through eight synthetic tasks. While very similar to our work, the dataset uses masking to probe the language model. Mask tokens are only applicable to encoder language models, while we are interested in generative language models.

Previous studies have shown that providing generative language models with background information improves their performance. (Borgeaud et al., 2021; Lewis et al., 2020b; Komeili et al., 2022; De Bruyn et al., 2020; Lazaridou et al., 2022) Similar to Lazaridou et al. (2022), we find that including external knowledge improves the language model's performance, however, we obtain better results by restricting the source of knowledge to Wikipedia instead of the entire Internet.

To summarize, we are the first to analyze the world knowledge of generative language models through the game of Twenty Questions. We depart from the work of Roberts et al. (2020) and Wang et al. (2021) in several ways. First, we only have yes/no answers, which simplifies the evaluation and removes the surface-form problem (Holtzman et al., 2021). Second, using generic questions allows disentangling the understanding of the object and the question.

[1] https://huggingface.co/datasets/maximedb/twentle

	Twenty Questions
Questions	2,832
Generic questions	915
Entities	126
Words (per question)	6.8
Yes	35%
No	65%

Table 1: Summary of the Twenty Questions dataset. We collected 2,832 questions from 126 different entities. We make the distinction between generic and regular questions. Generic questions refer to the entity as "it" (e.g. does it [a rake] have a seat). Generic questions are asked multiple times over different entities (on average 3). We use this unique feature to disentangle the understanding of the question and the entity.

3 Data

This section presents our dataset based on the Twenty Questions game — the first boolean closed-book question answering dataset regarding world and commonsense knowledge. We start this section by introducing the Twenty Questions game. We then explain our data collection process. Finally, we analyze the type of knowledge required to perform well on this dataset.

3.1 Twenty Questions Game

Wikipedia describes Twenty Questions as a spoken parlor game that encourages deductive reasoning and creativity. In the traditional game, one player (the answerer) chooses a subject and does not reveal it. The other players are questioners and must find the hidden entity by asking yes/no questions.

Previous research focused on playing the questioner (Hu et al., 2018; Chen et al., 2018), however, we are interested in the role of the answerer — the player responsible for answering the yes/no questions using his knowledge of the world. According to our research, this is the first attempt at playing the role of the answerer.

3.2 Akinator

Instead of organizing games using Amazon Mechanical Turk, we used Akinator[2] to collect many questions. Akinator is an online game where users can play games of Twenty Questions against a probabilistic model.

Users first pick an entity (without revealing it), and Akinator will then ask yes/no questions to find the hidden entity. It can guess animals, objects, or characters. The player can answer with 5 possible options: *yes*, *no*, *probably yes*, *probably not*, and *don't know*. Although the original Twenty Questions game used a maximum of 20 questions, Akinator will ask questions until it finds the correct entity. We provide examples of questions and entities in Table 2. We were pleasantly surprised by the quality of the Akinator model. It was able to find our hidden entities in most instances. We removed questions from the few instances where it was not capable of finding the correct entity.

3.2.1 Generic Questions

Akinator does not know the entity when asking the question and refers to the entity using *"it"*. Because of its probabilistic nature, Akinator will likely ask the same generic question for multiple entities. We list the most common generic questions in Table 3. For example *is a rake bigger than a foot* and *is a tennis ball bigger than a foot* are two different questions but share the same generic question *is it bigger than a foot*. The average generic question (e.g., *is it bigger than a foot*) is asked for three different entities. However, the distribution is highly skewed, with many specific questions asked only once.

3.2.2 Choice of Entities

We restricted our choice of entities to objects, as we think characters and animals are too culture-dependent to be deemed general knowledge. As much as possible, we tried to choose objects which are not specific to a particular place or culture.

3.2.3 Post-processing

As we are interested in yes/no questions, we remove all questions with *probably yes*, *probably not*, or *don't know* as answer. We use simple regex rules to inject entities into generic questions. We removed all questions about sex or the user's personal experience (e.g., *do you have one at home?*) as these require personal knowledge.

3.3 Knowledge Category

In order to understand the reasoning abilities of the language model, we need to understand the type of knowledge required to answer each question correctly.

After carefully reviewing the questions in our dataset, we classified each question into one of the following nine categories: usage, size & shape,

[2]https://akinator.com/

Generic Question	Entity	Answer
Is it bigger than a foot?	Padlock	No
Does it work with electricity?	Magnifying glass	No
Does it have a seat?	Forklift	Yes
Does it work with the feet?	Lawn mowner	No
Can it be made of wood?	Rake	Yes
Is it mostly for girls?	Belt	No
Does it have a relationship with school?	Wallet	No
Can it be read?	Worldmap	Yes
Is it made of rubber?	Balloon	Yes
Is it bigger than a foot?	Saw	Yes

Table 2: Example questions in our dataset. Akinator does not know the entity when asking the question, and refers to the entity using "it". To avoid any bias toward a specific culture we only used well-known objects as hidden entities. We did not use animals or characters.

Question	Entities
Is it bigger than a foot?	68
Does it go into the mouth?	67
Is it something we wear?	56
Can we buy it?	55
Is it a toy?	50
Is it made of metal?	48
Is it soft?	45
Can it be opened or closed?	42
Is it electronic?	34
Can it be found in a kitchen?	31

Table 3: Most common generic questions in the dataset.

location, composition, description, relatedness, appearance, functioning, and purpose. Finally, we provide an overview with examples in Table 4.

Shape and Size To answer this kind of question, the model should understand an object's shape and be able to compare it with others. For example, *is it bigger than a foot?*

Usage The model should know how an object is used in everyday life to answer these questions. For example, the model should know that a question like *is it something we wear?* applies to a pair of sunglasses, but not a forklift.

Location The model must know in which place or circumstances an object is used. For example, *can we find it in a bathroom* or, *is it outside*.

Composition These questions require knowing the composition of an object. For example, *is it liquid*, or *is it made of glass*.

Description The model should know how humans describe this object with adjectives. For example, *is it heavy*, or *is it sticky*.

Relatedness To answer these questions, the model must be able to relate two categories of objects or concepts together. For example, *does it have a relation with water*, or *is it a toy*.

Functioning These questions require knowing how an object works. This category is broad and includes questions such as *can it be opened or closed*, or *does it work with electricity*.

Appearance This category is related to the description category but focuses on how an object looks. For example, it includes questions such as *does it have a seat*, or *does it have eyes*.

Purpose This kind of question focuses on the purpose of objects. It is related to the usage category but focuses on why we use objects instead of how. It includes questions like *is it useful to sleep*, or *do we use it for travel*.

3.4 Human Agreement

Answering yes/no question is not always straightforward. A single question can be approached in multiple ways. For example, some people answer the question, " *is a DVD smaller than a tennis ball* with yes because the height of a DVD is smaller than that of a tennis ball, while others look at the diameter and answer *no*. We asked four annotators to answer 100 randomly sampled questions. On average, they share the same answer as the one in the dataset 94% of the time. The inter-annotator agreement is good, with a Cohen's Kappa score of 0.76 (Cohen, 1968).

Object Knowledge	Example Question	Percentage
Shape and Size	Is it bigger than a foot? Is it flat?	12.7
Usage	Is it something we wear? Do we use it for a sport?	15.5
Location	Can it be found in houses? Is it outside?	10.9
Composition	Is it liquid? Is it made of glass?	7.8
Description	Is it heavy? Is it sticky?	7.1
Relatedness	Does it have a relation with water? Is it a toy?	14.5
Functioning	Does it work with electricity? Can it be opened or closed?	14.8
Appearance	Does it have eyes? Does it have a seat?	6.9
Purpose	Is it useful to sleep? Do we use it for travel?	7.4

Table 4: We classified each question of the dataset into nine categories depending on the type of knowledge required to answer the question.

4 Language Models

In this section, we review the subjects of this work: generative language models. Language models come in all forms and shapes. However, we focus on two types: encoder-decoder and decoder-only models.

4.1 Encoder-Decoder Models

Encoder-decoder models treat every NLP task as a text-to-text problem using an encoder-decoder Transformer. When this framework is applied to question answering, the model is trained to generate the literal text of the answer in a free-form fashion (Roberts et al., 2020).

T5 is a text-to-text model pre-trained on multiple tasks simultaneously: translation, summarization, classification, reading comprehension, and an unsupervised span corruption task (Raffel et al., 2020). We experiment with the 11 billion parameters version.

T0 further trains T5 on 1700 English datasets (Sanh et al., 2022). The resulting model outperforms GPT-3 (Brown et al., 2020) on several tasks despite being 16x smaller. We use the T0pp version with 11 billion parameters. Conveniently, T0 has already been pre-trained on BoolQ (Clark et al., 2019), a reading comprehension dataset with boolean answers.

4.2 Decoder Models

Decoder models use the decoder part of the original Transformer (Vaswani et al., 2017) model. These models were not trained for a specific task but with an unsupervised objective: predict the next token in a piece of text. Due to their extensive training corpora, these models have already seen many examples of Trivia style questions.

GPT-3 is an auto-regressive language model (Brown et al., 2020). The largest version has 175 billion parameters. The model weights are not publicly available, although the model's predictions are available through a paid API.[3]

GPT-J is a 6 billion parameters autoregressive language model (Wang and Komatsuzaki, 2021) trained on the Pile (Gao et al., 2021).

GPT-Neo-X is a 20 billion parameters autoregressive language model (Black et al., 2022) trained on the Pile (Gao et al., 2021).

OPT is a similar model to GPT-3, but the models' weights were publicly released (Zhang et al., 2022), except for the largest version (175 billion parameters), which is available upon request. Similar to GPT-J, it was trained on the Pile along with data from Reddit. We experiment with the 30 billion parameters version.

5 Experiments

In this section, we report on our experiments using our dataset of Twenty Questions. We experimented with three setups: zero-shot, few-shot, and zero-shot with knowledge augmentation. We use these results in the section to understand the scale of the world knowledge stored by language models.

5.1 Experimental Settings

Our experiments do not require any training, we use language models as-is without fine-tuning. We use the entirety of our dataset for evaluation. We

[3] https://openai.com/api/

Model	Size	F1	Accuracy
Majority	-	0	65.0
GPT-J	6B	48.6	49.0
T5	11B	24.6	68.4
T0	11B	**68.5**	**81.9**
GPT-Neo-X	20B	51.8	34.9
OPT	30B	52.8	38.2
GPT-3	13 B	59.4	60.2
GPT-3	175B	66.4	81.3

Table 5: Result of the zero-shot evaluation. Best performance is achieved by GPT-3 and T0. The other models struggle to reach the majority vote baseline.

Model	Size	F1	Accuracy
Majority	-	0,0	65.0
GPT-J	6B	57.7	57.7
T5	11B	0.0	65.8
T0	11B	6.7	65.8
GPT-Neo-X	20B	58.4	58.3
OPT	30B	60.4	71.6
GPT-3	13B	58.2	60.2
GPT-3	175B	**83.0**	**87.9**

Table 6: Result of the few-shot evaluation. GPT-3's F1 improves by 9% to reach 83%. The performance of OPT barely improves compared to the zero-shot reasoning, while as expected the performance of encoder-decoder models plummets.

measure the probability of the *yes* answer by summing the probability of the *yes*, *Yes*, *true*, and *True* tokens. The same is done for the *no* answer with *no*, *No*, *false* and *False*. Our dataset contains 65% of *no* answers, we use F1 (binary) as primary evaluation metric and also report accuracy.

5.2 Zero-shot

In the zero-shot setting, models answer the question with only a textual description of the task. We expect T5 and T0 to perform well in this setup as they were pre-trained using the same setup, while this is not the case for decoder-only models.

Prompt We use the same prompt for both encoder-decoders and decoder-only models.

```
You are playing a game of 20 questions.
Answer the following question with yes or no.
Question: {{ question }}
Answer:
```

Results We report the results of our zero-shot experiment in Table 5. As expected, T0 achieves the best results with an F1 of 68.5% and an accuracy of 81.9%. GPT-3 also performs nicely in this setup, with 16x more parameters than T0. However, all the other models show an accuracy lower than the majority vote baseline.

5.3 Few-shot

In the few-shot setup, models receive identical instructions as in the zero-shot setup, in addition to a few examples. This setup benefits decoder-only models as they can now learn the task on the fly using in-context learning (Beltagy et al., 2022).

Prompt We augment the zero-shot prompt with four examples. There are two examples with *yes* and two with *no*. We randomly select examples from different entities and generic questions.[4]

```
You are playing a game of 20 questions.
Answer the following question with yes or no.
Question: {{ question_example_1 }}
Answer: {{ answer_example_1 }}
...
Question: {{ question_example_n }}
Answer: {{ answer_example_n }}
Question: {{ question }}
Answer:
```

Results We provide an overview of the few-shots results in Table 6. As expected, the performance of decoder-only models increases, while the performance of encoder-decoder decreases[5]. For example, GPT-3's F1 increased from 66.4% to a record 83.0%. Unfortunately, these results also show that (relatively) smaller decoder-only models do not reach T0's performance in a zero-shot setup.

5.4 Zero-shot with Knowledge Augmentation

The performance of GPT-3 is exceptional. However, it comes at a steep computational and environmental cost. Moreover, as T0 has fewer parameters than GPT-3, it has less "space" to store world knowledge. In this section, we try to augment T0 with external knowledge to help it bridge the performance gap with GPT-3. We use two sources of background knowledge: the entire Internet using Bing search and the Wikipedia page of the entity.

Prompt We follow the same prompt as in the zero-shot analysis. In addition, we augment it with a space for background knowledge.

[4]This setup is similar to the start of a Twenty Questions game where the model does not have previous examples for the same entity.

[5]These models were zero-shot inference, not few-shot.

Model	Size	F1	Accuracy
T0 (ZS)	11B	68.5	81.9
T0 (Bing)	11B	69.7	75.7
T0 (Wiki)	11B	**79.3**	**86.0**
GPT-3 (FS)	175B	**83.0**	**87.9**

Table 7: Augmenting T0 with background information improves its F1 score by 10% and brings it to a competitive level with GPT-3.

```
Text: {{ background_knowledge }}
You are playing a game of 20 questions.
Answer the following question with yes or no.
Question: {{ question }}
Answer:
```

Bing We run a bing search for every question and only keep the text snippet returned by Bing. We compare each text snippet to the question using a cross-encoder from Sentence Transformers (Reimers and Gurevych, 2019). We then keep the snippet with the highest score. We do not restrict Bing, so it can also choose to return pages from Wikipedia.

Wikipedia We chunk the Wikipedia page of each entity into passages of around 256 tokens. Then, we re-rank the passages using the same cross-encoder.

Results We provide an overview of the few-shots results in Table 7. The Bing search results are disappointing. The F1 score barely improves by 1%. On the other hand, the Wikipedia search results are outstanding: F1 improves by over 10% and accuracy by 4%.

This section concludes that GPT-3 (few-shot) is the best model for playing the answerer in a game of Twenty Questions. However, GPT-3 is computationally and environmentally costly. We showed that incorporating background knowledge from Wikipedia can improve T0's performance to a competitive level with GPT-3 despite having 16 times fewer parameters.

6 World Knowledge Analysis

We now use the results of the previous section to analyze the world knowledge of the three best models: GPT-3, T0, and T0 Knowledge Grounded (KG).

6.1 Knowledge Category

We list the accuracy by category of knowledge in Table 8. The most striking result is the low performance of the three models in the Shape & Size category. For example, GPT-3 has a difference of 20% between the worst category (Shape & Size) and the second-worst category (Usage).

Knowledge Type	GPT-3	T0	T0-KG	OPT
Shape & Size	66	56	69	60
Usage	86	82	86	75
Location	88	74	89	60
Composition	90	78	78	69
Description	81	69	73	65
Relatedness	95	94	88	79
Functioning	87	79	74	71
Appearance	91	83	83	89
Purpose	91	88	82	75

Table 8: Accuracy (%) by category of knowledge. GPT-3 outperforms T0 on every knowledge type. Shape & Size questions stand out as a weak spot for GPT-3 and T0.

On the other hand, GPT-3 and T0 can answer questions relating to two objects or concepts exceptionally well (e.g., *is it related to water* or *is it a toy*). Intriguingly, incorporating knowledge into the prompt diminishes the score on relatedness for T0-KG.

We now dig deeper into *size & shape* questions and try to understand if there are specific kinds of questions mishandled by the language models. We list the average accuracy by questions in the Shape & Size category in Table 9. We notice that questions 1, 3 & 4 are not specific enough. On which dimension should we compare the size of the tennis ball? [6] The inter-annotator score on Shape & Size question is 0.75, almost equivalent to the global inter-annotator score of 0.76. We believe humans have enough common sense to decide on which dimension to evaluate the size of objects.

6.2 Entities

Inspired by previous research (Razeghi et al., 2022), we look for a correlation between the average accuracy of an entity and its frequency in the pretraining data.[7] We do not find any significant correlation, except a small 0.05 correlation for T0. We believe the conclusion would be different with lesser-known objects.

We notice that ambiguous entities such as *a rule*[8]

[6] Is a DVD smaller than a tennis ball because of its thickness?

[7] We use the first 10 billion tokens of the C4 dataset (Raffel et al., 2020) to estimate the frequency of entities in the pretraining data.

[8] As in a 30 cm rule/ruler

Question	GPT-3	T0	T0-KG
Is it smaller than a tennis ball?	50	55	60
Is it globe-shaped?	55	77	77
Is it bigger than a foot?	60	47	67
Can we transport it in a pocket?	62	50	50
Is it flat?	66	55	61
Is it round?	68	43	69
Is it long?	71	28	57
Is it rectangular?	72	81	72
Is it taller than a man?	78	78	71
Does it have a square shape?	80	80	100
Is it pointed?	85	71	71
Is it bigger than a bus?	100	100	100

Table 9: Accuracy (%) of GPT-3, T0, and T0-KG on Shape & Size questions. GPT-3 struggles with comparing the size of entities with the size of a tennis ball.

and *a racket*[9] are not well managed by all models for understandable reasons.

6.3 Knowledge Augmentation

In this section, we try to understand why Wikipedia is a much better source of background knowledge than Bing's search over the Internet.

Knowledge Source We manually reviewed and compared the background knowledge provided by Bing and Wikipedia. We found that the knowledge returned by Bing can be specific, whereas the game of Twenty Questions requires general knowledge. For example, when asked *does a printer have a seat*, the obvious answer is no. However, Bing returns a text saying *[...] each used printer takes one license seat. [...]* confusing the model into thinking printers do have seats. Another example is the question *is a litter box a weapon*. The correct answer is no. Bing, however, returns a text saying *[...] cat litter box used as a weapon in fight over prescription drugs [...]* confusing the model into thinking a litter box is a weapon. In both instances, the knowledge returned by Wikipedia is the introductory paragraph describing the entity.

Knowledge Category According to Table 8, incorporating background knowledge helps in Location (+15%) and Usage (+13%) questions. On the other hand, it hurts performance on Relatedness questions (-6%).

This section concludes that GPT-3 performs consistently on all categories of questions, except Shape and Size. Although competitive, T0 does not show the same consistency as GPT-3, even when augmented with background information.

[9] As in a tennis racket

7 Twentle

We present an interactive website to let anyone test the world knowledge of T0-KG by playing the game of Twenty Questions. Inspired by Wordle, we named our website Twentle, available at twentle.com.

8 Future Work

Reducing the world to yes/no questions is not an easy task. Our human agreement section demonstrates that humans do not agree on all answers. Future work is needed to compare the agreement of humans and language models by category of question. In this study, we limited ourselves to the study of the answerer. However, GPT-3 could potentially also play the role of the questioner. Future work is needed to study the knowledgeability of language models on lesser-known objects. In this case, we anticipate that large models will also need to leverage the web for information.

9 Conclusion

In this work, we analyzed the world knowledge of language models through the game of Twenty Questions. Our analysis reveals that most language models do not have the world knowledge required to play this game. GPT-3 is a notable exception. It displays impressive world knowledge on all categories of questions identified, except for shape & size questions — *is it smaller than a tennis ball*. Furthermore, we showed how grounding smaller models on information from the web improves their knowledgeability. Through this work, we demonstrated the need for more clarity on which model architecture and pre-training method best captures world knowledge.

10 Limitations

We intentionally limited our analysis to well-known objects. We anticipate a lower performance on lesser-known objects. Furthermore, our work uses well-defined questions with little noise, whereas real-world questions by humans could be more challenging for language models to understand. The dataset we collected could contain biases already present in our society. Unfortunately, the same is true for the answers given by the language model.

Acknowledgement

We thank the reviewers for their helpful feedback. This research received funding from the Flemish Government under the *Onderzoeksprogramma Artificiële Intelligentie (AI) Vlaanderen* programme.

References

Iz Beltagy, Arman Cohan, Robert Logan IV, Sewon Min, and Sameer Singh. 2022. Zero- and few-shot NLP with pretrained language models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: Tutorial Abstracts*, pages 32–37, Dublin, Ireland. Association for Computational Linguistics.

Jonathan Berant, Andrew Chou, Roy Frostig, and Percy Liang. 2013. Semantic parsing on Freebase from question-answer pairs. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1533–1544, Seattle, Washington, USA. Association for Computational Linguistics.

Sidney Black, Stella Biderman, Eric Hallahan, Quentin Anthony, Leo Gao, Laurence Golding, Horace He, Connor Leahy, Kyle McDonell, Jason Phang, Michael Pieler, Usvsn Sai Prashanth, Shivanshu Purohit, Laria Reynolds, Jonathan Tow, Ben Wang, and Samuel Weinbach. 2022. GPT-NeoX-20B: An open-source autoregressive language model. In *Proceedings of BigScience Episode #5 – Workshop on Challenges & Perspectives in Creating Large Language Models*, pages 95–136, virtual+Dublin. Association for Computational Linguistics.

Sebastian Borgeaud, Arthur Mensch, Jordan Hoffmann, Trevor Cai, Eliza Rutherford, Katie Millican, George van den Driessche, Jean-Baptiste Lespiau, Bogdan Damoc, Aidan Clark, Diego de Las Casas, Aurelia Guy, Jacob Menick, Roman Ring, Tom Hennigan, Saffron Huang, Loren Maggiore, Chris Jones, Albin Cassirer, Andy Brock, Michela Paganini, Geoffrey Irving, Oriol Vinyals, Simon Osindero, Karen Simonyan, Jack W. Rae, Erich Elsen, and Laurent Sifre. 2021. Improving language models by retrieving from trillions of tokens. *CoRR*, abs/2112.04426.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Yihong Chen, Bei Chen, Xuguang Duan, Jian-Guang Lou, Yue Wang, Wenwu Zhu, and Yong Cao. 2018. Learning-to-ask: Knowledge acquisition via 20 questions. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery, Data Mining*, KDD '18, page 1216–1225, New York, NY, USA. Association for Computing Machinery.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayana Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. 2022. Palm: Scaling language modeling with pathways.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. 2019. BoolQ: Exploring the surprising difficulty of natural yes/no questions. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2924–2936, Minneapolis, Minnesota. Association for Computational Linguistics.

J. Cohen. 1968. Weighted kappa: nominal scale agreement with provision for scaled disagreement or partial credit. *Psychological bulletin*, 70 4:213–20.

Maxime De Bruyn, Ehsan Lotfi, Jeska Buhmann, and Walter Daelemans. 2020. Bart for knowledge grounded conversations. In *Converse@ KDD*.

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.

Ari Holtzman, Peter West, Vered Shwartz, Yejin Choi, and Luke Zettlemoyer. 2021. Surface form competition: Why the highest probability answer isn't always right. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7038–7051, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Huang Hu, Xianchao Wu, Bingfeng Luo, Chongyang Tao, Can Xu, Wei Wu, and Zhan Chen. 2018. Playing 20 question game with policy-based reinforcement learning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3233–3242, Brussels, Belgium. Association for Computational Linguistics.

Zhengbao Jiang, Frank F. Xu, Jun Araki, and Graham Neubig. 2020. How can we know what language models know? *Transactions of the Association for Computational Linguistics*, 8:423–438.

Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. 2017. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1601–1611, Vancouver, Canada. Association for Computational Linguistics.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners.

Mojtaba Komeili, Kurt Shuster, and Jason Weston. 2022. Internet-augmented dialogue generation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8460–8478, Dublin, Ireland. Association for Computational Linguistics.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466.

Angeliki Lazaridou, Elena Gribovskaya, Wojciech Stokowiec, and Nikolai Grigorev. 2022. Internet-augmented language models through few-shot prompting for open-domain question answering.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020a. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020b. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS'20, Red Hook, NY, USA. Curran Associates Inc.

Patrick Lewis, Pontus Stenetorp, and Sebastian Riedel. 2021. Question and answer test-train overlap in open-domain question answering datasets. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1000–1008, Online. Association for Computational Linguistics.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. 2019. Language models as knowledge bases? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2463–2473, Hong Kong, China. Association for Computational Linguistics.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Yasaman Razeghi, Robert L. Logan, Matt Gardner, and Sameer Singh. 2022. Impact of pretraining term frequencies on few-shot reasoning.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Adam Roberts, Colin Raffel, and Noam Shazeer. 2020. How much knowledge can you pack into the parameters of a language model? In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5418–5426, Online. Association for Computational Linguistics.

Victor Sanh, Albert Webson, Colin Raffel, Stephen Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Teven Le Scao, Arun Raja, et al. 2022. Multitask prompted training enables zero-shot task generalization. In *The Tenth International Conference on Learning Representations*.

Alon Talmor, Yanai Elazar, Yoav Goldberg, and Jonathan Berant. 2020. oLMpics-on what language model pre-training captures. *Transactions of the Association for Computational Linguistics*, 8:743–758.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Ben Wang and Aran Komatsuzaki. 2021. GPT-J-6B: A 6 Billion Parameter Autoregressive Language Model. https://github.com/kingoflolz/mesh-transformer-jax.

Cunxiang Wang, Pai Liu, and Yue Zhang. 2021. Can generative pre-trained language models serve as knowledge bases for closed-book QA? In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3241–3251, Online. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, Todor Mihaylov, Myle Ott, Sam Shleifer, Kurt Shuster, Daniel Simig, Punit Singh Koura, Anjali Sridhar, Tianlu Wang, and Luke Zettlemoyer. 2022. Opt: Open pre-trained transformer language models.

A Computing Infrastructure

We ran all our experiments on a server running 8 NVIDIA GPU (12GB) with 128GB of RAM and 24 CPU. All models ran in parallel using the device_map argument of the from_pretained method.

B Hyperparameter Search

We did not engage in a hyperparameter search. Future research could look for the optimal prompt, balance of yes and no examples.

C Correlation With Token Frequency

We display the correlation between the average accuracy of an entity and its relative frequency in the pre-training data in Table 10.

Model	Correlation	P-value
GPT-3	-0.02	0.35
T0	0.05	0.01
T0-KG	-0.01	0.45

Table 10: Spearman correlation of the average accuracy of an entity with its frequency in the pre-training data.