

Do LLMs Dream of Ontologies?

MARCO BOMBIERI and PAOLO FIORINI, University of Verona, Verona, Italy
SIMONE PAOLO PONZETTO, University of Mannheim, Mannheim, Germany
MARCO ROSPOCHER, University of Verona, Verona, Italy

Large Language Models (LLMs) have demonstrated remarkable performance across diverse natural language processing tasks, yet their ability to memorize structured knowledge remains underexplored. In this article, we investigate the extent to which general-purpose pre-trained LLMs retain and correctly reproduce concept Identifier (ID)–label associations from publicly available ontologies. We conduct a systematic evaluation across multiple ontological resources, including the Gene Ontology, Uberon, Wikidata, and ICD-10, using LLMs such as PYTHIA-12B, GEMINI-1.5-FLASH, GPT-3.5, and GPT-4. Our findings reveal that only a small fraction of ontological concepts is accurately memorized, with GPT-4 demonstrating the highest performance. To understand why certain concepts are memorized more effectively than others, we analyze the relationship between memorization accuracy and concept popularity on the Web. Our results indicate a strong correlation between the frequency of a concept’s occurrence online and the likelihood of accurately retrieving its ID from the label. This suggests that LLMs primarily acquire such knowledge through indirect textual exposure rather than directly from structured ontological resources. Furthermore, we introduce new metrics to quantify prediction invariance, demonstrating that the stability of model responses across variations in prompt language and temperature settings can serve as a proxy for estimating memorization robustness.

CCS Concepts: • **Computing methodologies** → **Natural language generation**; • **General and reference** → **Experimentation**; **Evaluation**;

Additional Key Words and Phrases: Large Language Models, Memorization, Ontologies

ACM Reference format:

Marco Bombieri, Paolo Fiorini, Simone Paolo Ponzetto, and Marco Rospocher. 2026. Do LLMs Dream of Ontologies? *ACM Trans. Intell. Syst. Technol.* 17, 6, Article 130 (December 2026), 23 pages.
<https://doi.org/10.1145/3725852>

This research has received funding from the European Research Council (ERC) under the European Union’s Horizon 2020 research and innovation programme (grant agreement No. 742671, “ARS”), the University of Mannheim’s “Gastwissenschaftler*innenprogramm Nachhaltigkeit,” and the MUR funded 2023–2027 Project of Excellence “Inclusive Humanities: Perspectives for Development in the Research and Teaching of Foreign Languages and Literatures” of the Department of Foreign Languages and Literatures of the University of Verona. Part of this work was conducted within the Digital Arena for Inclusive Humanities (DAIH) Research Centre at the University of Verona.

Authors’ Contact Information: Marco Bombieri (corresponding author), University of Verona, Verona, Italy; e-mail: marco.bombieri_01@univr.it; Paolo Fiorini, University of Verona, Verona, Italy; e-mail: paolo.fiorini@univr.it; Simone Paolo Ponzetto, University of Mannheim, Mannheim, Germany; e-mail: ponzetto@uni-mannheim.de; Marco Rospocher, University of Verona, Verona, Italy; e-mail: marco.rospocher@univr.it.



This work is licensed under Creative Commons Attribution International 4.0.

© 2026 Copyright held by the owner/author(s).

ACM 2157-6912/2026/12-ART130

<https://doi.org/10.1145/3725852>

1 Introduction

Large Language Models (LLMs) have revolutionized how natural language is computationally processed. Pre-trained models, such as GPT-3 [7], LLaMA [47], and BLOOM [4], have shown that human-level performance can be achieved on various tasks (e.g., those found in MMLU [21] or ChatBot Arena [11]) with no or minimal task-specific adaptation.

Recent works [8–10, 34, 41, 51, 52] have investigated memorization aspects related to these models, i.e., to what extent can the content used to train an LLM be extracted as-is through appropriate interaction with the model. These works have mainly focused on extracting the actual sequences of tokens used to train the model. Here, we investigate instead a different aspect of memorization, namely to what extent general-purpose pre-trained LLMs can memorize *information*, meaning the ability of an LLM to recall the association between two tokens that may not occur consecutively in the text, following an approach similar to [52].¹ To this end, we assess whether and to what extent some popular pre-trained general-purpose LLMs have memorized information from known ontologies, focusing on a basic and fundamental piece of information, namely the association of concept **Identifiers (IDs)** with their corresponding natural language labels (e.g., GO:0001822 with *kidney development*).

We first evaluate how much LLMs such as GPT-3.5, GPT-4, GEMINI-1.5-FLASH, and PYTHIA-12B [3] have acquired information about concepts from the text and can retrieve it when prompted without further training. Our experiments, conducted on both domain-specific (Gene [1], Uberon [38], and ICD-10 [40]) and general-purpose (**Wikidata (WD)** [48]) ontologies, demonstrate that, while LLMs show awareness of each of the ontologies we consider, they have not memorized completely and consistently all concepts. Consequently, we next investigate which ontological concepts have been better memorized and why. Experimental findings indicate that the correct memorization of concepts varies among them while correlating with the frequency of their occurrence on the Web, thus suggesting that the information was not acquired (or at least not exclusively) by processing the ontology itself but rather from other textual Web materials mentioning it. Indeed, our experiments advocate that the more information on some ontology concepts is present in textual form on the Web, which is probably largely used as training material for LLMs, the better they memorize it. Our findings may loosely resemble some theories about how human memory functions, such as the *law of repetition*: Indeed, while for humans, “the speed and accuracy of remembering improve with repeated opportunities to study and retrieve the to-be-remembered material” [27], it seems that also for LLMs, memorizing information about some ontology appears to depend on the number of times that information has been seen in the training material. We finally focus on quantifying the extent of memorization observed in previous experiments. For this, we ask for the very same information multiple times to the LLM, using different prompt repetitions, setting different temperature levels to control the model’s output, and using different prompting languages. The experiments show that the more the information is popular on the Web and thus likely to be more often seen in the training material, the more the output of the model is consistent and accurate independently of the prompt language and temperature level used, thus suggesting that the invariance of the model output can be used as evidence of correct information memorization. To foster further research on these aspects, we make all code and data publicly available.²

¹We emphasize that this article focuses on *memorization* rather than *learning*, acknowledging that the distinction between the two is complex and widely debated: While learning in some cases involves some degree of memorization, they remain distinct concepts [6, 14].

²<https://github.com/marcobombieri/do-LLM-dream-of-ontologies>.

2 Related Work

Several recent works have studied the memorization of different kinds of information in language models for various purposes. Ranaldi et al. [41] investigate the interplay between memorization and performance in downstream tasks with BERT. By defining a measure for evaluating memorization from pre-training, they establish a correlation between highly memorized examples and improved classification. Chang et al. [10] explore memorization in ChatGPT and GPT-4 through a “data archaeology” approach. By employing a name cloze membership inference query, they show that these models have memorized a diverse range of copyrighted materials, with the extent of memorization linked to the prevalence of passages on the Web. Carlini et al. [8] delve into the undesirable consequences of memorization in LLMs: They demonstrate that LLMs, when prompted appropriately, emit verbatim memorized training data, leading to privacy violations, degraded utility, and fairness issues, a problem already identified for clinical notes in BERT [32] and personal information in generative LLMs [23, 36] and generally in generative neural networks [9]. In response to privacy and security concerns surrounding these models, some authors initially developed methods to decrease the total quantity of memorized text [8, 28, 31]. These approaches were later reconsidered because, alongside “bad” memorization, there is also “good” memorization. The latter involves accurately recalling factual events and details, helping to avoid generating plausible but incorrect information or “hallucinations” [2], such as the case of our article. In the same direction, a very recent work [43] presented a method to estimate the degree of dataset contamination and copyrighted books in LLMs, based on the hypothesis that an example unseen during training given in input to the detection method tends to contain a few outlier words with low probabilities, whereas a seen example is less likely to contain words with such low probabilities. Liu et al. [34] investigate the same problem by examining next-word-prediction-based language models with membership adversarial attacks to determine whether a given text was included in pre-training data. Zhou et al. [51] addressed the same issue using a reference model and a classifier. To detect whether a text was used during pre-training, they first memorize it with a reference model. The prefix of the text is then fed into a LLM to generate a continuation. Both the original text and the generated text are input into the reference model to extract language modeling probabilities. If the text was part of the LLM’s pre-training, the generated continuation closely matches the original, leading to high probabilities. A classifier then uses these features to determine if the text was used in pre-training.

All the above papers mainly focus on *verbatim* detection in LLMs. In contrast, Zhou et al. [52] explored the concept of entity memorization, noting that verbatim detection methods might miss certain memorized content when key or sensitive information is embedded only in specific parts of the data. Specifically, they define entity memorization as occurring when a model is prompted with input derived from partial entities within the training data, and the model’s output correctly includes the expected entity information. In this work, we complement their study of memorization using a broad definition of entity with a more classic, ontology-centric view of entities as concepts from the vocabulary of a reference ontology.

Recently, Ishihara [25] conducts a comprehensive survey on training data extraction from **Pre-Trained Language Models (PLMs)**, providing a taxonomy of memorization definitions and systemizing approaches for both attack and defense. In our work, we follow this line of research while focusing on *memorization of information* rather than textual tokens, entities, and sequences. He et al. [20] proposed to investigate PLMs’ knowledge of ontologies using a set of inference-based probing tasks, thus showing that these models encode little subsumption relations. Sun et al. [45] also demonstrated that LLMs still struggle with accurately grasping factual knowledge, as revealed by their analysis of LLMs’ memorization of certain knowledge-graph relationships. For this reason,

Boer et al. [5] proposed to combine triplets-based searches in a semi-structured knowledge base with the generative capacities of LLMs to improve their answers. Finally, Wu et al. [49] tested BERT- and RoBERTa-based language models' ability to memorize and reason with ontological knowledge, finding that they struggle on these tasks. A common challenge observed in these tasks is the tendency of LLMs to generate hallucinations, i.e., statements that are plausible and contextually coherent but factually incorrect [50]. This issue has been widely studied, both proposing detection methods and evaluation metrics with a rapidly growing body of literature [12, 39, 53]. Our work also contributes to this field by assessing LLMs' hallucination tendencies when interacting with ontologies. In this work, we focus primarily on the terminology component and its provenance with respect to the training data. The importance of our work stems from the fact that PLMs are becoming ever more critical for ontology-centric tasks like ontology mapping and alignment [22, 37], whose evaluation may also suffer from dataset leakage [13, 42].

3 Do LLMs Dream of Ontologies?

3.1 Task Description

To evaluate the memorization of ontological information in LLMs, we propose a simple task inspired by that of *exact memorization* [46]. In particular, we ask the LLMs to return in a zero-shot fashion the ID of an ontological concept, given in input, through a natural language prompt, the concept's label. No further training on domain data is thus performed, and no information is inserted into the LLMs context other than that in the prompt.

We remark that the ID-concept label association is very basic information contained in (many) ontologies, which typically include more complex and structured knowledge—e.g., concept taxonomies, concept relations, and axioms [20]: However, IDs and concept labels are (i) inherently sequences of characters and (ii) frequently mentioned together in textual content (e.g., Web pages, scientific articles), thus making the ID-concept label association something likely to be observed in the training material of LLMs and thus helpful in evaluating memorization in LLMs.

3.2 Resources

Ontologies. We opt for ontologies whose entities³ are uniquely identified with an ID syntactically unrelated to the entity's label to follow a similar approach to that of *exact memorization* analysis. Doing so guarantees that if the LLM associates the correct ID to an input label, it is because the ID and the corresponding label have been encountered in the training material and thus have been memorized, as it is impossible to derive the ID from the label only. We focus on three domain ontologies (**Gene Ontology (GO)** [1], **Uberon Ontology (UO)** [38], and **ICD-10** [40]) because domain-specific terms tend to have a lower polysemy [30] and thus allow us to ignore issues related to label disambiguation (i.e., cases where the same term can be associated with multiple IDs). Moreover, these ontologies and their associated concepts are extensively referenced in scientific literature, much of which is accessible on the Web and can be utilized as training data for LLMs. We, however, also test the performance on a general-purpose ontology-based resource, i.e., **WD** [48], to study the memorization of LLMs in a more complex setting, including polysemy to provide a complete range of experiments.

GO provides a computational representation of current scientific knowledge about the functions of protein and non-coding RNA molecules produced by genes from many different organisms. It is widely used to support scientific research and is thus popular in Web publications. It contains 42,854 different concepts, each having a label and uniquely identified by a numerical ID prefixed by "GO:" (a.k.a., GO ID). For example, the label *kidney development* is associated with the concept whose GO

³In this work, use the terms *entity* and *concept* interchangeably.

ID is GO:0001822, while the label *kidney morphogenesis* to GO ID GO:0060993. As evident from the examples, without memorization, it would be impossible for the model to derive the correct GO ID given only the label. UO is an integrated cross-species anatomy ontology representing a variety of entities classified according to anatomical structure, function, and developmental lineage. In the UO, there are 15,543 terms (a.k.a., labels) that are uniquely associated with a numerical UBERON ID that is specific to the ontology, i.e., identified by the prefix “UBERON.” The ontology also incorporates terms from other ontologies (e.g., GO and BFO), which we have excluded from the analysis as they are not native to Uberon. While the UO is more recent and less prevalent in the scientific literature than GO,⁴ it allows us to compare performance across different resources and subdomains, i.e., proteins vs. anatomy. The third analyzed resource is ICD-10, a very popular and globally recognized classification system⁵ developed by the World Health Organization for coding and classifying diseases, injuries, and various health conditions. As for Gene and Uberon, each concept is associated with a unique alphanumeric code. It is widely used by healthcare providers, insurers, and health organizations to maintain records, track diseases, and collect data for health statistics. While GO and UO belong to the biomedical domain, ICD-10 belongs to the clinical one. Of all the concepts present in ICD-10 (73,201), we consider only those whose ID has no more than four digits, that is, those that describe diseases that are not too specific, obtaining a list of 11,494 concepts. This choice is mainly made to reduce the costs of the experiments. The fourth analyzed resource is WD, a general-purpose knowledge base that stores concepts by unique IDs. Analyzing all the concepts in WD (approximately 100M) would be very expensive and time-consuming. For this reason, a representative subsample of 30K concepts is extracted, having different popularity according to QRank, a measure that aggregates page view statistics for estimating the popularity of a concept.⁶ The WD setting is more complex because, being a general-purpose resource, it may have cases of polysemy (different IDs having the same label) that have to be addressed during the evaluation phase.

LLMs. We consider four different LLMs. The first one is the 12 billion parameters release of EleutherAI’s PYTHIA [3] (named PYTHIA-12B now on). It is trained on ThePile dataset [16], comprising 825 GiB of English text from different sources, including PubMed Central (90.27 GiB of data) and PubMed Abstract (19.26 GiB), which are biomedical portals and thus can include literature inherent to the GO and UO. Although initially ThePile was publicly released, it was later withdrawn, and at the time of these experiments, it is now no longer accessible, thus preventing the possibility of more in-depth analyses. The second analyzed LLM is Google GEMINI-1.5-FLASH (named GEMINI-1.5F now on), an API-accessible LLM whose training data is not publicly available nor declared. Finally, we test two OpenAI chatbot models,⁷ i.e., GPT-3.5-Turbo-0613 (named GPT-3.5 now on) and GPT-4-0613 (called GPT-4 now on), respectively. The details of both GPT-3.5 and GPT-4 training datasets are not publicly available.

3.3 Research Questions (RQs)

To study the degree of memorization of ontology IDs and labels in LLMs, we address the following RQs:

RQ1. Can LLMs correctly predict the ID of the concepts in a known ontology, only given the concepts’ label in the input prompt? Are there differences in performance between the LLMs

⁴To quantify, on 19 December 2023, the GO’s paper [1] is cited on Google Scholar 38,553 times, while the Uberon’s paper [38] 686 times.

⁵Classification systems, such as thesauri, are generally regarded as *lightweight* ontologies [17].

⁶Wikidata QRank: <https://github.com/brawer/wikidata-qrank> [Last access on 7 October 2024].

⁷<https://openai.com/chatgpt> [Last access on 7 October 2024].

analyzed? Are there any common patterns in errors made by LLMs? What is the impact of hallucination on models' performance? This RQ is addressed in Section 4.

RQ2. Does the accuracy of the ID prediction correlate with the number of times the ID-concept label association is found on the Web and thus (likely) present and frequent in the training material? Does the latter also influence errors made by LLMs? This RQ is addressed in Section 5.

RQ3. How does the observed consistency of ID prediction for a given prompted label, under various prompt repetitions or perturbations, inform our understanding of the memorization of the ID-concept label association? This RQ is addressed in Section 6.

The first two RQs investigate whether, how, and why LLMs have memorized some basic ontological information. The third one looks more closely into ways to empirically assess the extent of memorization in the model of such ontological information by observing whether the prediction of the LLMs changes when prompting it multiple times for the same information.

4 Memorization of Ontologies (RQ1)

4.1 Methodology

To address RQ1, we evaluate the performance of the considered LLMs on the prediction task described in Section 3.1 for all concepts defined in each ontology presented in Section 3.2. We calculate the accuracy as the proportion of all instances where the model returns the exact correct ID for a given concept's label over all the instances in the ontology (e.g., for the label *kidney development* from GO, the prediction is considered correct only if the ID is precisely GO:0001822). For WD, where polysemy can occur, we consider a predicted ID correct if it belongs to the set of IDs whose label matches the one specified in the prompt.

Quantifying Error Patterns. To analyze possible common patterns in errors made by LLMs on the given prediction task, i.e., predicting a sequence of digits (the concept ID) given one or more word tokens (the concept label), we propose to investigate whether some *syntactic* similarity (e.g., character or token-based) exists between the gold and wrongly predicted IDs, or the corresponding concept labels, following two complementary strategies.

The first approach investigates if the model tends to make mistakes by providing an ID similar to the correct one. For example, the GO ID GO:0060219 (*camera-type eye photoreceptor cell differentiation*) is close to GO:0060519 (*cell adhesion involved in prostatic bud elongation*) because only one digit needs to be changed to go from the first to the second. To do this analysis, we extract the list of wrong IDs predicted by the models and the gold one and compute the Levenshtein distance [33] on them.

The second approach investigates if the model tends to make mistakes by providing an ID whose corresponding label is similar to the correct one. For example, the label *heart valve morphogenesis* (GO ID GO:0003179) is close to *heart trabecula morphogenesis* (GO ID GO:0061384) due to two words in common, or the label *regulation of cell division* (GO ID GO:0051302) with *regulation of cell motility* (GO ID GO:2000145) because of three words in common. In these examples, the numerical IDs of the two concepts are very different, but the label has sub-tokens in common, which could lead to a prediction error by the model. To do this analysis, we extract the list of errors committed by the models by collecting the gold label (i.e., the correct one) and the label corresponding to the wrong ID predicted by the model. Then, we divide the two strings into tokens and compute the Jaccard similarity [26] on them.

Model Hallucinations. We also check if the models tend to predict IDs that do exist in the ontology or if they sometimes make up the proposed IDs. To do so, we first collect all the unique IDs

predicted by the model and determine how many are invented, i.e., not present in the corresponding ontology.⁸ Additionally, we calculate the percentage of incorrect predictions where the model invents a plausible, yet non-existent, ID.

4.2 Experimental Setup

For RQ1, we assessed the accuracy of PYTHIA-12B, GEMINI-1.5F, GPT-3.5, and GPT-4 on GO, UO, ICD-10, and WD. For each model, we first tried slightly different prompts on a sample of 100 random entities, achieving comparable performance across the variations.⁹ Therefore, we selected the ones with the least number of input tokens to reduce experimentation costs for paid models. For chatbots models (GPT-3.5, GPT-4, and GEMINI-1.5F), the following prompts were selected:

- *Provide the GO ID for the label “l.” In the answer write only the corresponding GO ID.*
- *Provide the UBERON ID for the label “l.” In the answer write only the corresponding UBERON ID.*
- *Provide the ICD-10 ID for the label “l.” In the answer write only the corresponding ICD-10 ID.*
- *Provide the WD ID for the label “l.” In the answer write only the corresponding WD ID.*

While for PYTHIA-12B, we selected the following text completion prompts (requesting the addition of maximum 10 new tokens):

- *In the GO, the GO ID of the label “l” is GO:*
- *In the UO, the UBERON ID of the label “l” is UBERON:*
- *In the ICD-10, the ICD-10 ID of the label “l” is*
- *In the WD, the WD ID of the label “l” is*

All the prompts are executed with the temperature set to .0.¹⁰

4.3 Results

Table 1 reports the accuracy of the four compared models on the task described in Section 3.1, for the four considered ontologies. Table 2 summarizes the resulting Jaccard similarity and Levenshtein distance, averaged over all the wrong predictions by the models on all considered ontologies. Finally, Table 3 provides details about model hallucinations: Table 3(a) summarizes the number of unique IDs produced by the models across the different ontology settings, along with the percentage of IDs invented by the model (i.e., non-existent IDs in the ontology), while Table 3(b) shows the percentage of wrong predictions caused by invented IDs.

4.4 Discussion

Table 1 shows that GPT-4 and GPT-3.5 are partially familiar with both ICD-10 and GO by obtaining an accuracy of .37 and .25, respectively, on the first and .13 and .06, respectively, on the second. GEMINI-1.5F and PYTHIA-12B reach on ICD-10 and GO lower performance than GPT-3.5 and GPT-4, but they still know ICD-10 and GO more than WD and UO. On all the datasets, GPT-4 has an accuracy higher than all the other models: This result aligns with the findings of other papers on LLMs tokens memorization, e.g., Carlini et al. [8]. PYTHIA-12B achieves the lower accuracy scores

⁸In the cases where the prediction task described in Section 3.1 is assessed on a subset of the original ontology (i.e., ICD-10 and Wikidata), an ID is considered invented only if it is not present in the entire original ontology (rather than just the subset).

⁹All considered prompts are available in the GitHub repository of the work.

¹⁰We used a regular expression to extract only the concept ID from responses to ensure an accurate evaluation of the outputs. This approach allows us to handle outputs containing additional text, such as “The ID is GO:XXXX” by isolating the relevant ID and avoiding unnecessary penalties. As a result, such responses are not discarded or penalized, thereby mitigating potential issues related to output constraint violations, which have been highlighted as a concern for recent OpenAI models [15].

Table 1. Accuracy of PYTHIA-12B, GEMINI-1.5F, GPT-3.5, and GPT-4 on the Task Described in Section 3.1, on the GO, UO, ICD-10, and WD Datasets

Dataset	PYTHIA-12B	GEMINI-1.5F	GPT-3.5	GPT-4
GO	.0067	.0140	.0611	.1270
UO	.0000	.0003	.0035	.0129
ICD-10	.0061	.0648	.2487	.3749
WD	.0000	.0001	.0026	.0073

The score of the best-performing model on each ontology is in bold.

Table 2. Average Similarity Measures of Wrongly Predicted IDs (Levenshtein Distance, Abbreviated with Levenshtein d.) and Associated Concept Labels (Jaccard Similarity, Abbreviated with Jaccard s.) for PYTHIA-12B, GEMINI-1.5F, GPT-3.5, and GPT-4 Models, on GO, UO, ICD-10 (ICD), and WD

Measure	PYTHIA-12B	GEMINI-1.5F	GPT-3.5	GPT-4
Levenshtein d. (GO)	4.409	4.317	3.964	3.814
Jaccard s. (GO)	.149	.186	.301	.338
Levenshtein d. (UO)	6.570	4.048	3.533	4.580
Jaccard s. (UO)	.001	.016	.047	.033
Levenshtein d. (ICD)	5.348	5.339	5.387	5.382
Jaccard s. (ICD)	≈ .000	.001	.004	.008
Levenshtein d. (WD)	4.593	5.039	4.669	4.662
Jaccard s. (WD)	≈ .000	.003	.001	.014

in all datasets and gets close to zero accuracy in both WD and UO. GEMINI-1.5F also achieves an accuracy that is nearly zero on them, but its performance on ICD-10 and GO is much higher than PYTHIA-12B's. We can thus observe that:

- Overall, the accuracy scores are quite low for all models and ontologies considered, with the exception of GPT-3.5 and GPT-4 on ICD-10.
- For all datasets, GPT-4 and GPT-3.5 perform better than the others, with GPT-4 being the best-performing LLM, followed by GPT-3.5 and GEMINI-1.5F, with PYTHIA-12B achieving the lowest scores: A possible explanation is the different number of model parameters (much higher for GPT-4 and GPT-3.5 than GEMINI-1.5F and PYTHIA-12B) or the different amount of data used in the training phase¹¹: For instance, the zero accuracy of PYTHIA-12B on UO can be ascribed to the almost total absence of Uberon IDs in its training dataset (15,494 of the 15,543 Uberon IDs are not present at all in PubMed, a Web site used to train PYTHIA-12B), which instead contains occurrences of several GO and ICD-10 IDs.
- The performances on the UO and WD are lower than on the ICD-10 and GO for all the considered models: A possible explanation could be that in ICD-10 and GO, the IDs are more frequently used in practice by domain experts than UO and WD ones, and thus the concept ID-label pairs for the latter may have been seen far less in the training material of the models (more on this in Section 5).

¹¹We recall that, while for PYTHIA-12B the dataset used for training is declared (but not available anymore), for GPT-3.5, GPT-4, and GEMINI-1.5F models it is not.

Table 3. (a) Number of Unique IDs (Unq) Produced by the Models (PYTHIA-12B, GEMINI-1.5F, GPT-3.5, and GPT-4) for GO, UO, ICD-10 (ICD), and WD, and Percentage of Unique IDs Being Invented, i.e., Hallucinated, by the Model (%Inv); (b) Percentage of Wrong Predictions Caused by Invented IDs

(a) Unique predicted IDs and percentage of them being invented due to hallucination

	PYTHIA-12B		GEMINI-1.5F		GPT-3.5		GPT-4	
	Unq	%Inv	Unq	%Inv	Unq	%Inv	Unq	%Inv
GO (42,854)	1,353	10.64	5,558	16.61	7,182	7.64	12,308	9.60
UO (15,543)	30	30.00	237	5.06	641	10.92	2,971	33.52
ICD (11,494)	1,002	61.98	3,288	44.65	5,593	27.50	6,633	20.08
WD (30,000)	125	26.40	8,817	28.16	7,734	16.89	15,539	19.31

(b) Percentage of wrong predictions caused by invented IDs due to hallucination

	PYTHIA-12B	GEMINI-1.5F	GPT-3.5	GPT-4
GO	6.85	11.84	6.35	7.63
UO	12.10	1.96	2.34	15.94
ICD	53.62	43.51	32.59	30.18
WD	2.76	28.91	11.55	15.13

Next to each ontology in (a), the number of expected unique IDs is reported.

Table 2 summarizes the results of the error analysis. Concerning the Levenshtein distance of the wrongly predicted IDs with respect to the gold ones, the difference in performance across the model is rather minimal on all considered ontologies. This may be because all models tend to predict IDs that, even if wrong, have a format comparable with the gold ones, especially in terms of used characters and length. The differences across the ontologies are also quite minimal, ranging from 3.814 on the GO (with GPT-4) to 6.570 on UO (with PYTHIA-12B). Looking at the Jaccard similarity scores, the models achieving the highest values when making wrong predictions are GPT-4 and GPT-3.5, followed by GEMINI-1.5F and PYTHIA-12B, meaning that the former tend to predict (wrong) IDs whose associated concept labels are more similar to the one corresponding to gold IDs than the latter. Overall, the Jaccard similarity of all models is much higher in GO than the other ontologies.

Concerning models' hallucinations, Table 3 first shows that all models tend to produce only a portion of the expected IDs for the considered ontologies, thus indicating that each model repeatedly predicts the same (wrong) IDs for different concept labels (Table 3(a)). The lower numbers of unique IDs across all ontology settings are generated by PYTHIA-12B, meaning that this model tends to predict few, frequently repeated IDs for multiple labels, while GPT-4 is by far the model producing the higher number of different IDs for each ontology. The models literally invent several of the wrongly predicted IDs. The percentage values (%Inv) of invented IDs over the produced ones actually vary a lot across models and ontologies, ranging from 5.06% for GEMINI-1.5F on UO to 61.98% for PYTHIA-12B on ICD-10. For PYTHIA-12B, GEMINI-1.5F, and GPT-3.5, the highest proportion of invented IDs occur on ICD-10 (respectively, 61.98%, 44.65%, and 27.50%), while for GPT-4 on UO (33.52%). Except for GEMINI-1.5F, the models invent proportionally fewer unique IDs on the GO. Finally, looking at the percentage of errors due to invented IDs (Table 3(b)), the highest values are

observed on ICD-10 for all models, ranging from 30.18% for GPT-4 to 53.62% for PYTHIA-12B. The results indicate that hallucinations significantly impact model performance, likely due to issues in the training data, the training process, or inference. While beyond the scope of our work, we acknowledge that mitigation methods (e.g., [24]) could be applied to partially address the model's hallucinations.

Answer to RQ1. The results show that the considered LLMs have rather minimal knowledge of the information in the considered ontologies. GPT-4 substantially outperforms all the other analyzed models in terms of accuracy, and even when making wrong predictions, the errors made are generally closer to the gold one than the other models. Finally, all models exhibit some form of hallucination when performing the task.

5 Memorization and Popularity on the Web (RQ2)

5.1 Methodology

RQ2 aims to assess the correlation (if any) between the popularity of a concept on the Web—and by extension, its likely presence in LLMs' training material—and the model's memorization of it. It is important to note that precise information on the actual training data used for certain models (namely GPT-3.5, GPT-4, and GEMINI-1.5F) is not publicly available. Additionally, the dataset used to train PYTHIA-12B, known as ThePile, is no longer accessible. However, much of ThePile consists of publicly available Web material, including sources like PubMed, and it is well-established that models such as GPT-4, GPT-3.5, and GEMINI-1.5F were trained, at least in part, on large volumes of Web content [10].

To explore this relationship, we construct a dataset that includes information on the popularity of ontology (*ID, label*) pairs on the Web. Popularity is approximated by the number of times the pair appears together in the same Web documents. To estimate this, we use Google Search APIs with the query format “label” “ID” to obtain exact matches from Web content. While the implementation details of Google Search are not publicly disclosed, making it a black-box tool [29], it is widely used for similar research purposes [10].

For the analysis, we group the ontology concepts into buckets based on the popularity of their (*ID, label*) pairs on the Web. We then examine how the model's average accuracy in predicting IDs varies across these buckets (i.e., the proportion of correctly predicted IDs for the concepts within each bucket). To assess potential correlations between Web popularity and model accuracy, we apply Spearman's rank correlation [44]. Additionally, we evaluate whether Web popularity Granger-causes [18] model accuracy, i.e., whether a series of values for the former helps predict the latter.¹²

Web Occurrences vs. PubMed Occurrences. To provide some further evidence for our hypothesis that Web occurrences are a viable way to approximate the distribution of ontology (*ID, label*) pairs in the training material of LLMs, we assess the correlation and Granger causality between the occurrences of domain-specific ontology concepts (GO) in the Web and in a domain-specific resource (PubMed) that was used, among other non-domain-specific resources (and thus unlikely to contain substantial information about the given ontology), to train one of the considered LLMs (PYTHIA-12B). We use the Google Search APIs, restricted to the PubMed domain, to collect the required dataset-specific (*ID, label*) occurrences for the considered ontology. Spearman's rank statistical correlation and Granger causality are computed between the distribution of Web occurrences and

¹²While typically applied to time series data, the concept of Granger causality can also be applied to cross-sectional data [35], i.e., collected at a single point in time, provided that there is a clear ordering or progression of the data, in our cases given by the ranking of Web occurrences. Similar considerations also apply to later uses of Granger causality in the article.

PubMed occurrences, as well as between PubMed occurrences and the PYTHIA-12B's prediction accuracy. The statistical significance of the values is computed according to the Permutation test.

Error Patterns vs. Popularity. We also complement the study of possible common patterns in mistakes made by LLMs on the given prediction task (cf. Section 4.1) with further analysis related to the popularity of the concepts. First, we check whether the similarity between wrong and gold predictions based on Levenshtein distance and Jaccard similarity correlates with the number of Web occurrences of the concepts. Then, we investigate if the model is biased toward popular concepts, i.e., if it tends to answer with the ID of a very popular concept. For the latter, we extract the most common incorrectly predicted IDs and check which buckets they belong to estimate if there is a Spearman's rank statistical correlation between frequently predicted IDs and their bucket number, with statistical significance computed according to the Permutation test.

Moreover, we further investigate the repeated IDs phenomenon discussed at the end of Section 4.3 in light of the popularity data. We collect the top- k most repeated IDs for each model for the GO and check if they are uniformly distributed in the buckets according to the actual distribution of the concepts in the buckets. More in detail, for each bucket B_i , we compute

$$R_{B_i} = \frac{n_{B_i}}{\frac{N_{B_i}}{N} \cdot k},$$

where n_{B_i} is the number of top- k repeated IDs that belong to bucket B_i , N_{B_i} is the total number of IDs in bucket B_i , and N is the total number of IDs across all buckets.

5.2 Experimental Setup

We computed the Web occurrences for all datasets considered in our work (GO, UO, ICD-10, and WD). However, given the very low performance on the UO and WD ontology for all models—cf. Section 4.3, we considered only GO and ICD-10 to address RQ2. To group elements into buckets based on Web popularity, we calculated the 50 percentiles of the observed Web occurrences and allocated the ontology elements into the corresponding frequency-based buckets. Given the distribution of occurrences in GO and ICD-10, the first buckets, corresponding to less frequent ($ID, label$) pairs, contain much more elements than the last ones, where very frequent ($ID, label$) are allocated. We arbitrarily set the value to 50 to ensure that the obtained buckets have a substantial number of elements to conduct the analysis. Similarly, 50 buckets were also created to organize the PubMed occurrences of the GO, to study the correlation between Web occurrences and PubMed occurrences, and between PubMed occurrences and the PYTHIA-12B's prediction accuracy. To study the relation between repeated IDs and popularity we empirically set $k = 500$, thus considering the top-500 most repeated IDs for each model. All the prompts are executed with the temperature set to .0.

5.3 Results

Figure 1 reports the popularity of ID-label pairs for all considered datasets on the Web. The two plots in Figure 2 show how the accuracy of the ID predictions of the models, based on the prompted concept label for the GO (Figure 2(a)) and ICD-10 (Figure 2(b)), varies according to how frequently that ($ID, label$) pair occurs on the Web. IDs were organized in buckets, labeled from B1 (least frequent ($ID, label$) pairs—rarely observed) to B50 (most frequent ($ID, label$) pairs—observed tens of thousands of times) based on the number of occurrences in the Web. Table 4 reports the resulting Spearman's rank correlation coefficient (Table 4(a)) and the Granger causality F-statistic (Table 4(b)) between the number of Web occurrences of the ($ID, label$) pairs and the accuracy of all the models for the GO and ICD-10.

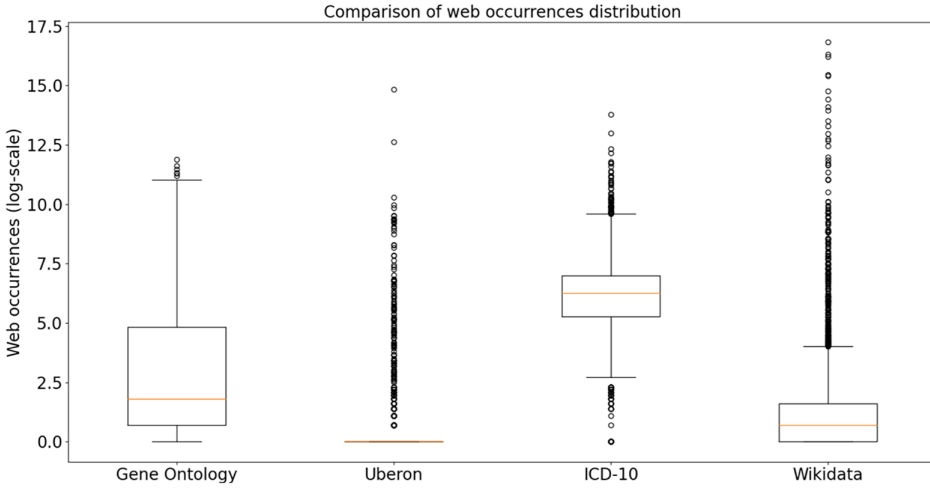


Fig. 1. Distribution of the number of Web occurrences (log-scaled, base e) for each dataset (GO, UO, ICD-10, and WD).

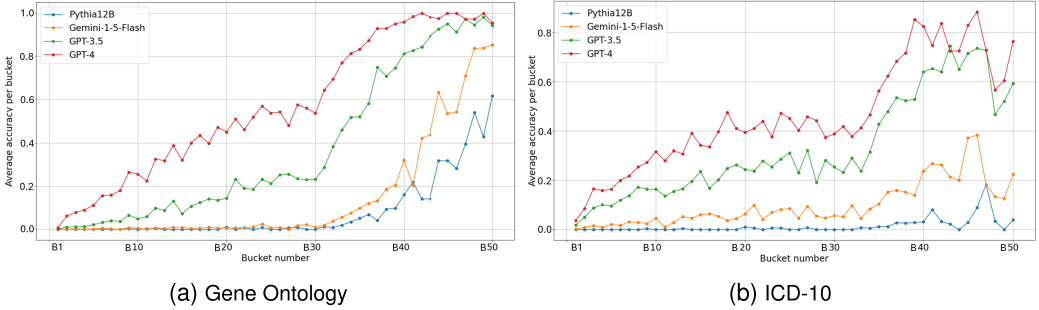


Fig. 2. Average accuracy of the model's ID prediction (y-axis) according to the popularity of the concept on the Web (represented by the bucket number in x-axis) in the GO (a) and ICD-10 (b).

Web Occurrences vs. PubMed Occurrences. Table 5 reports the resulting Spearman's rank correlation coefficient and the Granger causality F-statistic between (i) the number of Web occurrences and the number of PubMed occurrences and (ii) the number of PubMed occurrences of the $(ID, label)$ pairs and the accuracy of PYTHIA-12B's prediction for the GO.

Error Patterns vs. Popularity. Table 6 reports the Spearman's rank correlation coefficient between the ranking of the buckets according to the averaged Levenshtein distance/Jaccard similarity per bucket and the ranking of the buckets. Finally, the plots in Figure 3 show the ratio R_{B_i} between actual and expected top-500 repeated IDs for all buckets B_i for both GO (Figure 3(a)) and ICD-10 (Figure 3(b)).

For GO, a strong correlation was observed for PYTHIA-12B (.868), GEMINI-1.5F (.871), and GPT-3.5 (.947), both statistically significant, while no correlation was detected for GPT-4. For ICD-10, a strong correlation was observed for PYTHIA-12B (.901) and GEMINI-1.5F (.827), both statistically significant, while no correlation was detected for GPT-3.5 and GPT-4.

Table 4. Spearman's Rank Correlation Coefficient (a) and Granger Causality F-Statistic with Lag = 3 (b) between the Number of Web Occurrences of the (*ID, label*) Pairs and the Accuracy of the Models for the GO and ICD-10

(a) Correlation results				
	PYTHIA-12B	GEMINI-1.5	GPT-3.5	GPT-4
GO Correlation	.850*	.924*	.993*	.982*
ICD-10 Correlation	.692*	.901*	.933*	.919*
(b) Causality results				
	PYTHIA-12B	GEMINI-1.5	GPT-3.5	GPT-4
GO Causality	6.753*	7.498*	3.336*	2.853*
ICD-10 Causality	30.338*	1.089	1.711	2.592

Scores marked with “*” means the reported value is statistically significant (p-value $\leq .05$).

Table 5. Spearman's Rank Correlation Coefficient and the Granger Causality F-Statistic (Lag = 3) between the Number of Web Occurrences and the Number of PubMed Occurrences (First Column) and the Number of PubMed Occurrences of the (*ID, label*) Pairs and the Accuracy of PYTHIA-12B's Prediction for the GO (Second Column)

	Web occurrences vs. PubMed occurrences	PubMed occurrences vs. PYTHIA-12B accuracy
GO Correlation	.796*	.865*
GO Causality	527.386*	6.825*

Scores marked with “*” means the reported value is statistically significant (p-value $\leq .05$).

Table 6. Spearman's Rank Correlation Coefficient between the Levenshtein Distance/Jaccard Similarity Measures and the Ranking of the Buckets According to the Averaged Number of Web Occurrences of the (*ID, label*) Pairs in the Bucket for GO and ICD-10

Similarity measure	PYTHIA-12B	GEMINI-1.5	GPT-3.5	GPT-4
(GO) Levenshtein d.	-.714*	-.853*	-.364*	-.208
(GO) Jaccard s.	.788*	.874*	.590*	.595*
(ICD-10) Levenshtein d.	-.948*	-.957*	-.945*	-.926*
(ICD-10) Jaccard s.	.082	.233	.242	.310*

Only wrong predictions are considered. Scores marked with “*” means the correlation is statistically significant (p-value $\leq .05$).

5.4 Discussion

Figure 1 shows that ICD-10 is the dataset with the highest median number of documents on the Web containing both the ID and label for its concepts, followed by GO, WD, and UO. Partly surprisingly given its wide adoption and its broad general-domain coverage, WD IDs and labels are infrequently used together in the same Web documents. In contrast, IDs and labels of domain-specific resources like ICD-10 and GO are more frequently used together. This opposite situation may be because ICD-10 and GO are highly specialized ontologies used extensively in their respective

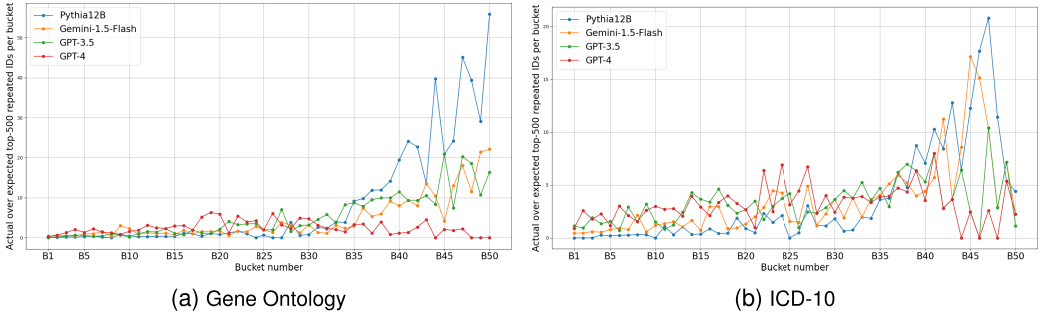


Fig. 3. Variation over all the buckets of the R_{B_i} values, capturing the ratio between the number of top-500 repeated IDs that belong to bucket B_i , and the proportion of them that should be in the bucket B_i according to the overall distribution of all IDs in the buckets for GO (a) and ICD-10 (b).

fields (clinical medicine and biology), where precise and standardized terminology is crucial for accurate communication and documentation, while WD entities may be referred to in many cases only by their label (or description) without explicit reference to the specific ID.

Comparing these values with the accuracy score of the various LLMs on the ontologies reported in Table 1, this may indeed suggest that the more some concept ($ID, label$) pairs are seen in Web documents (and, thus, likely in training material of LLMs), the more the information is memorized by the models, and thus correctly predicted when prompted about it. Indeed, this connection between Web occurrences and accuracy finds more confirmation in Figure 2, which shows how the accuracy scores of all the models vary based on the number of Web occurrences of the GO and ICD-10 concepts. Indeed, with minimal variations, we observe that accuracy substantially improves as we move from low-frequency concepts to high-frequency ones. That is, the plots show that the considered models tend to make correct predictions for concepts frequently observed on the Web (and, thus, likely commonly observed in the training material), and wrong predictions for less frequently observed concepts, thus suggesting a certain degree of correlation between the accuracy of the predictions (averaged per bucket), i.e., the degree of concepts correct memorization, and the occurrence of the concepts in the Web. Indeed, for GO, this is confirmed by the correlation and Granger causality scores reported in Table 4: There is a strong correlation (statistically confirmed for all LLMs) between Web occurrences and the model's accuracy, with the former that Granger causes the latter (statistically confirmed for all LLMs). For ICD-10, there is a strong and statistically relevant correlation between Web occurrences and the model's accuracy (statistically confirmed for all LLMs), with the former that Granger causes the latter for PYTHIA-12B.

The accuracy score of GPT-4 on (almost) every bucket is higher than that of GPT-3.5, which in turn is higher than that of GEMINI-1.5F, which is higher than that of PYTHIA-12B. For some high-frequency buckets, GPT-4 scores reach perfect accuracy on the GO, and above .800 on ICD-10.

Web Occurrences vs. PubMed Occurrences. The results in Table 5 confirm a strong correlation between Web and PubMed occurrences for the GO ($ID, label$) pairs with the latter that Granger causes the former, thus suggesting that, despite differences in the absolute value of the occurrences (the average number of occurrences for a GO concept on the Web is approximately 50 times higher than in PubMed), their overall distributions are comparable. Indeed, Table 5 also shows that a strong correlation exists between PubMed occurrences and PYTHIA-12B's prediction accuracy, with the former that Granger causes the latter, exactly as observed for Web occurrences (cf. Table 4). Even taking into account all the limitations previously mentioned, we believe this is yet another indication that Web occurrences could be taken as a reasonable approximation of the training

material of LLMs in all those (many) cases where no information on the latter is provided, at least for the considered task.

Error Patterns vs. Popularity. The results in Table 6 show that in most of the considered model/ontology configurations, when the models make prediction errors, the wrongly predicted IDs are syntactically closer to the gold one (directly or indirectly through the associated labels) for concepts frequently occurring on the Web rather than infrequent ones. Indeed, a strong inverse correlation is observed between Levenshtein distance and Web occurrences for both ontologies and for all models, except for GPT-3.5 (moderate) and GPT-4 (poor) on GO, all statistically significant (except for GPT-4). The situation is more varying for the Jaccard similarity, where moderate to strong correlation (statistically significant) is observed for all models on the GO, while the correlation is poor/moderate for all models on ICD-10. Again, these findings further suggest that the more the $(ID, label)$ association occurs on the Web, the more the models are likely closer to making the correct prediction.

Concerning the experiments on the distribution of the top-500 repeated IDs, the plot for the GO (Figure 3(a)) clearly shows that for PYTHIA-12B, GEMINI-1.5F, and GPT-3.5, most of them proportionally belong to the buckets containing the most occurring $(ID, label)$ pairs on the Web, while for GPT-4 these IDs seem to be more proportionally distributed across all buckets. Indeed, the reported correlation values confirm that PYTHIA-12B, GEMINI-1.5F, and GPT-3.5 are somehow biased toward frequently occurring concepts on the Web, while this does not hold for GPT-4. Although moderately noisier, especially toward the top-most high-frequency buckets (B48-B50), the plot for ICD-10 (Figure 3(b)), shows a similar trend as for the GO. In this case, the reported correlation values confirm that PYTHIA-12B and GEMINI-1.5F are more biased toward frequently occurring concepts on the Web, while this does not hold for GPT-3.5 and GPT-4. That is, when making errors, the less-performing (PYTHIA-12B and GEMINI-1.5F) models seem to repeatedly predict the IDs-label that they have likely seen the most in the training data, while the better-performing models are marginally (GPT-3.5) or not affected at all (GPT-4) by this.

Answer to RQ2. The results show that the prediction accuracy of the considered LLMs substantially correlates with the occurrence of the $(ID, label)$ associations in Web documents: The more the latter, the better the former. Indeed, under the assumption that these models are trained on a vast amount of Web content, and thus that Web occurrences can be taken as a good approximation of occurrences in the training material, these findings suggest that the more some information is seen in the training material, the more it is memorized by the models, as shown by the prediction accuracy as well as the way the models fail in predicting the correct IDs.

6 Assessing the Relationship between Prompt Invariance and Memorization (RQ3)

6.1 Methodology

To assess to which extent the $(ID, label)$ association for an ontological concept is correctly memorized in an LLM (cf. RQ3), we propose using the model's response invariance as a metric. We suggest measuring the variability in the model's answers when the prompt is repeatedly submitted or perturbed in different ways to indicate how well the correct ID for a given label is memorized. Specifically, we investigate three different strategies applied to the base prompts selected in Section 4.2:

- *PI-1:* We repeat the same prompt more times asking for determinism in the answer and counting the number of different answers. Intuitively, if the $(ID, label)$ association is well memorized in the model, only one answer should always be returned.

- *PI-2*: We repeat the same prompt adjusting the degree of randomness in the model’s output, ranging from deterministic to more creative behavior. Intuitively, if the model well memorizes the $(ID, label)$ association, the answer should remain consistent regardless of the temperature, meaning it should not change even as the model’s creativity increases.
- *PI-3*: We repeat the same prompt in different languages asking for determinism. In this implementation, we translate only the prompt while we keep the ontology’s labels in the original language, i.e., English. Intuitively, if the $(ID, label)$ association is well memorized in the model, the answer should be independent of the language used to make the request.

In detail, given a set of N buckets B_i , with $i \in [1, 2, \dots, N]$, each containing K random concepts $C_{B_i} = [c_1, c_2, \dots, c_K]$,¹³ we evaluate the performance by considering the mean number U of unique answers returned by the model for concepts belonging to each bucket B_i . In particular, for each concept $c_j \in B_i$, we compute the *Prediction Invariance* after M prompt repetitions as

$$PI_{c_j}^M = 1 - \left(\frac{U - 1}{M - 1} \right),$$

where U is the number of unique answers returned after M prompts. If the M prompts return always the same answer for a given c_j , PI_{c_j} will be 1.0 (i.e., max prediction invariance). If the M prompts return M different answers for a given c_j , PI_{c_j} will be .0 (i.e., minimum prediction invariance). We then average the different PI_{c_j} obtained for each concepts $c_j \in B_i$, obtaining the **Average Prediction Invariance (AvPI)** for each bucket B_i as

$$AvPI_{(B_i)} = \frac{1}{K} \sum_{c_j \in C_{B_i}} PI_{c_j}^M.$$

Finally, we use Spearman’s rank correlation coefficient to study the relationship between $AvPI_{(B_i)}$ and accuracy for the same bucket B_i (i.e., the correctness of the predictions).

6.2 Experimental Setup

Similar to RQ2, we address this RQ using the GO and ICD-10 datasets. For the LLMs, we conduct the tests with GPT-3.5, GEMINI-1.5F, and PYTHIA-12B.¹⁴ Starting from the same $N = 50$ buckets used for the previous experiments, we considered $K = 20$ random IDs per bucket, for a total of 1,000 instances, allowing us to study the impact of the three invariance strategies on the model’s behavior according to the popularity of the concepts. In PI-1, the same prompt is repeated $M = 10$ times with the temperature set to zero. In PI-2, we tested $M = 11$ different temperature levels, from .0 to 1.0 with .1 increments. In PI-3, we tested $M = 5$ different languages, namely English, Italian, German, French, and Spanish. The prompts were manually translated by people proficient in the respective languages.

6.3 Results

Figure 4 shows the trend of $AvPI$ (left) and accuracy (right) for GPT-3.5, GEMINI-1.5F, and PYTHIA-12B on the GO. The Spearman’s rank coefficient between $AvPI$ and accuracy is high (moderate to very strong correlation) for both PI-2 and PI-3 for GPT-3.5 (.950 and .850, respectively), GEMINI-1.5F (.813 and .743, respectively), and PYTHIA-12B (.795 and .581, respectively), all statistically confirmed ($p < .05$) by the Permutation test. No statistical correlation is observed for PI-1.

¹³While the approach could be applied to all concepts of an ontology, we limited their number to economize on the costs of the LLM APIs.

¹⁴To minimize experiment costs, we excluded GPT-4; however, initial assessments indicated a similar trend to that observed with GPT-3.5.

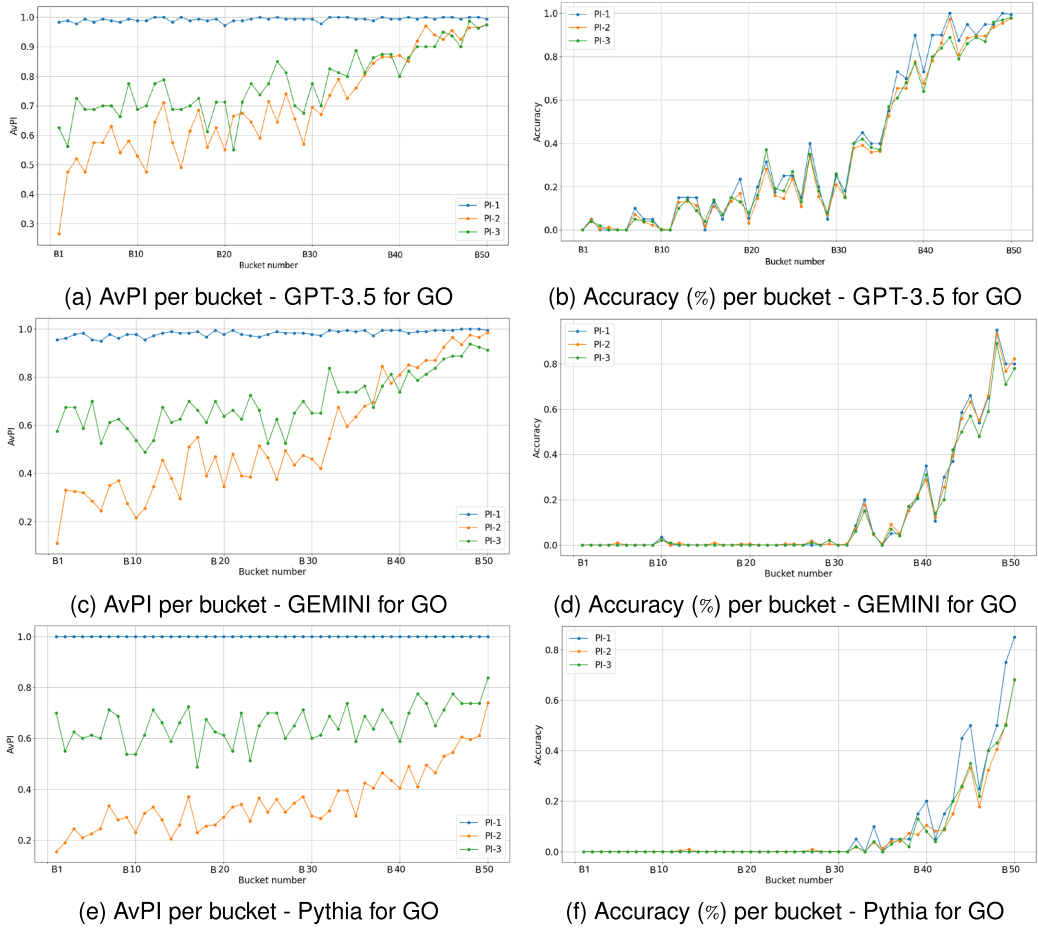


Fig. 4. Variation of AvPI (left) and accuracy (right) on the different buckets of the GO when applying the PI-1, PI-2, and PI-3 invariance strategies to GPT-3.5 (a, b), GEMINI-1.5F (c, d), and PYTHIA-12B (e, f).

Figure 5 shows instead the trend of AvPI (left) and accuracy (right) for GPT-3.5, GEMINI-1.5F, and PYTHIA-12B on ICD-10. The Spearman's rank coefficient between AvPI and accuracy is high (moderate to very strong correlation) for both PI-2 and PI-3 for GPT-3.5 (.874 and .774, respectively), for GEMINI-1.5F (.700 and .535, respectively), and for PYTHIA-12B (.613 and .609, respectively). The Permutation test statistically confirms all these correlation values ($p < .05$). No correlation is observed for PI-1 for GPT-3.5, GEMINI-1.5F, and PYTHIA-12B.

6.4 Discussion

Figure 4(a) shows that for GPT-3.5 on the GO, for frequently occurring concepts, i.e., those belonging to the very last buckets, all the methods achieve an AvPI score close to 1; that is, the model almost always predicts the same ID for a given concept label, and the prediction is almost always the correct one (accuracy close to 1 as well). For infrequently occurring concepts, i.e., those belonging to the first buckets, while PI-1 achieves an AvPI score close to 1, the application of PI-2 and PI-3 results in a higher variability of the model predictions, i.e., lower AvPI score, which is more substantial when the prompt is repeated with different temperature levels (PI-2). Interestingly, in bucket B1,

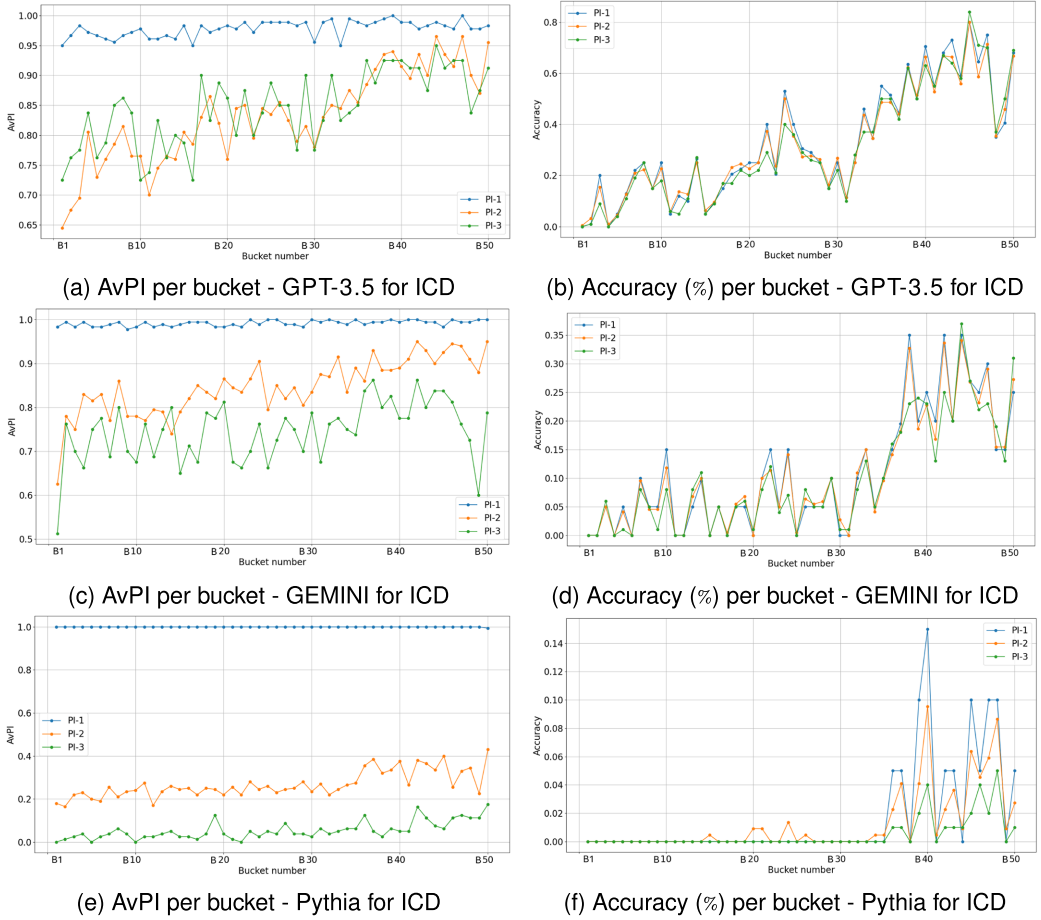


Fig. 5. Variation of AvPI (left) and accuracy (right) on the different buckets of ICD-10 when applying the PI-1, PI-2, and PI-3 invariance strategies to GPT-3.5 (a, b), GEMINI-1.5F (c, d), and PYTHIA-12B (e, f).

all methods return an average accuracy of zero, meaning that even if the methods tried different ID predictions among the various repetitions, none of them was the correct one. The increasing trend for both PI-2 and PI-3, which strongly correlates with the accuracy in Figure 4(b), indicates that the more a concept is repeated on the Web (and therefore potentially in the model's training material), the more invariant the model predictions are when solicited with PI-2 and PI-3 methods. In contrast, the PI-1 method is not particularly effective for assessing correctly memorized information, as the model tends to return the same answers consistently, regardless of their accuracy. Similar trends on the GO can also be observed for GEMINI-1.5F (Figure 4(c) and (d)) and PYTHIA-12B (Figure 4(e) and (f)), although for the latter the AvPI vs. accuracy correlation is weaker for PI-3 than PI-2: This may be explained by the fact that PYTHIA-12B was primarily trained on English material, and therefore, a method based on multilingualism might not be optimal.

For ICD-10, similar conclusions can be reached for GPT-3.5: The plots for AvPI (Figure 5(a)) and accuracy (Figure 5(b)) have similar increasing trends, confirmed by the high correlation for both PI-2 and PI-3. Again, as for the GO, the PI-1 method is not particularly effective for assessing correctly memorized information. GEMINI-1.5F has similar trends on ICD-10 as on the GO, with

moderate/high correlation for PI-2, while a less marked correlation is reported for PI-3. This may be explained by its lower knowledge of ICD-10 compared to the GO, as noted by the lower accuracy scores in Figure 2, especially for high-frequent concepts. Similar conclusions can also be reached for PYTHIA-12B even if it presents lower AvPI variations and lower accuracy values. This is likely due to the overall lower performance of PYTHIA-12B on ICD-10 than the GO, as shown in Figure 2.

Answer to RQ3. Our results indicate that when an LLM has some knowledge of a given ontological resource—like GPT-3.5, GEMINI-1.5F, and partly PYTHIA-12B of the GO, and GPT-3.5 and partly GEMINI-1.5F of ICD-10—looking at the variation of the predictions when invoked multiple times with the same prompt but varying temperature level (PI-2) or language (PI-3) may provide hints into the extent to which that information is memorized in the model: If the model (almost) always predicts the same information (AvPI close to 1), it is potentially because it has seen that information many times in the training material, and thus has memorized it. Furthermore, this prompt-invariant answer is likely correct, as suggested by the (moderate to very strong) correlation between AvPI and accuracy. Conversely, if the model tends to predict different information (AvPI heading toward 0) when varying the prompt, it may be because the LLM has not memorized that concept due to its scarce occurrence in the training data.

7 Conclusions

In this article, we investigate to what extent LLMs have acquired, i.e., correctly memorized, the vocabulary of concept IDs and labels from various publicly available ontologies. Our experiments, which span several LLMs (namely, PYTHIA-12B, GEMINI-1.5F, GPT-3.5, and GPT-4) and ontological resources (namely, GO, UO, WD, ICD-10), show that only a tiny fraction of concepts is memorized. Among the LLMs analyzed, GPT-4 is the one that obtains the highest performance. While trying to answer the question of why some instances are correctly memorized and others not, we found for all the LLMs a substantial correlation between the popularity of the concept (i.e., the number of concept occurrences on the Web) and the correctness of the prediction. This correlation may be ascribed to the fact that the resources used for training the models likely include substantial textual material from the Web. This is indeed in line with previous research, which has shown that LLMs often achieve high performance across multiple benchmarks but also that such performance could be attributed to potential data contamination during training and thus memorization of gold annotations [13]. Moreover, in our experiments, we show that the more a concept is widespread on the Web and thus likely seen in the LLMs' training data, the more the wrong prediction is closely related, in terms of Jaccard similarity, to the gold-standard one. Finally, we proposed three different strategies for computing a prediction invariance metric for estimating the correct memorization of concepts in LLMs, showing that the invariance of the model output to the language of the prompt or the configured model temperature can be used as evidence of information memorization.

Overall, our findings indicate that while LLMs acquire some structured knowledge from its presence on the Web, similar to other forms of knowledge, they also exhibit a significant tendency to “dream” (i.e., hallucinate) about them. This behavior aligns with their overarching objective of providing responses at all costs, prioritizing assistive interaction over factual accuracy.

8 Limitations and Future Work

We acknowledge that this study has some limitations.

First, our analysis was conducted on a limited set of LLMs, each varying in size and popularity. While models such as GPT-4 and GPT-3.5 are widely used, others like PYTHIA-12B are comparatively less popular, which may influence the generalizability of our findings. A broader evaluation of

LLMs across different architectures and sizes would provide deeper insights into how ontology memorization changes, also in relation to model advancements over time.

Second, our investigation focused exclusively on the memorization of ID-label associations in ontologies, without analyzing their structural and relational content. Exploring more complex ontological relationships, such as subsumption/generalization, could provide deeper insights into how LLMs internalize structured semantic knowledge.

Third, while we analyzed memorization patterns, we did not investigate practical interventions to mitigate hallucinations, leaving open the question of how to enhance LLMs' reliability when dealing with structured knowledge. A promising direction for future work is exploring techniques to reduce hallucinations in LLMs when handling ontologies, potentially improving their ability to generate more accurate and reliable responses.

Moreover, this study opens several promising directions for future research. Future work could focus on refining prompting strategies to better assess how different formulations influence ID-label retrieval. Additionally, developing new metrics that complement those proposed in this study could provide a more nuanced understanding of memorization patterns. Finally, examining the impact of training data composition in a controlled setting—using models with transparent pre-training corpora, such as OLMO [19]—could offer valuable insights into how exposure to structured data shapes LLM memory.

Our study highlights the limitations of LLMs in retaining structured knowledge and underscores the need for further research on mitigating hallucinations and enhancing ontology-aware learning. We hope this work fosters further exploration into the interaction between LLMs and structured knowledge, as addressing these challenges will ultimately improve their ability to process structured data (e.g., for answer validation and enhancing factual accuracy) while reducing unintended memorization artifacts.

Data Availability Statement

We publicly release the resources from this study to support future research on the interaction between LLMs and structured knowledge. The code and dataset used in our experiments are available at: <https://github.com/marcobombieri/do-LLM-dream-of-ontologies>.

References

- [1] Michael Ashburner, Catherine Ball, Judith Blake, David Botstein, Heather Butler, Michael Cherry, Allan Davis, Kara Dolinski, Selina Dwight, Janan Eppig, et al. 2000. Gene ontology: Tool for the unification of biology. The Gene Ontology Consortium. *Nature Genetics* 25, 1 (May 2000), 25–29. DOI: <https://doi.org/10.1038/75556>
- [2] Stella Biderman, USVSN Sai Prashanth, Lintang Sutawika, Hailey Schoelkopf, Quentin Anthony, Shivanshu Purohit, and Edward Raff. 2023. Emergent and predictable memorization in large language models. In *Proceedings of the Advances in Neural Information Processing Systems*, Vol. 36. Curran Associates Inc., Red Hook, NY, 28072–28090. Retrieved from https://proceedings.neurips.cc/paper_files/paper/2023/file/59404fb89d6194641c69ae99ecd8f6d-Paper-Conference.pdf
- [3] Stella Biderman, Hailey Schoelkopf, Quentin Gregory Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, et al. 2023. Pythia: A suite for analyzing large language models across training and scaling. In *Proceedings of the 40th International Conference on Machine Learning*. PMLR, 2397–2430. Retrieved from <https://proceedings.mlr.press/v202/biderman23a.html>
- [4] BigScience Workshop. 2023. BLOOM: A 176B-parameter open-access multilingual language model. arXiv:2211.05100. Retrieved from <https://arxiv.org/abs/2211.05100>
- [5] Derian Boer, Fabian Koch, and Stefan Kramer. 2024. Harnessing the power of semi-structured knowledge and LLMs with triplet-based prefiltering for question answering. arXiv:2409.00861. Retrieved from <https://arxiv.org/abs/2409.00861>
- [6] Gavin Brown, Mark Bun, Vitaly Feldman, Adam Smith, and Kunal Talwar. 2021. When is memorization of irrelevant training data necessary for high-accuracy learning? In *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing (Virtual)*. ACM, New York, NY, 123–132. DOI: <https://doi.org/10.1145/3406325.3451131>
- [7] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D. Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. In *Proceedings of the*

- Advances in Neural Information Processing Systems, Virtual*, Vol. 33. Curran Associates, Inc., 1877–1901. Retrieved from https://proceedings.neurips.cc/paper_files/paper/2020/file/1457c0d6bfbcb4967418bfb8ac142f64a-Paper.pdf
- [8] Nicholas Carlini, Daphne Ippolito, Matthew Jagielski, Katherine Lee, Florian Tramer, and Chiyuan Zhang. 2023. Quantifying memorization across neural language models. arXiv:2202.07646. Retrieved from <https://arxiv.org/abs/2202.07646>
 - [9] Nicholas Carlini, Chang Liu, Úlfar Erlingsson, Jernej Kos, and Dawn Song. 2019. The secret sharer: Evaluating and testing unintended memorization in neural networks. In *Proceedings of the 28th USENIX Security Symposium*. USENIX Association, 267–284. Retrieved from <https://www.usenix.org/conference/usenixsecurity19/presentation/carlini>
 - [10] Kent Chang, Mackenzie Cramer, Sandeep Soni, and David Bamman. 2023. Speak, memory: An archaeology of books known to ChatGPT/GPT-4. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 7312–7327. DOI: <https://doi.org/10.18653/v1/2023.emnlp-main.453>
 - [11] Wei-Lin Chiang, Lianmin Zheng, Ying Sheng, Anastasios Nikolas Angelopoulos, Tianle Li, Dacheng Li, Hao Zhang, Banghua Zhu, Michael Jordan, Joseph E. Gonzalez, and Ion Stoica. 2024. Chatbot Arena: An open platform for evaluating LLMs by human preference. arXiv:2403.04132. Retrieved from <https://arxiv.org/abs/2403.04132>
 - [12] Yung-Sung Chuang, Yujia Xie, Hongyin Luo, Yoon Kim, James R. Glass, and Pengcheng He. 2024. DoLa: Decoding by contrasting layers improves factuality in large language models. arXiv:2309.03883. Retrieved from <https://arxiv.org/abs/2309.03883>
 - [13] Chunyuan Deng, Yilun Zhao, Xiangru Tang, Mark Gerstein, and Arman Cohan. 2024. Investigating data contamination in modern benchmarks for large language models. arXiv:2311.09783. Retrieved from <https://arxiv.org/abs/2311.09783>
 - [14] Vitaly Feldman. 2020. Does learning require memorization? A short tale about a long tail. In *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing*. ACM, New York, NY, 954–959. DOI: <https://doi.org/10.1145/3357713.3384290>
 - [15] Johannes Frey, Lars-Peter Meyer, Natanael Arndt, Felix Brei, and Kirill Bulert. 2023. Benchmarking the abilities of large language models for RDF knowledge graph creation and comprehension: How well do LLMs speak Turtle? arXiv:2309.17122. Retrieved from <https://arxiv.org/abs/2309.17122>
 - [16] Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, et al. 2020. The Pile: An 800GB dataset of diverse text for language modeling. arXiv:2101.00027. Retrieved from <https://arxiv.org/abs/2101.00027>
 - [17] Fausto Giunchiglia and Ilya Zaihrayeu. 2009. *Lightweight Ontologies*. Springer, Boston, 1613–1619. DOI: https://doi.org/10.1007/978-0-387-39940-9_1314
 - [18] Clive William John Granger. 1969. Investigating causal relations by econometric models and cross-spectral methods. *Econometrica* 37, 3 (July 1969), 424–438. Retrieved from <https://ideas.repec.org/a/ect/emetrp/v37y1969i3p424-38.html>
 - [19] Dirk Groeneveld, Iz Beltagy, Evan Pete Walsh, Akshita Bhagia, Rodney Kinney, Oyvind Tafjord, Ananya Harsh Jha, Hamish Ivison, Ian Magnusson, Yizhong Wang, et al. 2024. OLMo: Accelerating the science of language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, 15789–15809. DOI: <https://doi.org/10.18653/V1/2024.ACL-LONG.841>
 - [20] Yuan He, Jiaoyan Chen, Ernesto Jiménez-Ruiz, Hang Dong, and Ian Horrocks. 2023. Language model analysis for ontology subsumption inference. In *Findings of the Association for Computational Linguistics: ACL*. Association for Computational Linguistics, 3439–3453. DOI: <https://doi.org/10.18653/v1/2023.findings-acl.213>
 - [21] Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. arXiv:2009.03300. Retrieved from <https://arxiv.org/abs/2009.03300>
 - [22] Sven Hertling and Heiko Paulheim. 2023. OLaLa: Ontology matching with large language models. In *Proceedings of the 12th Knowledge Capture Conference*. ACM, 131–139. DOI: <https://doi.org/10.1145/3587259.3627571>
 - [23] Jie Huang, Hanyin Shao, and Kevin Chen-Chuan Chang. 2022. Are large pre-trained language models leaking your personal information? In *Findings of the Association for Computational Linguistics: EMNLP*. Association for Computational Linguistics, 2038–2047. DOI: <https://doi.org/10.18653/v1/2022.findings-emnlp.148>
 - [24] Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, et al. 2025. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Transactions on Information Systems* 43, 2, Article 42 (Jan. 2025), 55 pages. DOI: <https://doi.org/10.1145/3703155>
 - [25] Shotaro Ishihara. 2023. Training data extraction from pre-trained language models: A survey. arXiv:2305.16157. Retrieved from <https://arxiv.org/abs/2305.16157>
 - [26] Paul Jaccard. 1901. Étude comparative de la distribution florale dans une portion des Alpes et des Jura. *Bulletin de la Société Vaudoise des Sciences Naturelles* 37 (1901), 547–579.
 - [27] Michael J. Kahana, Nicholas B. Diamond, and Akram Aka. 2022. Laws of Human Memory. PsyarXiv. DOI: <https://doi.org/10.31234/osf.io/aczu9>

- [28] Nikhil Kandpal, Eric Wallace, and Colin Raffel. 2022. Deduplicating training data mitigates privacy risks in language models. In *Proceedings of the 39th International Conference on Machine Learning*. PMLR, 10697–10707. Retrieved from <https://proceedings.mlr.press/v162/kandpal22a.html>
- [29] Adam Kilgarriff. 2007. Last words: Googleology is bad science. *Computational Linguistics* 33, 1 (2007), 147–151. DOI : <https://doi.org/10.1162/coli.2007.33.1.147>
- [30] Rob Koeling, Diana McCarthy, and John Carroll. 2005. Domain-specific sense distributions and predominant sense acquisition. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 419–426. Retrieved from <https://aclanthology.org/H05-1053>
- [31] Katherine Lee, Daphne Ippolito, Andrew Nystrom, Chiyuan Zhang, Douglas Eck, Chris Callison-Burch, and Nicholas Carlini. 2022. Deduplicating training data makes language models better. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, 8424–8445. DOI : <https://doi.org/10.18653/v1/2022.acl-long.577>
- [32] Eric Lehman, Sarthak Jain, Karl Pichotta, Yoav Goldberg, and Byron Wallace. 2021. Does BERT pretrained on clinical notes reveal sensitive data? In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, 946–959. DOI : <https://doi.org/10.18653/v1/2021.naacl-main.73>
- [33] Vladimir Iosifovich Levenshtein. 1966. Binary codes capable of correcting deletions, insertions and reversals. *Soviet Physics Doklady* 10, 8 (Feb. 1966), 707–710.
- [34] Zhenhua Liu, Tong Zhu, Chuanyuan Tan, Bing Liu, Haonan Lu, and Wenliang Chen. 2024. Probing language models for pre-training data detection. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, 1576–1587. DOI : <https://doi.org/10.18653/v1/2024.acl-long.86>
- [35] Xun Lu, Liangjun Su, and Halbert White. 2017. Granger causality and structural causality in cross-section and panel data. *Econometric Theory* 33, 2 (2017), 263–291. Retrieved from <https://www.jstor.org/stable/26173622>
- [36] Nils Lukas, Ahmed Salem, Robert Sim, Shruti Tople, Lukas Wutschitz, and Santiago Zanella Béguelin. 2023. Analyzing leakage of personally identifiable information in language models. In *Proceedings of the 44th IEEE Symposium on Security and Privacy*. IEEE, 346–363. DOI : <https://doi.org/10.1109/SP46215.2023.10179300>
- [37] Huu Tan Mai, Cuong Xuan Chu, and Heiko Paulheim. 2024. Do LLMs really adapt to domains? An ontology learning perspective. arXiv:2407.19998. Retrieved from <https://arxiv.org/abs/2407.19998>
- [38] Christopher J. Mungall, Carlo Torniai, Georgios V. Gkoutos, Suzanna E. Lewis, and Melissa A. Haendel. 2012. Uberon, an integrative multi-species anatomy ontology. *Genome Biology* 13, 1 (Jan. 2012), 20 pages. DOI : <https://doi.org/10.1186/gb-2012-13-1-r5>
- [39] Niels Mündler, Jingxuan He, Slobodan Jenko, and Martin Vechev. 2024. Self-contradictory hallucinations of large language models: Evaluation, detection and mitigation. arXiv:2305.15852. Retrieved from <https://arxiv.org/abs/2305.15852>
- [40] World Health Organization. 2019. International Classification of Diseases, 10th Revision (ICD-10). Retrieved October 15, 2024 from <https://icd.who.int/browse10>
- [41] Leonardo Ranaldi, Elena Sofia Ruzzetti, and Fabio Massimo Zanzotto. 2023. PreCog: Exploring the relation between memorization and performance in pre-trained language models. In *Proceedings of the 14th International Conference on Recent Advances in Natural Language Processing*. INCOMA Ltd. Shoumen, 961–967.
- [42] Oscar Sainz, Jon Campos, Iker García-Ferrero, Julen Etxaniz, Oier Lopez de Lacalle, and Eneko Agirre. 2023. NLP evaluation in trouble: On the need to measure LLM data contamination for each benchmark. In *Findings of the Association for Computational Linguistics: EMNLP*. Association for Computational Linguistics, 10776–10787. DOI : <https://doi.org/10.18653/v1/2023.findings-emnlp.722>
- [43] Weijia Shi, Anirudh Ajith, Mengzhou Xia, Yangsibo Huang, Daogao Liu, Terra Blevins, Danqi Chen, and Luke Zettlemoyer. 2024. Detecting pretraining data from large language models. arXiv:2310.16789. Retrieved from <https://arxiv.org/abs/2310.16789>
- [44] Charles Spearman. 1904. The proof and measurement of association between two things. *The American Journal of Psychology* 15, 1 (1904), 72–101. Retrieved from <http://www.jstor.org/stable/1412159>
- [45] Kai Sun, Yifan Ethan Xu, Hanwen Zha, Yue Liu, and Xin Luna Dong. 2024. Head-to-tail: How knowledgeable are Large Language Models (LLMs)? A.K.A. Will LLMs replace knowledge graphs? In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*. Association for Computational Linguistics, 311–325. DOI : <https://doi.org/10.18653/V1/2024.NAACL-LONG.18>
- [46] Kushal Tirumala, Aram H. Markosyan, Luke Zettlemoyer, and Armen Aghajanyan. 2022. Memorization without overfitting: Analyzing the training dynamics of large language models. In *Proceedings of the Advances in Neural*

- Information Processing Systems*, Vol. 35. Curran Associates, Inc., 38274–38290. Retrieved from https://proceedings.neurips.cc/paper_files/paper/2022/file/fa0509f4dab6807e2cb465715bf2d249-Paper-Conference.pdf
- [47] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. LLaMA: Open and efficient foundation language models. arXiv:2302.13971. Retrieved from <https://arxiv.org/abs/2302.13971>
 - [48] Denny Vrandečić and Markus Krötzsch. 2014. Wikidata: A free collaborative knowledgebase. *Communications of the ACM* 57, 10 (Sept. 2014), 78–85. DOI: <https://doi.org/10.1145/2629489>
 - [49] Weiqi Wu, Chengyue Jiang, Yong Jiang, Pengjun Xie, and Kewei Tu. 2023. Do PLMs know and understand ontological knowledge? In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, 3080–3101. DOI: <https://doi.org/10.18653/v1/2023.acl-long.173>
 - [50] Yue Zhang, Yafu Li, Leyang Cui, Deng Cai, Lemao Liu, Tingchen Fu, Xinting Huang, Enbo Zhao, Yu Zhang, Yulong Chen, et al. 2023. Siren’s song in the AI ocean: A survey on hallucination in large language models. arXiv:2309.01219. Retrieved from <https://arxiv.org/abs/2309.01219>
 - [51] Baohang Zhou, Zezhong Wang, Lingzhi Wang, Hongru Wang, Ying Zhang, Kehui Song, Xuhui Sui, and Kam-Fai Wong. 2024. DPDLLM: A black-box framework for detecting pre-training data from large language models. In *Findings of the Association for Computational Linguistics: ACL, Virtual Meeting*. Association for Computational Linguistics, 644–653. DOI: <https://doi.org/10.18653/v1/2024.findings-acl.35>
 - [52] Zhenhong Zhou, Jiuyang Xiang, Chaomeng Chen, and Sen Su. 2024. Quantifying and analyzing entity-level memorization in large language models. In *Proceedings of the 38th AAAI Conference on Artificial Intelligence and 36th Conference on Innovative Applications of Artificial Intelligence and 14th Symposium on Educational Advances in Artificial Intelligence*. AAAI Press, 19741–19749. DOI: <https://doi.org/10.1609/aaai.v38i17.29948>
 - [53] Derui Zhu, Dingfan Chen, Qing Li, Zongxiong Chen, Lei Ma, Jens Grossklags, and Mario Fritz. 2024. PoLLMgraph: Unraveling hallucinations in large language models via state transition dynamics. In *Findings of the Association for Computational Linguistics: NAACL*. Association for Computational Linguistics, 4737–4751. DOI: <https://doi.org/10.18653/v1/2024.findings-naacl.294>

Received 26 February 2024; revised 1 February 2025; accepted 10 March 2025